Détection de changements entre vidéos aériennes avec trajectoires arbitraires

Nicolas Bourdis

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

EADS

TELECOM ParisTech

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Signal et Image »

présentée et soutenue publiquement par

Nicolas BOURDIS

le 24 mai 2013

DÉTECTION DE CHANGEMENTS ENTRE VIDÉOS AÉRIENNES AVEC TRAJECTOIRES ARBITRAIRES

Directeur de thèse : **Hichem SAHBI**
Co-encadrement de la thèse : **Denis MARRAUD**

Jury

M. Jocelyn CHANUSSOT, Professeur, INP Grenoble	Rapporteur
M. Cédric RICHARD, Professeur, Université de Nice Sophia-Antipolis	Rapporteur
M. Rachid DERICHE, Directeur de recherche, INRIA Sophia-Antipolis	Examinateur
Mme Florence TUPIN, Professeur, Télécom ParisTech	Examinateur
M. Hichem SAHBI, Chargé de recherche CNRS (HDR), Télécom ParisTech	Directeur de thèse
M. Denis MARRAUD, Senior Expert, EADS Innovation Works	Co-encadrant de thèse

TELECOM ParisTech
École de l'Institut Télécom - Membre de ParisTech
46 Rue Barrault - 75013 Paris

EADS France - Innovation Works
12 Rue Pasteur BP 76 - 92150 Suresnes

THÈSE

Résumé

Les activités basées sur l'exploitation de données vidéo se sont développées de manière fulgurante ces dernières années. En effet, non seulement avons-nous assisté à une démocratisation de certaines de ces activités, telles que la vidéo-surveillance, mais également à une diversification importante des applications opérationnelles (e.g. suivi de ressources naturelles, reconnaissance aérienne et bientôt satellite). Cependant, le volume de données vidéo généré est aujourd'hui astronomique et l'efficacité des activités correspondantes est limitée par le coût et la durée nécessaire à l'interprétation humaine de ces données vidéo.

Par conséquent, l'analyse automatique de flux vidéos est devenue une problématique cruciale pour de nombreuses applications. Les travaux réalisés dans le cadre de cette thèse s'inscrivent dans ce contexte, et se concentrent plus spécifiquement sur l'analyse automatique de vidéos aériennes. En effet, outre le problème du volume de données, ce type de vidéos est particulièrement difficile à exploiter pour un analyste image, du fait des variations de points de vue, de l'étroitesse des champs de vue, de la mauvaise qualité des images, etc. Pour aborder ces difficultés, nous avons choisi de nous orienter vers un système semi-automatique permettant d'assister l'analyste image dans sa tâche, en suggérant des zones d'intérêt potentiel par détection de changements.

Plus précisément, l'approche développée dans le cadre de cette thèse cherche à exploiter les données disponibles au maximum de leur potentiel, afin de minimiser l'effort requis pour l'utilisateur et de maximiser les performances de détection. Pour cela, nous effectuons une modélisation tridimensionnelle des apparences observées dans les vidéos de référence. Cette modélisation permet ensuite d'effectuer une détection en ligne des changements significatifs dans une nouvelle vidéo, en identifiant les déviations d'apparence par rapport aux modèles de référence. Des techniques spécifiques ont également été proposées pour effectuer l'estimation des paramètres d'acquisition ainsi que l'atténuation des effets de l'illumination. De plus, nous avons développé plusieurs techniques de consolidation permettant d'exploiter la connaissance a priori relative aux changements à détecter.

L'intérêt de notre approche de détection de changements est démontré dans ce manuscrit de thèse, par la présentation des résultats issus de son évaluation minutieuse et systématique. Cette évaluation a été effectuée à l'aide de données réelles et synthétiques permettant d'analyser, d'une part la robustesse de l'approche par rapport à des perturbations réalistes (e.g. bruit, artefacts de compression, apparences et effets complexes, etc), et d'autre part la précision des résultats en conditions contrôlées.

Mots-Clés

Détection de changements, Masque de changements, Données d'observation, Vidéo aérienne, Modélisation 3D d'apparence, Quad-Tree augmenté, Redondance, Géo-localisation, Interpolation de poses, Asservissement visuel, Atténuation de l'illumination, Consolidation temporelle, Lissage temporel, Optimisation spatio-temporelle, Propagation de croyance, Binarisation, Retour interactif de pertinence, Descripteur de régions

English PhD Title

CHANGE DETECTION IN AERIAL VIDEOS WITH ARBITRARY TRAJECTORIES

Abstract

Business activities based on the use of video data have developed at a dazzling speed these last few years. Indeed, not only has the market of some of these activities widely expanded, such as video-surveillance, but the operational applications have also greatly diversified (e.g. natural resources monitoring, aerial video and soon satellite video intelligence). However, nowadays, the volume of generated data has become overwhelming and the efficiency of these activities is now limited by the cost and the time required by the human interpretation of this video data.

As a consequence, automatic analysis of video streams has become a critical problem for numerous applications. The work conducted in this thesis fall within this context, and focuses more specifically on the automatic analysis of aerial videos. Indeed, besides the problem regarding the data volumes, this type of videos is particularly difficult to use for an image analyst, due to the frequent viewpoint variations, the narrow fields of view, the poor image quality, etc. To address these difficulties, we chose to direct our work toward a semi-automatic system used to assist the image analyst in his task, by suggesting areas of potential interest identified using change detection.

More precisely, the approach developed in this thesis tries to use available data up to its full potential, in order to minimize the effort required from the user and to maximize the detection effectivity. For that purpose, our approach proceeds to a tridimensional modeling of the appearances observed in the reference videos. Such a modeling then enables the online detection of significant changes in a new video, by identifying appearance deviations with respect to the reference models. Specific techniques have also been developed to estimate the acquisition parameters and to attenuate illumination effects. Moreover, we developed several consolidation techniques making use of a priori knowledge related to targeted changes, in order to improve detection accuracy.

The interest of our change detection approach is demonstrated in this thesis using the results obtained from a resolute and thorough evaluation. This evaluation was carried out using both real and synthetical data, in order to analyze respectively the robustness of our approach with respect to realistic acquisition conditions (e.g. noise, compression artifacts, complex appearances and illumination effects, etc) and the accuracy of the results under controlled conditions.

Note that the body of this thesis is written in French, but an appendix providing an English synthesis of the contributions is included (see appendix A.2).

Keywords

Change detection, Change mask, Observation data, Aerial video, Augmented Quad-Tree, 3D appearance modeling, Redundancy, Geo-localization, Pose interpolation, Visual servoing, Illumination attenuation, Temporal consolidation, Temporal smoothing, Spatio-temporal optimization, Belief propagation, Binarization, Relevance feedback, Region descriptor

Remerciements

« [. . .] parce que, (les Anciens) s'étant élevés jusqu'à un certain degré où ils nous ont portés, le moindre effort nous fait monter plus haut, et avec moins de peine et moins de gloire nous nous trouvons au-dessus d'eux. C'est de là que nous pouvons découvrir des choses qu'il leur était impossible d'apercevoir. Notre vue a plus d'étendue, et, quoiqu'ils connussent aussi bien que nous tout ce qu'ils pouvaient remarquer de la nature, ils n'en connaissaient pas tant néanmoins, et nous voyons plus qu'eux. »

Blaise Pascal. 1647**, Préface de « Traité du vide ».**

Les travaux présentés dans ce manuscrit de thèse représentent ma contribution à la progression de la connaissance, qui, je l'espère, permettra à d'autres de « monter plus haut » et de « voir plus loin ». Quoiqu'il en soit, cette contribution est également le fruit de la bonne volonté de nombreuses autres personnes, que je souhaiterais ici remercier (sans toutefois pousser jusqu'à les qualifier « d'Anciens ») . . .

Pour commencer, je souhaiterais exprimer ma plus profonde gratitude à Hichem et Denis, sans qui cette thèse n'aurait assurément pas été aussi productive. Ils ont su se rendre disponibles, malgré des emplois du temps chargés, pour fournir un suivi régulier de mes travaux. Ils m'ont également fait profiter de leur précieuse connaissance des mondes scientifiques académiques et industriels. Enfin et surtout, ils ont su partager avec moi leur expertise technique, me permettant de me familiariser avec des problèmes complexes, et ainsi, de proposer à mon tour des solutions pertinentes.

Je voudrais également remercier les membres de mon jury de thèse, Jocelyn, Cédric, Florence et Rachid, qui, malgré la justesse des délais qui leur ont été donnés, ont porté un intérêt attentif à mes travaux et ont pris les dispositions nécessaires pour assister à ma soutenance. Je les remercie sincèrement d'avoir ainsi accepté d'être les garants de la qualité scientifique de mes travaux.

Enfin, je voudrais remercier toutes les personnes qui m'ont fait confiance à des moments clés et, par leurs avis éclairés, au cours de discussions passionnantes ou de chaleureux moments de détente, m'ont soutenu durant ces trois années et m'ont aidé à les mener à bien. Pour tout cela, je remercie en particulier Livier et Franck, et plus généralement mes collègues d'EADS et de Télécom, mes amis et ma famille.

Ces années de doctorat représentent pour moi l'aboutissement de 28 *années d'éveil, d'apprentissage puis d'études, au cours desquelles les acteurs de mon orientation ont été innombrables. Toutefois, je tiens à mentionner plus particulièrement mes parents, Isabelle et Pierre, qui ont (entre autres !) su éveiller en moi un désir de compréhension du monde qui m'est aujourd'hui indispensable, ainsi que mon épouse, Élodie, source continuelle d'inspiration. Ces quelques lignes ne rendront certainement pas justice à leur rôle fondamental, mais j'espère qu'elles parviendront à leur signifier la mesure de ma reconnaissance ...*

Nicolas

Table des matières

Liste des figures

Liste des algorithmes

Liste des tableaux

Chapitre 1

Introduction

LES activités basées sur l'exploitation de données vidéo se sont développées de manière fulgurante ces dernières années. En effet, non seulement avons-nous assisté à une démocratisation de certaines de ces activités, telles que la vidéo-surveillance, mais également à une diversification importante des applications opérationnelles (e.g. suivi de ressources naturelles, reconnaissance aérienne et bientôt satellite). Cependant, le volume de données vidéo généré est aujourd'hui astronomique et l'efficacité des activités correspondantes est limitée par le coût et la durée nécessaire à l'interprétation humaine de ces données vidéo.

Par conséquent, l'analyse automatique de flux vidéos est devenue une problématique cruciale pour de nombreuses applications. Les travaux réalisés dans le cadre de cette thèse s'inscrivent dans ce contexte, et se concentrent plus spécifiquement sur l'analyse automatique de vidéos aériennes [46, 64, 68]. En effet, outre le problème du volume de données, les vidéos aériennes sont particulièrement difficiles à exploiter pour un analyste image, du fait des variations de points de vue, de l'étroitesse des champs de vue, de la mauvaise qualité des images, etc. Pour aborder ces difficultés, nous avons choisi de nous orienter vers un système semi-automatique permettant d'assister l'analyste image dans sa tâche, en suggérant des zones d'intérêt potentiel par détection de changements.

La suite de ce chapitre d'introduction présente, à la section 1.1, une description détaillée des difficultés posées par le problème de l'interprétation de vidéos aériennes, et montre comment le cadre de la détection de changements peut constituer une solution adaptée pour l'assistance à l'analyse des observations. Par la suite, la section 1.2 introduit la problématique de la détection de changements, d'abord de manière générale puis de manière spécifique par rapport aux applications visées. Enfin, la section 1.3 présente l'organisation de ce manuscrit de thèse, puis récapitule les productions et publications générées dans le cadre de cette thèse et fournit une brève description des contributions techniques.

Sommaire

1.1 Positionnement du problème

1.1.1 Contexte opérationnel

Le nombre de nouveaux produits et services commerciaux basés sur l'exploitation de données vidéos n'a cessé d'augmenter ces dernières années. Les activités associées sont très diverses et s'intéressent par exemple à la reconnaissance géographique (e.g. recherche de survivants lors de catastrophes naturelles, analyse de théâtres d'opérations, détection d'engins explosifs improvisés) ou la surveillance (e.g. suivi de l'évolution de feux de forêts, sécurisation de sites, surveillance d'activités terroristes). L'analyse de vidéos ou de séquences d'images, qui est encore très majoritairement prise en charge par des équipes d'opérateurs humains, est donc devenue une tâche essentielle pour de nombreuses applications, à la fois civiles et militaires.

En parallèle, les progrès technologiques ont significativement amélioré les capacités des systèmes d'acquisitions, par exemple vis-à-vis de la résolution des images, des fréquences d'acquisition, des taux de compression, des capacités de stockage, etc. L'agence américaine de recherche avancée pour la défense (DARPA) a par exemple récemment annoncé le déploiement prochain d'un système aérien de surveillance vidéo appelé ARGUS. Ce système, voué a être porté par des drones, permettra de capturer un volume astronomique de données vidéo, avec une taille d'image de 1.8 giga-pixels, une fréquence de 12 images par seconde et sur une durée a priori illimitée.

Ainsi, associée à l'augmentation du nombre de caméras déployées, l'évolution technologique s'est traduite en pratique par une explosion de la quantité de données générées. Par conséquent, de nos jours, il n'est pas rare que les opérateurs chargés de l'analyse vidéo doivent rester concentrés durant de longues heures en étant attentifs à plusieurs flux en parallèle. De plus, dans un bon nombre d'applications, les événements ou objets recherchés sont relativement rares (e.g. intrusion sur site privé, vol ou agression, survivant d'une avalanche), ce qui demande une concentration continue malgré le caractère inintéressant de la majorité des données. Tout ceci rend donc l'analyse de grands volumes de données extrêmement éprouvante.

1.1.2 Difficultés d'analyse

Outre le problème dû au volume des données, la tâche de l'analyse vidéo en elle-même, qui consiste généralement en la détection d'objets ou d'événements spécifiques, n'est pas triviale. En effet, l'interprétation de la scène observée peut être gênée par la faible qualité des images issues du flux vidéo. Cette faible qualité génère des variations importantes de l'apparence des objets observés. Ces variations peuvent par exemple être dues à la présence de bruit dans les images, à l'étroitesse de la gamme dynamique, qui peut engendrer des zones trop ou trop peu exposées en présence de forts contrastes dans la scène, ou encore à des variations d'illumination dans la scène.

Dans le cas de caméras mobiles, et plus précisément dans le cas de l'observation aérienne, ces variations d'apparence prennent une importance encore plus considérable. Pour commencer, le fait que la plate-forme d'acquisition soit mobile peut introduire un léger flou dans les images, qui peut être accentué par les conditions extérieures, telles que les vibrations dues au vent. Par ailleurs, les conditions météorologiques, comme par exemple la pluie ou le brouillard, affectent également les acquisitions de manière beaucoup plus visible et inévitable que dans le cas de caméra fixes. Ces conditions météorologiques peuvent réduire considérablement la qualité des images, par exemple en générant des déformations dues aux gouttes d'eau, des diminutions de contraste, etc. Ainsi, les variations d'apparence du contenu de la scène observée constituent une gêne importante pour l'interprétation des acquisitions, et peuvent donc perturber l'analyse vidéo.

Le manque de contexte est également un facteur important qui complexifie l'interprétation des acquisitions. Ce facteur est lié à l'étroitesse du champ de vue, qui fait que pour obtenir une résolution au sol correcte permettant une bonne compréhension de la scène observée, il est généralement nécessaire d'utiliser un facteur de zoom important qui empêche d'avoir une vision globale de la scène [1]. Ce manque de contexte a notamment pour effet de gêner la perception de la position relative des objets observés et peut aussi affecter l'interprétation du contenu des images.

Enfin, la variabilité des points de vues selon lesquels les images sont acquises nuit également à l'exploitation des acquisitions. En effet, les images d'observation aérienne peuvent être acquises selon des orientations très différentes et peu habituelles. Cette variabilité des points de vue demande un effort important pour la localisation spatiale du contenu des images, ce qui complexifie certaines tâches d'analyse, comme par exemple la comparaison de plusieurs observations. D'autre part, le fait que les points de vue et les orientations de caméra soient inhabituels peut gêner la perception d'une scène, par exemple vis-à-vis de la taille relative des objets.

Ces difficultés font de l'analyse vidéo une tâche exigeante et coûteuse, ce qui a deux conséquences en pratique. D'une part, elles peuvent mener à des erreurs d'analyse, qui débouchent sur des conséquences plus ou moins graves selon l'enjeu associé à chaque application. D'autre part, l'analyse détaillée de l'ensemble des données acquises étant irréaliste, il est courant qu'une large majorité de ces données soient simplement stockées en attente d'exploitation, puis finalement effacées sans avoir été exploitées. Cela est par exemple courant dans le cas de la vidéo-surveillance dans les lieux publics, où en l'absence d'enquête judiciaire, les données sont effacées au bout d'une durée réglementaire maximale d'un mois.

1.1.3 Potentiel d'une approche semi-automatique

Au vu des difficultés opérationnelles rencontrées dans le cadre de l'analyse vidéo, nous pouvons donc nous demander comment améliorer la performance et l'efficacité de l'opérateur dans sa tâche, ou en d'autres termes, comment maximiser l'information extraite des acquisitions disponibles pour les utiliser à leur plein potentiel.

Pour cela, la solution consistant à proposer une approche complètement automatique semble irréaliste, du fait de la maturité limitée des techniques d'interprétation automatique d'images. En effet, une telle solution présenterait le risque de commettre un nombre important d'erreurs, la rendant contre-productive en pratique, voire dangereuse dans le cas d'applications critiques.

Il semble en revanche plus réaliste d'adopter une approche semi-automatique, c'est-à-dire mettant ponctuellement l'opérateur à contribution pour augmenter la pertinence des résultats. Le principe d'une approche semi-automatique consiste en effet à combiner les avantages d'un traitement automatique, adapté pour l'exécution rapide et systématique d'opérations fastidieuses, avec ceux d'une analyse humaine, capable d'une grande précision pour les tâches de classification. Ce type d'approche est donc tout à fait approprié dans le cadre de l'analyse de grands volumes de données vidéo, qui nécessite à la fois une mise à l'écart rapide pour une majorité de données sans intérêt et une classification précise en présence de cas ambigus.

Plus précisément, une méthode possible peut consister à utiliser des algorithmes appropriés pour analyser de multiples flux vidéos et effectuer des suggestions à l'opérateur lorsque des zones d'intérêt potentiel sont détectées. L'opérateur pourrait alors effectuer lui-même la distinction entre les objets ou événements pertinents et les cas sans intérêt mais ambigus. Un avantage notable de cette méthode est qu'une fois passée la phase initiale de réglage, elle peut être mise en œuvre parallèlement à l'analyse de l'opérateur, qui garde la possibilité d'inspecter directement les données.

1. Cette difficulté est souvent évoquée en parlant du problème de l'observation à travers une paille (*soda straw view* dans la littérature, [64]).

Ainsi, le rôle central d'un système semi-automatique dans le cadre de l'assistance à l'analyse de données vidéo consiste à mettre à l'écart les données sans-intérêt, qui ne nécessitent pas l'attention de l'opérateur. Or, dans de nombreux cas applicatifs (e.g. détection d'intrusions sur site, détection d'engins explosifs improvisés), la distinction entre données sans-intérêt et données d'intérêt potentiel peut être effectuée en déterminant l'absence ou la présence de changements significatifs par rapport à une référence. Cette notion de changements significatifs est importante, car tous les changements dans la scène observée ne représentent pas un intérêt pour l'opérateur. Cette notion sera donc abordée plus en détails un peu plus loin.

Le problème de la comparaison entre plusieurs données, ou plus généralement de la détection de changements, qui représente une tâche associée à un bas niveau de sémantique (détection de changements génériques), constitue donc un prérequis pour de nombreuses tâches de niveau sémantique plus élevé (e.g. interprétation, classification des changements). Par conséquent, le problème de la détection de changements présente un intérêt considérable dans le cadre de l'analyse de données vidéo.

1.2 Problème de la détection de changements

Le terme *détection de changements* possède plusieurs significations dans la littérature scientifique, selon le domaine d'application sous-entendu. Dans le contexte de l'analyse d'un unique flux de données, il peut en effet signifier la détection de changements abrupts de comportement, comme par exemple les brusques changements de point de vue dans un film ou des attaques informatiques dans un trafic de données réseau (e.g. DDOS). Dans le contexte de deux ensembles de données catégorisées (i.e. discrètes) acquis à des dates différentes, il peut signifier la détection de changements de catégorie dans les données. Par exemple, étant donné deux images satellitaires d'une même région, il peut être intéressant de quantifier la progression des zones urbaines par rapport aux zones forestières. Dans ce cas, les observations sont d'abord classées en deux catégories (zone urbaine et zone forestière) de manière indépendante dans chaque image, puis les changements de catégorie sont détectés. Devant l'ambiguïté du terme, il convient donc de préciser ce qui est entendu par *détection de changements* dans le cadre de cette thèse.

Dans ce manuscrit, nous définissons le problème de la détection de changements comme la tâche visant à comparer des données d'observation radiométriques acquises à des dates différentes, dans le but d'y détecter les changements significatifs. Cette formulation permet de définir le périmètre du problème, par rapport à des problèmes proches ou connexes, en insistant sur les points importants. D'une part, nous nous intéressons à la comparaison de données acquises à des dates différentes, par opposition aux problèmes visant l'analyse d'une donnée unique (e.g. détection de changements abrupts [4], suivi d'objets mobiles [119], soustraction de fond [102]). D'autre part, nous nous intéressons à la comparaison d'observations radiométriques (i.e. de mesures de l'information lumineuse), par conséquent à valeurs continues bien qu'éventuellement quantifiées, par opposition aux problèmes traitant des données discrètes issues d'une classification préalable (e.g. détection de changements de catégories [73]). Enfin, nous nous intéressons à la détection de changements significatifs, ce qui introduit un caractère subjectif dépendant de l'application visée, par opposition aux problèmes intéressés par toute forme de changements (e.g. compression vidéo [9]) ou par des changements extrêmement subtils (e.g. détection de stéganographie dans des images [70]).

La suite de cette section est organisée de la manière suivante. La section 1.2.1 présente la problématique générale de la détection de changements ainsi que les spécificités liées à l'utilisation de vidéos aériennes. La section 1.2.2 analyse ensuite les différents types de changements rencontrés dans le cadre de l'observation aérienne, et propose une caractérisation des changements pertinents que nous chercherons à y détecter. Enfin, la section 1.2.3 pose les hypothèses

de travail de nos algorithmes et introduit le scénario applicatif générique visé par les travaux effectués.

1.2.1 Problématique générale

Pour schématiser, il est possible de faire le rapprochement entre le problème de la détection de changement et le jeu populaire des 7 différences, dont l'objectif consiste à identifier les changements entre deux images. La détection de changements vise un objectif similaire, mais dans le cas général où il n'y a pas forcément de correspondance pixel à pixel entre les images et où les différences dues aux changements à identifier ne sont pas les seules différences entre les images.

De manière plus formelle, le problème de la détection de changements [95] consiste à comparer une nouvelle donnée, qualifiée de donnée *de test*, par rapport à une donnée *de référence*. L'objectif de cette comparaison est double. D'une part, elle vise à détecter la présence ou l'absence de changements entre la donnée de référence et la donnée de test. D'autre part, elle peut aussi viser à localiser ces changements dans les données considérées. Cette localisation est déterminée via l'estimation du *masque de changement* associé à la donnée de test, qui attribue à chaque zone (pixel ou groupe de pixels) un label binaire selon qu'elle présente un intérêt potentiel ou qu'elle soit non pertinente. Dans le cadre de cette thèse, nous nous sommes intéressés à ces deux aspects, et dans la suite de ce manuscrit, nous désignerons par *détection de changements* le problème visant à la fois la détection et la localisation des changements.

Plus précisément, le problème de la détection de changements est constitué de plusieurs sous-problèmes différents. Naturellement, le sous-problème central correspond à la comparaison effective entre la donnée de test et la donnée de référence, dont l'objectif est d'identifier les zones contenant des changements d'intérêt potentiel. Un autre sous-problème important, lorsque les données considérées sont issues de points de vue différents, concerne la mise en cohérence géométrique des données, qui vise à déterminer les zones comparables entre des données non-alignées.

Par ailleurs, dans le cas où de multiples images de références et/ou de test peuvent être utilisées pour effectuer la comparaison des données, ce qui est typiquement le cas pour la détection de changements entre vidéos, il peut être intéressant de s'intéresser à deux autres sous-problèmes. Le premier correspond à la constitution d'une référence, dont l'objectif est de déterminer, par rapport à une image de test donnée, une référence optimale, pouvant consister en une image de référence particulière ou une composition de plusieurs images de référence. Le second sous-problème concerne la gestion de la similarité des images de test successives, qui peut permettre d'améliorer les performances en exploitant la nécessaire cohérence entre les résultats successifs. Une discussion plus détaillée de ces sous-problèmes sera effectuée dans le chapitre 2.

D'autre part, les techniques d'estimation du masque de changements peuvent bénéficier de différentes opportunités selon la nature des données considérées. Par exemple, en ne considérant qu'une paire d'observations (e.g. deux pixels), nous ne disposons que de leurs deux valeurs pour les comparer. En revanche, en considérant deux ensembles d'observations organisées spatialement (e.g. deux images), nous disposons de plus d'information et nous pouvons donc parvenir à de meilleures performances si nous parvenons à l'exploiter correctement. De la même manière, travailler sur des vidéos plutôt que sur des images rend disponible une dimension temporelle en supplément des deux dimensions spatiales. Ainsi, il est intuitif que, plus la quantité d'information disponible est importante, plus les performances de détection de changements peuvent être améliorées. Ce point sera notamment mis en évidence dans le chapitre 6, qui présente les résultats d'évaluation.

Par conséquent, dans les cas d'application où des données d'observation sont acquises régulièrement sur une même scène, il peut être intéressant d'exploiter autant de données de référence que possible. Cette idée soulève cependant une question concernant la marche à suivre en présence de changements entre les différentes données de référence. En effet, dès lors que les observations de référence considérées sont acquises à des instants différents (e.g. vidéo de référence, ensemble d'images et/ou de vidéos de référence), elles présentent le risque de contenir des changements survenus dans la scène observée entre les différentes acquisitions. Deux approches sont alors possibles. Une première approche peut consister à essayer de détecter ces changements dans les données de référence, afin d'ignorer la zone correspondante durant la comparaison avec la donnée de test. Une approche alternative peut également être de supposer que les changements survenus entre les observations de référence sont en fait des variations non pertinentes, qui correspondent à des changements dont la détection n'est pas souhaitée dans la donnée de test.

Cette seconde approche semble la plus pertinente dans le cadre de la détection de changements entre vidéos aériennes. En effet, prenons l'exemple de deux vidéos acquises sur une scène donnée à deux dates différentes, dont la première présente une grande variabilité sur une zone donnée, comme par exemple une route très fréquentée sur laquelle de nombreuses voitures circulent. Dans ce cas, il semble inutile de détecter les variations similaires comme changement d'intérêt potentiel dans la seconde vidéo, puisque nous savons que ces variations sont courantes sur cette zone.

Ainsi, ceci montre que la notion de pertinence d'un changement donné est relativement difficile à définir, et cela d'autant plus que la quantité de données de référence est importante. Une discussion plus détaillée de cette notion de changement pertinent, ainsi que des différentes catégories de changements, est donnée à la section suivante.

1.2.2 Catégories de changements

Outre la résolution des différents sous-problèmes évoqués plus haut, la principale difficulté dans le cadre de la détection de changements vient du fait que deux vues quelconques d'une même scène peuvent contenir un grand nombre de différences. Ces différences peuvent par exemple provenir de changements dans le contenu de la scène observée, de variations issues du processus d'acquisition des observations ou encore de la transmission des observations. La figure 1.1 présente l'exemple de deux images aériennes acquises depuis des points de vues différents à environ un mois d'intervalle. Il est possible de reconnaître assez aisément que les deux images représentent la même scène. Cependant, l'image des différences, obtenue par différence après un recalage simple, puis par seuillage, montre qu'un grand nombre de différences existent entre les intensités des couples de pixels correspondants. Ainsi, bien que la ressemblance entre ces deux images de la même scène soit importante, il est très peu probable qu'un couple de pixels correspondants aient la même intensité. Cela illustre donc bien la difficulté rencontrée dans le cadre de la détection de changements, qui vise à ne détecter que les changements significatifs et à ignorer les changements non-pertinents.

Cette notion de changements non-pertinents, que nous désignerons dans la suite du manuscrit par *variations* ou *variabilités*, pour insister sur le fait qu'elles sont sans intérêt dans le cadre de la détection de changements, peut dépendre de l'application visée. Cependant, un certain nombre de ces variations, qui sont de nature très diverses, sont communes à la plupart des cas applicatifs.

Il est ainsi fréquent que des différences sans intérêt dans le contenu de la scène observée donnent lieu à des variations gênantes pour la détection de changements, comme par exemple les variations de conditions d'illumination, les changements de direction ou d'intensité des ombres, les mouvements répétitifs dans la scène (e.g. feuilles d'arbres, vagues sur les plans d'eau), les

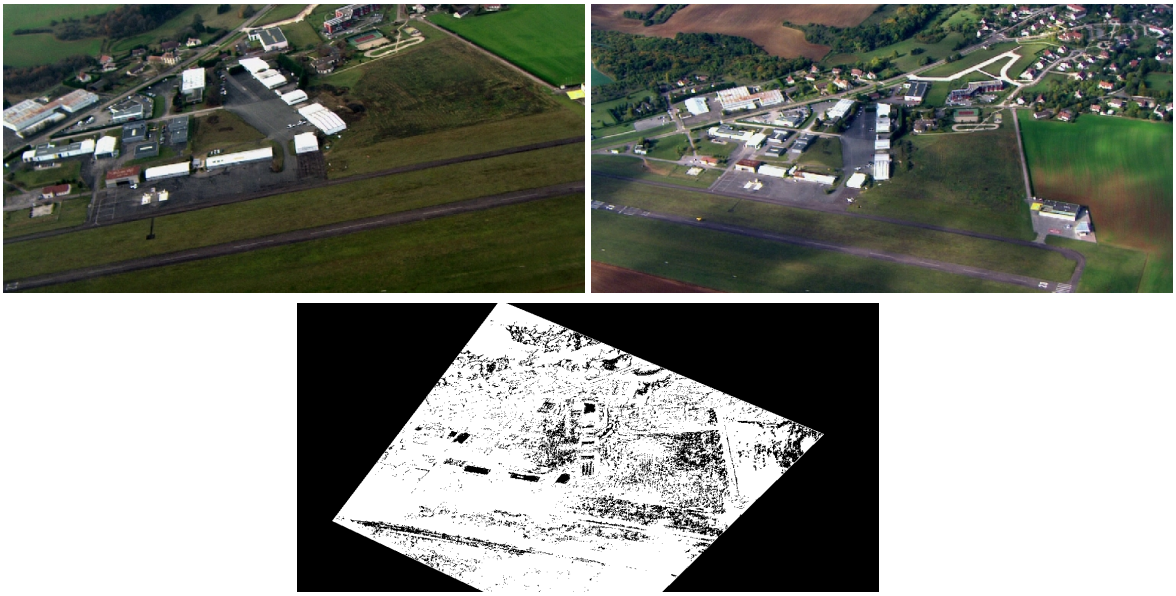


FIGURE 1.1 – *Cette figure présente deux images aériennes acquises sur la même scène à environ un mois d'intervalle (en haut) ainsi que l'image des différences seuillées après recalage simple (en bas), où les pixels dont la valeur a changé sont affichés en blanc. Ceci montre que de nombreuses différences peuvent exister entre deux vues quelconques d'une même scène, ce qui illustre la principale difficulté du problème de la détection de changements.*

variations de conditions météorologiques (e.g. pluie, brouillard, nuages), les variations saisonnières (e.g. neige, changements de végétation), et ainsi de suite.

Des variations gênantes peuvent également survenir lors de l'acquisition des images. Les plus évidentes sont celles dues aux effets géométriques issus des changements de point de vue. Toutefois, le processus d'acquisition lui-même génère également de nombreuses variations, provenant par exemple des halos lumineux dûs à l'optique du système d'acquisition, des saturations et débordements de registres d'acquisition ou encore du bruit généré par les capteurs de lumière. Enfin, le processus de transmission des images peut également générer des variations non-pertinentes, les plus fréquentes étant dues par exemple aux artefacts issus de la compression des images ou encore à la perte de paquets lors d'une transmission sans fil.

Ces variations non-pertinentes dans les observations sont le plus souvent sans intérêt dans des applications exploitant la détection de changements, et il est donc souhaitable qu'elles n'apparaissent pas dans les résultats de détection afin de ne pas submerger l'opérateur d'informations inutiles. Néanmoins, malgré leur caractère non-pertinent, ces variations peuvent causer des différences visuelles extrêmement marquées, ce qui les rend difficiles à traiter de manière automatique. De plus, le nombre et l'intensité de ces variations augmentent généralement avec le délai séparant l'acquisition des observations à comparer. Ceci est particulièrement vrai pour les variations issues des conditions météorologiques ou saisonnières et impose des contraintes supplémentaires sur les techniques utilisables pour les applications relatives à de tels cas.

Tout comme la notion de variation non-pertinente, celle de changement pertinent dépend aussi de l'application visée. La détection des changements d'apparence de la végétation, la détection de véhicules mobiles ou encore la détection de personnes peuvent ainsi être intéressantes pour des applications telles que le suivi de l'évolution de cultures ou de forêts, la vidéo-surveillance aérienne ou la recherche de survivants.

Dans le cadre de cette thèse, nous avons commencé par étudier le problème de la détection de changements dans le contexte général de données vidéo d'observation aérienne, puis nous nous sommes intéressés à une définition plus précise du type de changements pertinents à dé-

tecter. L'intérêt de disposer d'une définition claire de ces changements visés est qu'elle permet d'obtenir une connaissance a priori sur leurs caractéristiques. L'exploitation de cette connaissance a priori permettra notamment d'améliorer la pertinence des résultats en adéquation avec l'application visée par l'opérateur (voir chapitre 5).

Pour cela, nous avons étudié plus particulièrement la détection de changements correspondant à des structures ou objets artificiels fixes (e.g. bâtiments, champs, véhicules, personnes). L'intérêt de se restreindre à des structures ou objets artificiels vient du fait qu'ils possèdent généralement des frontières bien définies dans les images, ce qui facilite leur distinction, en particulier par rapport à certaines variations non-pertinentes (e.g. effets de l'illumination). D'autre part, dans le cadre de la détection de changements, les changements visés par la détection correspondent généralement à des objets fixes, par opposition aux objets mobiles qui peuvent être détectés et suivis de manière plus efficace par d'autres types de techniques.

Cette caractérisation des changements visés par la détection restreint nécessairement le nombre d'applications directes des algorithmes associés. Toutefois, les applications compatibles avec cette caractérisation restent nombreuses et variées, telles que par exemple le suivi de cultures, la recherche de survivants à des catastrophes naturelles, la surveillance d'activités terroristes, la mise à jour de bases de données cartographiques ou d'observation, et ainsi de suite.

1.2.3 Hypothèses de travail

Les sections précédentes ont mentionné les raisons qui font du problème de la détection de changements entre de multiples vidéos aériennes un problème vaste et difficile. Pour l'aborder, nous avons donc posé un certain nombre d'hypothèses afin de restreindre le champ des difficultés.

Ainsi, nous avons posé comme hypothèse de travail le fait que les vidéos à comparer soient acquises selon un intervalle de temps relativement faible, ce qui, comme évoqué plus haut, permet de limiter les variations subies par la scène observée (e.g. changement de végétation, neige). De plus, nous supposons que les vidéos considérées sont acquises dans des résolutions au sol similaires, ce qui permet de garantir qu'elles sont composées d'observations comparables. Malgré leur caractère limitatif, ces hypothèses restent compatibles avec la plupart des applications de recherche ou de surveillance. D'autre part, notons que nous ne faisons aucune hypothèse sur la trajectoire d'acquisition des vidéos aériennes, ni sur le fait que cette trajectoire soit connue.

De manière plus concrète, nous considérons, dans ce manuscrit, un scénario applicatif générique faisant intervenir une plate-forme d'observation aérienne (e.g. avion, hélicoptère, ballon, drone) qui effectue des passages réguliers sur une région géographique donnée. Ces passages réguliers se font selon un intervalle temporel allant de quelques heures à quelques jours et, à chaque passage, une nouvelle vidéo est acquise dans une gamme de résolution au sol approximativement constante, la trajectoire étant arbitraire par ailleurs.

1.3 Travaux réalisés

Les travaux réalisés dans le cadre de cette thèse portent sur la mise en œuvre d'une approche semi-automatique permettant la détection en ligne de changements dans des vidéos aériennes. Pour cela, nous avons notamment laissé de coté les problématiques liées à la reconstruction 3D et à l'accélération matérielle, pour concentrer notre effort sur les algorithmes de détection de changements et sur les techniques périphériques qui nous ont paru essentielles pour obtenir des performances satisfaisantes. En particulier, nous avons porté un effort considérable sur la manière d'exploiter au maximum l'information disponible dans les vidéos considérées, afin de minimiser l'effort requis pour l'utilisateur et de maximiser les performances de détection.

Notation	Signification
$a, b, c, \alpha \ldots$	Scalaires (minuscules)
$\mathbf{a}, \mathbf{b}, \mathbf{c}, \boldsymbol{\alpha} \ldots$	Vecteurs (lettres grasses)
$\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{ID} \ldots$	Matrices (capitales)
A^T	Transposée de la matrice A
$\tilde{X}$	Valeur grossière ou initiale de la variable X
$\hat{X}$	Estimateur de la variable X
$\overline{X}$	Moyenne de la variable X
$I, \{I_t\}_t$	Image ou séquence d'images
$\mathrm{P}, \{\mathrm{P}_t\}_t$	Matrice(s) de projection de caméra
$\mathbf{M}, \{\mathbf{M}_i\}_i$	Coordonnées 3D de point(s) dans la scène
$\mathbf{m}, \{\mathbf{m}_i\}_i$	Coordonnées 2D d'un point(s) dans une image
$[\![n,m]\!]$ pour $n < m,\ (n,m) \in \mathbb{N}^2$	Intervalle des nombres entiers compris entre n et m
$\langle \mathbf{u} \vert \mathbf{v} \rangle$	Produit scalaire entre les vecteurs $\mathbf{u}$ et $\mathbf{v}$

TABLE 1.1 – *Définition des notations utilisées le plus fréquemment dans ce manuscrit.*

Pour terminer ce chapitre d'introduction, la section 1.3.1 donne une description de l'organisation de la suite de ce manuscrit. Enfin, la section 1.3.2 dresse la liste des contributions techniques principales apportées par les travaux réalisés et récapitule les productions générées dans le cadre de cette thèse.

1.3.1 Organisation de l'exposé

Le chapitre 2 commence par donner une description détaillée des sous-problèmes associés au problème de la détection de changements entre vidéos. Il dresse également une taxonomie des différentes approches selon lesquelles les travaux de la littérature ont abordé ces sous-problèmes. Après l'analyse de l'état de l'art en détection de changements, ce chapitre propose une discussion des différentes manières d'évaluer les algorithmes associés. Enfin, ce chapitre détaille également les motivations justifiant l'orientation choisie pour nos travaux.

Les chapitres suivants présentent le détail technique des travaux réalisés dans le cadre de cette thèse. Pour plus de clarté, l'exposé est effectué en trois parties relativement indépendantes entre elles, les notations les plus courantes étant résumées dans la table 1.1.

Le chapitre 3 présente les méthodes de pré-traitement des données. Ces pré-traitements concernent, d'une part l'estimation des trajectoires et des paramètres d'acquisition des vidéos considérées, et d'autre part la conversion des observations dans une représentation invariante par rapport aux variations d'illumination.

Ensuite, le chapitre 4 présente, dans un premier temps, l'approche bi-dimensionnelle que nous avons développé pour la détection de changements dans des images aériennes. Les difficultés rencontrées pour aborder le cas de vidéos aériennes avec cette méthode justifient le passage à l'utilisation d'un modèle 3D d'apparence. La suite de ce chapitre décrit donc les algorithmes utilisés pour la génération de ce modèle 3D d'apparence, puis pour son exploitation pour la détection de changements.

Le chapitre 5 termine l'exposé technique avec la présentation des différentes méthodes développées pour consolider les détections, en exploitant la connaissance a priori relative aux changements à détecter. Pour cela, nous utilisons une consolidation temporelle des résultats successifs et une analyse fine des scores de détection pour l'estimation du masque de changements. Par la suite, une méthode de retour interactif de pertinence a été développée pour généraliser le principe de l'exploitation de connaissance a priori.

Ensuite, le chapitre 6 présente, dans un premier temps, les données utilisées pour l'évaluation des performances de notre approche de détection de changements. Dans un second temps, l'analyse systématique de cette approche et de ses différents modules de traitement est effectuée et les performances obtenues sont discutées en détail.

Pour finir, le chapitre 7 conclut ce manuscrit en effectuant un bilan des travaux réalisés dans le cadre de cette thèse et en donnant un certain nombre de perspectives pour la poursuite des travaux.

1.3.2 Contributions

L'ensemble des contributions techniques apportées par les travaux réalisés sont présentées en détail dans les chapitres techniques. Cependant, pour plus de clarté, les contributions principales sont synthétisées ci-dessous :

- Conception et développement d'une approche semi-automatique de détection de changements, permettant l'analyse hors-ligne de multiples vidéos de référence, puis la détection incrémentale de changements dans une vidéo de test donnée. Cette approche permet d'obtenir d'excellentes performances de détection de changements, et est ainsi capable de générer moins d'un faux positifs toutes les trois minutes (voir section 6.5), lorsque l'on considère une vidéo acquise à 5 images par seconde dont 10% des images contiennent un changement pertinent.
- Introduction d'un nouveau type de modèle 3D d'apparence, qui combine une structure de Quad-Tree, dont la résolution s'adapte aux données considérées, avec une carte d'élévations, pour modéliser la troisième dimension spatiale [2]. Par opposition à un Octree, ce modèle 3D permet une bonne robustesse d'extrapolation aux nouveaux points de vue, ainsi qu'une plus faible occupation mémoire, permettant de modéliser des scènes plus vastes.
- Conception et développement d'une méthode semi-automatique pour l'estimation hors-ligne de la trajectoire et des paramètres d'acquisition des vidéos de référence. Cette méthode est basée sur la saisie manuelle d'annotations sur quelques images-clés, puis sur la propagation automatique de ces annotations au reste des images de la vidéo considérée. Cette propagation automatique est effectuée grâce à l'estimation automatique du tenseur trifocal, défini entre deux images-clés et une image intermédiaire donnée, qui permet ensuite le transfert des annotations des images-clés vers l'image intermédiaire. Cette méthode permet notamment de traiter les vidéos dans lesquelles les paramètres de calibration sont variables, telles que celles faisant l'objet d'une stabilisation optique des images.
- Conception et développement d'une méthode d'asservissement visuel, permettant l'estimation en ligne de la trajectoire et des paramètres d'acquisition d'une vidéo à partir d'un modèle 3D de la scène observée. Cette méthode fonctionne en affinant l'estimation des paramètres d'acquisitions à l'aide de la transformation de recalage, entre d'une part l'image réelle, et d'autre part le rendu du modèle 3D vu depuis l'estimation courante des paramètres d'acquisition. De plus, cette méthode peut être adaptée pour correspondre à différentes contraintes en termes de précision, de rapidité, ou de variabilité des paramètres de calibration.
- Conception et développement d'une méthode de consolidation temporelle des résultats de détection de changements, qui permet l'amélioration des performances. Cette méthode, qui exploite une modélisation spécifique au problème considéré dans le cadre de la propa-

2. Notons que dans le cadre de cette thèse, nous ne nous sommes pas intéressé aux méthodes de reconstruction 3D. Les élévations utilisées sont issues du Modèle Numérique de Terrain SRTM3, mis gratuitement à disposition par la NASA.

gation de croyance, fonctionne en optimisant la cohérence spatio-temporelle des résultats successifs sur la vidéo de test.
- Introduction d'une méthode de retour interactif de pertinence, exploitant un descripteur de régions spécifique, permettant d'adapter les résultats de détection de changements en cohérence avec les besoins de l'utilisateur. Cette méthode permet en particulier d'éliminer la quasi-totalité des fausses alarmes résiduelles, tout en conservant la grande majorité des changements correctement détectés.
- Conception d'un descripteur de régions d'images utilisé dans le cadre de la méthode de retour interactif de pertinence. Ce descripteur combine plusieurs critères basés entre autres sur la forme des régions, leur intensité et leur couleur, et permet une bonne distinction entre les régions correspondant à des changements d'intérêt potentiel et les régions non pertinentes.

D'autre part, les travaux réalisés et les contributions apportées ont débouché sur les productions scientifiques suivantes :
- Article [14] publié et présenté oralement à la conférence *International Geoscience And Remote Sensing Symposium* (IGARSS) de 2011,
- Article [15] publié et présenté oralement à la conférence *International Geoscience And Remote Sensing Symposium* (IGARSS) de 2012,
- Article [16] publié à la conférence *International Geoscience And Remote Sensing Symposium* (IGARSS) de 2012,
- Brevet [13] en cours de dépôt auprès de l'Institut National de la Propriété Industrielle,
- Article [18] présentant le contenu du brevet, en attente de soumission dans le journal *IEEE Transactions on Geoscience and Remote Sensing*,
- Publication [12] d'une base de données de synthèse pour le benchmarking de techniques de détection de changements entre paire d'images aériennes,
- Rapport de recherche [17] publié dans la revue interne de Télécom ParisTech,
- Développement d'un démonstrateur basé sur Qt, OpenGL et OpenCV pour la démonstration des fonctionnalités développées.

Chapitre 2

État de l'art

La problématique de la détection de changements dans des vidéos aériennes soulève de nombreuses questions, comme par exemple celle de la constitution d'une référence correspondant à une image de test donnée, ou encore celle de la gestion de la similarité des images successives d'une vidéo, qui ont été évoquées dans le chapitre précédent. Les deux questions précédentes, qui découlent de l'exploitation de vidéos plutôt que d'images, sont très peu abordées dans la littérature. Plus généralement, les publications scientifiques qui abordent dans son ensemble la problématique de la détection de changements entre vidéos sont très rares.

En revanche, une bonne partie des autres questions soulevées par cette problématique, par exemple la gestion des effets géométriques ou des variations d'illumination, ont été étudiées indépendamment, dans la littérature relative à la détection de changements ou aux divers domaines connexes. De même, l'état de l'art relatif aux méthodes de comparaison de données d'observation est extrêmement riche. Pour mieux positionner les travaux réalisés par rapport aux techniques existantes, le présent chapitre propose une présentation thématique de l'état de l'art considérant plusieurs critères orthogonaux. Notons que, vu la diversité de la littérature correspondante, cette revue de l'état de l'art ne se veut pas exhaustive mais mentionne les approches les plus intéressantes, du point de vue de cette thèse, pour chacun des critères envisagés.

Pour commencer, la section 2.1 introduit une taxonomie des méthodes de détection de changements de la littérature. Cette taxonomie permet de présenter les techniques existantes de manière à mieux mettre en évidence les diverses approches possibles pour aborder les différents aspects de la problématique générale. Les sections 2.2 à 2.5 présentent les techniques existantes, en mentionnant leurs forces et leurs limites. Notons que les travaux publiés dans le cadre de cette thèse ont été clairement intégrés à cet état de l'art, de manière à permettre la lecture du présent chapitre de manière indépendante des chapitres techniques. La section 2.6 discute ensuite de la question de l'évaluation des algorithmes de détection de changements. Enfin, la section 2.7 effectue une synthèse de l'état de l'art, et motive les choix de conception relatifs aux travaux réalisés dans le cadre de cette thèse.

Sommaire

2.1 Taxonomie des techniques de détection de changements

Comme mentionné précédemment, la littérature dans le domaine de la détection de changements est extrêmement riche. La revue de littérature par Radke et al. [95], référence dans le domaine, a exploré les méthodes de détection de changements dans des paires d'images, qu'elles soient satellitaires, aériennes ou médicales. Elle mentionne également un certain nombre de méthodes travaillant sur des séquences d'images (ou séries temporelles) ainsi que quelques méthodes exploitant une vidéo, dans le cadre de la soustraction de fond. Au premier abord, il peut sembler pertinent de présenter les méthodes existantes selon le type de données traitées, comme par exemple les paires d'images, les séquences de quelques images ou les vidéos. Il est vrai que différents types de données engendrent des difficultés et des opportunités différentes. Cependant, une telle présentation ne met pas en évidence le fait que de nombreux problèmes sont communs à tous les types de données. Par conséquent, il nous semble plus intéressant de présenter les méthodes existantes selon la manière d'aborder plusieurs problématiques indépendantes, et d'analyser les diverses solutions proposées, éventuellement dans le cadre d'applications différentes de la détection de changements (e.g. soustraction de fond, génération de mosaïques, amélioration d'image, etc).

La revue de littérature par Radke et al. [95] a concentré son effort d'exploration sur la problématique de la comparaison d'images, mais en décrit brièvement deux autres. La première concerne les pré-traitements incluant les ajustements géométriques, essentiellement réduits au recalage des images, et les ajustements radiométriques, tels que l'atténuation des variations d'illumination. La seconde problématique concerne la question de la cohérence spatiale du masque de changements, qui vise à filtrer le bruit d'estimation afin d'obtenir un masque de changements régulier et contenant des frontières lisses entre changements et non-changements.

En revanche, cette revue n'aborde pas les problèmes soulevés par la détection de changements dans de multiples vidéos, qui sont abordés par certaines méthodes publiées ultérieurement [24, 91, 92]. Pour intégrer ces nouvelles méthodes à la taxonomie générale des techniques de détection de changements, nous introduisons donc la problématique de la constitution d'une référence correspondant à une image de test donnée. Ce problème est abordé plus en détail dans la section 2.2.

Par ailleurs, il nous semble réducteur de vouloir restreindre la gestion des effets géométriques et des variations d'illumination aux seuls pré-traitements sur les données, car la gestion de ces aspects peut faire partie intégrante de l'approche utilisée. Par opposition à l'approche de la revue, nous discutons donc dans la section 2.3 des techniques traitant de ces aspects, qu'elles soient utilisées en pré-traitements ou non, de manière indépendante des autres problématiques. Enfin nous présentons également dans la section 2.5 un certain nombre de techniques utilisées en post-traitement des résultats de détection, et permettant d'affiner les résultats.

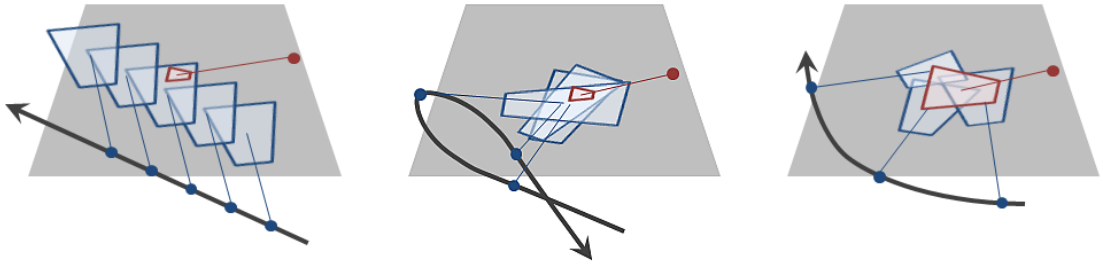
Par conséquent, la taxonomie que nous proposons dans ce chapitre analyse les méthodes de détection de changements selon les problématiques suivantes :

- Constitution d'une référence correspondant à une image de test donnée,
- Gestion des différentes sources de variabilité non pertinentes,
- Comparaison de la donnée de test avec la donnée de référence,
- Techniques de post-traitement pour l'affinage des résultats de détection.

Les sections suivantes font l'examen de l'état de l'art en détection de changements, selon l'angle de chacune des problématiques précédentes.

2.2 Constitution d'une référence pour une image donnée

Lorsqu'une vidéo de référence acquise sur une région donnée est disponible et que nous souhaitons détecter, dans une image de test, les changements par rapport à cette référence, le



(a) (b) (c)

FIGURE 2.1 – *Cette figure illustre les différents scénarios dans lesquels le problème de la constitution d'une référence peut se poser. Ce problème survient notamment lorsque la zone vue par l'image de test n'apparaît que dans un passage spécifique de la vidéo de référence (a), lorsque cette zone apparaît à plusieurs reprises et que les images correspondantes doivent être combinées (b) ou lorsque plusieurs images de la vidéo de référence sont nécessaires pour analyser entièrement l'image de test (c). La flèche noire représente la trajectoire de la vidéo de référence, les trapèzes bleus représentent les projections au sol des images de la vidéo de référence, et le trapèze rouge représente la projection au sol de l'image de test.*

premier problème à résoudre consiste à déterminer avec quelles données comparer l'image de test. En effet, la zone visible dans l'image de test peut n'apparaître que dans un passage spécifique de la vidéo de référence (voir 2.1a), et, pour obtenir un système efficace, il est alors nécessaire de savoir retrouver efficacement ce passage utile pour la détection de changements. Une autre difficulté réside dans le fait que la zone visible dans l'image de test peut apparaître dans plusieurs images de référence (voir 2.1b), en particulier lorsque la trajectoire d'acquisition de la vidéo de référence contient des boucles. Le même problème se pose également lorsqu'aucune image de référence unique n'est suffisante pour analyser entièrement l'image de test (voir 2.1c). Dans ces deux derniers cas, la référence correspondant à l'image de test est en réalité constituée de plusieurs images qu'il faut combiner. Plus généralement, le problème de la constitution d'une référence se pose dès qu'un ensemble de données de référence est disponible et que nous souhaitons détecter des changements dans une image de test par rapport à une ou plusieurs images de cet ensemble. Ce problème, qui se repose pour chaque nouvelle image de test à analyser, nécessite un traitement spécifique, qui doit être rapide afin de pouvoir traiter efficacement une vidéo de test contenant de nombreuses images.

La très large majorité des méthodes de détection de changements [2, 7, 8, 11, 25, 29, 48, 56, 84, 106, 111, 112], en particulier la plupart de celles décrites par Radke et al. [95], n'abordent tout simplement pas ce problème. En effet, ces méthodes travaillent sur des paires d'images et partent du principe que les deux images considérées observent la même scène. Or, dans la majorité des scénarios opérationnels de la reconnaissance aérienne ou satellitaire et de la vidéo-surveillance, un grand nombre de données sont acquises sur des zones géographiques différentes. De plus, il est rare que les acquisitions effectuées soient jetées immédiatement, vu leur intérêt potentiel et leur coût important. Par conséquent, à chaque acquisition d'une nouvelle donnée d'observation, l'ensemble de celles déjà acquises dans la même région est disponible pour servir de référence, mais il est nécessaire de pouvoir y accéder efficacement. Ainsi, la constitution d'une référence est une étape cruciale pour les scénarios opérationnels de détection de changements. La figure 2.2 présente une taxonomie des différentes catégories d'approches abordant ce problème.

Synchronisation temporelle Dans la littérature, un certain nombre d'approches [81, 92, 103] s'intéressent à la comparaison de deux vidéos acquises selon des trajectoires similaires, par exemple pour la détection d'Engins Explosifs Improvisés (EEI) le long d'itinéraires fréquemment utilisés par des convois. Dans ce contexte, l'approche la plus simple consiste à synchro-

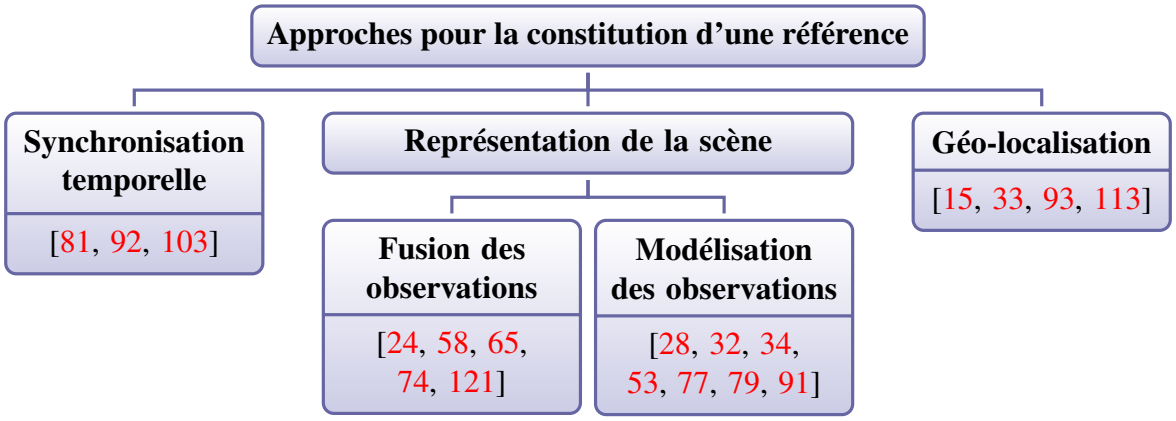


FIGURE 2.2 – *Cet arbre présente une taxonomie des différentes approches utilisées pour aborder le problème de la constitution d'une référence associée à une image de test donnée.*

niser temporellement ces vidéos, c'est-à-dire à associer, à chaque image de la vidéo de test, l'image de la vidéo de référence la plus proche le long de la trajectoire commune.

Par exemple, Primdahl et al. [92] proposent une méthode de détection de changements entre deux vidéos acquises par une caméra installée sur un véhicule terrestre. À l'aide de nombreux capteurs et d'une version non-linéaire du filtre de Kalman, une localisation précise est associée à chaque image des vidéos. Par la suite, une image de la vidéo de test est comparée avec l'image de la vidéo de référence la plus proche en termes de distance euclidienne entre centres optiques.

Lorsqu'il est impossible d'utiliser une batterie de capteurs pour l'estimation de la localisation, ou que la précision obtenue est insuffisante, Stennett et Evans [103] proposent une méthode pour effectuer une synchronisation temporelle grâce à l'analyse du contenu des images. L'idée, qui est régulièrement appliquée à d'autres problèmes, consiste à extraire des points d'intérêt de Harris, qui ciblent les coins apparaissant dans les images, à plusieurs résolutions et à les apparier d'une vidéo à l'autre grâce à un descripteur SIFT [69]. Cet appariement, qui contient initialement de nombreuses erreurs, est par la suite amélioré en exploitant la contrainte géométrique, mise en œuvre par estimation de la matrice fondamentale grâce à un algorithme de *RAndom SAmple Consensus* (RANSAC) [45].

L'approche par synchronisation temporelle est relativement naturelle dans le cas où les vidéos sont acquises selon des trajectoires similaires. En revanche, une première limite de cette approche est le problème de l'indexation, qui n'est abordé par aucune des deux méthodes décrites ci-dessus. En effet, si les vidéos sont longues il devient inefficace de parcourir systématiquement toutes les images de référence pour trouver la plus proche de l'image de test. Un système d'indexation est donc nécessaire pour permettre un accès efficace aux images potentiellement utiles. Cependant, le problème majeur de l'approche par synchronisation temporelle est qu'elle n'exploite pas le fort taux de recouvrement des images de référence. En effet, même dans le cas où les contenus de plusieurs images de référence correspondent à celui de l'image de test, une seule image bien précise est utilisée pour la comparaison et cette image peut être mal adaptée. Par exemple, il est possible que les conditions d'illumination soient très différentes du fait d'un éclat de lumière, ou qu'un piéton passe dans le champ de la caméra en cachant une partie de la scène. Dans de tels cas, le fait de ne comparer l'image de test qu'à une unique image de référence est une limitation importante qui génère de mauvaises performances en détection de changements.

Géo-localisation Lorsque les vidéos ou séquences d'images sont constituées de points de vue arbitraires, une approche très répandue [15, 33, 93, 113] consiste à géo-localiser les images, c'est-à-dire à les localiser précisément à la surface de la Terre. Une fois localisées dans un système de coordonnées commun, il est ensuite plus facile de les comparer. De nos jours, de plus en

plus de plates-formes d'observation (satellites, avions, drones militaires et même micro-drones) associent à leurs acquisitions les informations de localisation GPS et dans certains cas les paramètres d'acquisition (facteur de zoom et paramètres de calibration). Ces données sont rarement assez précises pour permettre la comparaison du contenu des données d'observation, mais elles peuvent néanmoins servir à initialiser des algorithmes de géo-localisation, qui permettent une estimation plus précise des paramètres d'acquisition.

Par exemple, Wiles et al. [113] présentent une méthode de géo-localisation de vidéos aériennes, utilisant une ortho-image satellitaire et un modèle numérique d'élévation (MNE). Grâce à l'estimation initiale des paramètres d'acquisition, l'ortho-image et le MNE sont projetés, par des techniques de rendu 3D, dans l'image courante de la vidéo pour constituer une référence géo-localisée. Un certain nombre de pré-traitements sont ensuite appliqués à cette référence et à l'image courante de la vidéo. Enfin la géo-localisation de l'image courante est affinée itérativement en alternant appariement local avec la référence géo-localisée et ajustement de faisceaux (*bundle adjustment* dans la littérature) grâce aux contraintes fournies par les images voisines.

En se contentant d'un MNE obtenu par relevés laser (LIDAR), Pritt et LaTourette [93] montrent qu'il est possible de géo-localiser les images d'une vidéo aérienne. Les conditions d'illumination observées dans la vidéo, supposées connues, sont utilisées pour éclairer le MNE et prédire l'image observée selon le point de vue de l'image courante. Les paramètres d'acquisition de la première image de la vidéo sont supposés connus grossièrement ou obtenus par calibration manuelle. Pour les suivantes, un recalage entre les images successives est effectué pour initialiser les paramètres d'acquisition. Par la suite, un recalage itératif est effectué entre l'image prédite par le MNE et l'image courante de la vidéo, permettant d'affiner progressivement l'estimation des paramètres d'acquisition.

Crispell et al. [33] proposent une approche similaire exploitant un modèle 3D avec apparences, mais utilisent le formalisme de l'asservissement visuel (*visual servoing* dans la littérature). L'idée, assez proche de la précédente, consiste à utiliser l'estimation courante des paramètres d'acquisition pour générer un rendu du modèle 3D. L'image courante de la vidéo et ce rendu sont ensuite recalés par transformation affine. Une matrice jacobienne, exprimée en considérant l'algèbre de Lie et les équations de projection, est utilisée pour convertir ce recalage affine en un déplacement 3D de la caméra associée à l'image courante. La géo-localisation de l'image courante est obtenue en itérant ces trois étapes, et un filtre de Kalman est ensuite implémenté pour modéliser le déplacement de caméra entre l'acquisition des images successives de la vidéo considérée. Notons que cette méthode employée suppose que les paramètres de calibration sont connus pour la première image et sont constants dans le reste de la vidéo. Elle n'est de plus est pas facilement extensible pour permettre leur estimation.

Dans le cadre de cette thèse, une technique analogue exploitant l'asservissement visuel a été développée et publiée [15]. Cette technique utilise un mécanisme unique pour l'estimation précise des paramètres d'acquisition et pour prédire les paramètres d'acquisition de l'image suivante à partir de ceux estimés pour l'image courante. Elle ne nécessite donc pas l'utilisation d'un filtre de Kalman. Depuis la publication de cette technique, un cadre plus général a été développé, qui permet d'inclure l'estimation des paramètres de calibration. Par ailleurs, une seconde technique a également été développée [18] pour interpoler les paramètres d'acquisition dans une séquence d'images, en connaissant ceux associés à quelques images de la séquence. Ces deux techniques seront présentées plus en détail au chapitre 3.

Une fois les images géo-localisées, elles peuvent être indexées dans une base de données, par exemple utilisant une structure arborescente telle que le R-Tree [43]. Cependant, dans le cadre de la détection de changements dans des vidéos, cette approche a ses limites. En effet, la fréquence généralement importante de l'acquisition des images d'une vidéo (e.g. 25Hz) génère beaucoup de recouvrement dans les images à indexer. Par conséquent, lors de la recherche d'une référence pour une image de test donnée, un grand nombre d'images de référence sont obtenues. Or, pour des raisons évidentes de temps de calcul, il est impossible de comparer chacune de

ces images de référence avec l'image de test. Une première solution consiste à n'en choisir qu'une seule, mais ceci débouche alors sur la limite, évoquée plus haut dans le contexte de la synchronisation temporelle, concernant la non exploitation du recouvrement des images de référence. La seconde possibilité consiste à fusionner tout ou partie des résultats de recherche, ce qui représente une charge de calcul non négligeable et rend le système lent et inefficace lors du traitement d'une vidéo de test contenant de nombreuses images.

Représentation synthétique des observations de référence En réalité, le fort taux de recouvrement des images de référence est connu avant même la donnée d'une quelconque image de test. Par conséquent, il semble plus adapté de chercher à résumer les observations de référence de manière préalable à toute considération sur les images de test. Cette approche, qui consiste finalement à calculer une représentation synthétique de la scène de référence, est extrêmement flexible et permet de décharger une partie importante des calculs en pré-traitement, et donc de rendre le système plus rapide. Cette approche permet également de s'assurer que la référence utilisée pour la comparaison avec l'image de test est la plus adaptée possible, en termes de recouvrement et de contenu. Les représentations utilisées [24, 28, 32, 34, 53, 58, 65, 74, 79, 91] sont relativement variées, et leur choix répond également à certains objectifs concernant d'autres problématiques indépendantes, que nous mentionnerons dans les sections suivantes.

Fusion des observations de référence Parmi les approches exploitant une représentation synthétique de la scène, Irani et al. [58] discutent des nombreuses manières de représenter une vidéo grâce à une mosaïque et mentionnent différents mécanismes de fusion des observations. Selon les applications, il peut par exemple être souhaitable de conserver dans la mosaïque l'information disponible la plus récente, l'information représentant le fond statique de la scène ou au contraire le premier plan avec les objets mobiles.

Cet article évoque également le fait que l'utilisation d'une mosaïque dont la résolution est fixée a priori peut causer la perte de certains détails disponibles dans la vidéo d'origine. Un certain nombre de méthodes ont donc été proposées pour construire une mosaïque dont la résolution est adaptative. Par exemple, Masood et Kanwal [74] utilisent le facteur de zoom de la transformation affine entre les coordonnées images et les coordonnées de référence, afin de déterminer la résolution globale de l'image courante. Ils construisent alors plusieurs mosaïques à des résolutions différentes, afin de conserver le maximum d'information. Plus originaux, Lee et Kim [65] introduisent la notion de carte de résolutions, qui indique, en chaque point d'un système de coordonnées de référence, les valeurs maximales, dans une séquence d'images, d'un descripteur de résolution. Ce dernier est défini formellement comme la dérivée (ou jacobienne, dans le cas 2D) de la fonction de transformation, ici homographique, des coordonnées images vers les coordonnées de référence. Pour construire la mosaïque, la carte de résolutions est ensuite segmentée afin de définir plusieurs couches, dans lesquelles les observations sont fusionnées à une résolution optimale.

Buchanan [24] effectue une reconstruction 3D des vidéos de référence et de test, à l'aide d'une méthode de *Structure From Motion*, et estime les deux trajectoires d'acquisition des vidéos dans le même système de coordonnées. Le modèle de référence est représenté par un maillage texturé à l'aide de fragments des images de la vidéo de référence. Par la suite, le point de vue de chaque image de la vidéo de test ayant été estimé, il est possible de générer une image de référence synthétique mais correctement recalée, par rendu du modèle selon le point de vue de l'image de test.

Une limite importante de ces approches est que l'image de référence utilisée pour la comparaison avec l'image de test est synthétique. Dans certains cas, elle est constituée de fragments des images de la vidéo de référence. Dans de tels, si cette vidéo contient des objets mobiles, ou plus généralement des variations d'apparence, l'image de référence synthétique risque de contenir des aberrations pouvant fausser la détection de changements. Une solution plus sûre,

employée dans certaines approches, peut consister à mélanger les intensités des observations de référence, par exemple en calculant leur moyenne ou leur médiane. Cependant, cela reste un choix arbitraire et a priori de fusion des intensités observées en une intensité unique, qui n'est pas forcément représentative de l'ensemble initial.

Modélisation des observations de référence Pour garantir cette représentativité, l'approche idéale consiste alors à construire un modèle de la scène dans lequel sont stockés des modèles probabilistes d'apparence, plutôt que des intensités fusionnées de manière arbitraire. Dans ce cas, aucune image de référence n'est retournée : les modèles d'apparence servent directement de référence et sont utilisés pour la comparaison avec l'image de test. Notons que les mécanismes spécifiques de modélisation des observations de référence [28, 32, 34, 53, 79, 91] relèvent du problème de la comparaison de données, et seront donc détaillés dans la section correspondante. Il est cependant intéressant d'étudier les diverses manières selon lesquelles ces modèles sont organisés.

Cette idée de modélisation des observations est par exemple utilisée par Mittal et Huttenlocher [79] et Hayman et Eklundh [53], qui proposent par exemple deux techniques relativement similaires de soustraction de fond visant à modéliser, à l'aide d'une mosaïque, la scène observée par une caméra mobile. Pour apparier une nouvelle observation avec le modèle de la scène, un recalage est estimé entre la nouvelle image et la mosaïque constituée des observations moyennes des images traitées précédemment. Mittal et Huttenlocher [79] commencent par un recalage grossier à l'aide d'une transformation affine, puis optimisent ce recalage grâce à l'algorithme de Levenberg-Marquardt. Hayman et Eklundh [53] supposent que les paramètres de calibration sont connus et estiment le recalage résiduel de manière robuste par une variante de l'algorithme de RANSAC. Dans les deux méthodes, les nouvelles observations sont par la suite intégrées dans les modèles d'apparence contenus dans les cellules de la mosaïque. Cho et Kim [28] proposent une autre méthode combinant la modélisation des apparences avec la technique de mosaïque multi-résolution introduite par Lee et Kim [65].

Pollard et Mundy [91] introduisent un modèle volumétrique de la scène, où chaque élément de volume, encore appelé voxel, contient une probabilité d'occultation et un modèle d'apparence. Un algorithme de lancer de rayons (*ray casting* dans la littérature) permet de mettre en correspondance les pixels de l'image de test et les voxels du modèle, permettant alors de détecter les changements en comparant les observations aux modèles d'apparence. Crispell et al. [34] proposent une extension de cette méthode en intégrant un mécanisme de résolution adaptative : les voxels ne sont plus organisés selon une grille tri-dimensionnelle, mais dans une structure arborescente 3D appelée Octree. Cette organisation arborescente permet des gains significatifs de place mémoire, et est exploitée pour modéliser précisément les objets dans la scène tout en représentant grossièrement les espaces vides. Il est à noter que ces deux dernières approches nécessitent la géo-localisation des images de manière à pouvoir mettre précisément en correspondance leurs observations avec les modèles d'apparence.

2.3 Gestion des sources de variabilité non pertinentes

Nous avons vu dans le chapitre 1 qu'en général, deux vues d'une même scène, acquises à des instants et depuis des points de vue différents, contiennent de nombreuses variations non pertinentes (e.g. effets géométriques lors du changement de point de vue, variations d'illumination, etc). Si aucune précaution n'est prise, ces variations peuvent générer de nombreuses fausses alarmes durant la phase de comparaison des données. Or, dans un système d'assistance à l'analyse, il est crucial que le nombre d'alertes injustifiées soit maintenu au minimum. En effet, dans un contexte opérationnel, un outil qui fait perdre du temps est un outil contre-productif, et l'utilisateur risque alors rapidement de s'en passer complètement.

Une approche relativement générale pour aborder les variations d'apparence consiste à exploiter des modèles d'apparence estimés à partir d'un ensemble d'observations. Cela permet en particulier de capturer la dynamique intrinsèque de la scène observée, par exemple le mouvement des feuilles d'arbres dû au vent, les vagues sur l'eau et ainsi de suite. Cette approche est adaptée à la modélisation des variations d'apparence périodiques et fréquentes, mais est relativement inefficace pour le traitement des variations aléatoires (e.g. ombres de nuages se déplaçant, effets de parallaxe, etc) ou de celles dont la période est largement plus grande que la durée de la vidéo (e.g. changement de direction des ombres portées).

Par conséquent, de plus en plus de travaux abordent explicitement certains types de variations au sein de leurs algorithmes de détection de changements, afin d'améliorer leur robustesse. Cela peut cependant avoir comme effet secondaire d'augmenter le nombre de changements significatifs non-détectés. En effet, il peut arriver que de tels changements aient, du point de vue de l'algorithme de traitement utilisé, un comportement similaire aux variations non pertinentes devant être ignorées, causant ainsi leur non détection au même titre que ces dernières. Il y a donc généralement un compromis à trouver entre la réduction des fausses alarmes et l'augmentation des non-détections.

Les sections 2.3.1 et 2.3.2 traitent des techniques utilisées pour rendre la détection de changements robuste aux variations les plus fréquemment abordées, c'est-à-dire respectivement les effets géométriques et les variations d'illumination. La section 2.3.3 traite des techniques, plus rares, abordant d'autres types de variations.

2.3.1 Effets géométriques

Afin de détecter les changements survenus sur une scène, il est crucial de parvenir à mettre en correspondance les observations disponibles, c'est-à-dire à associer, pour chaque pixel d'une première image, le pixel correspondant au même point physique dans la seconde image. En effet, cette mise en correspondance est nécessaire, afin de pouvoir comparer le contenus des pixels de chaque image. Or, les effets géométriques rendent difficile cette étape de mise en correspondance, qui nécessite alors un traitement spécifique.

Plus précisément, les effets géométriques sont dûs, d'une part aux changements de point de vue entre deux images, et d'autre part à la présence de relief dans la scène observée. Ils se manifestent sous la forme d'occultations, c'est-à-dire de parties de la scène visibles dans l'une des images mais cachées dans l'autre. Ils peuvent également se manifester sous la forme d'effets de parallaxe, qui désignent la différence de déplacement apparent de deux objets, lorsque ceux-ci sont situés à des distances différentes d'une caméra qui se déplace. Ces effets de parallaxe sont proportionnels au déplacement de la caméra, mais également au rapport de la différence de profondeur des objets par la distance entre la caméra et ces objets. Pour résumer, ces effets géométriques peuvent non seulement modifier l'ordre d'apparition des objets dans deux vues d'une même scène, mais également rendre visibles, dans l'une des vues, des parties de la scène introuvables dans la seconde. Par conséquent, ils complexifient grandement la mise en correspondance des observations, comme l'illustre la figure 2.3 à l'aide d'une paire d'images aériennes.

Lors de l'exploitation d'images acquises par une plate-forme mobile (e.g. véhicule terrestre, aérien, satellite, etc), il est rare que deux images données soient issues exactement du même point de vue. Par conséquent, la gestion des effets géométriques est un problème extrêmement fréquent, que de nombreux travaux ont eu à traiter. Les approches correspondantes peuvent être classées en trois catégories, présentées visuellement dans la taxonomie de la figure 2.4 : celles utilisant un recalage préalable des images, celles exploitant la géométrie épipolaire et celles exploitant une modélisation tri-dimensionnelle.

(a) (b)

FIGURE 2.3 – *Cette figure illustre la manifestation des effets géométriques dans une paire d'images aériennes, acquises sur la même scène selon deux points de vue différents. Par exemple, les objets occultés par le château d'eau dans l'une des deux images ne sont visibles que dans l'autre image. De plus, l'ordre d'apparition de certains objets est modifié, comme c'est le cas pour les poteaux électriques (flèches noires) situés à gauche (a) ou à droite (b) du château d'eau. Copyright © 2010 - 2012 Cassidian - All rights reserved.*

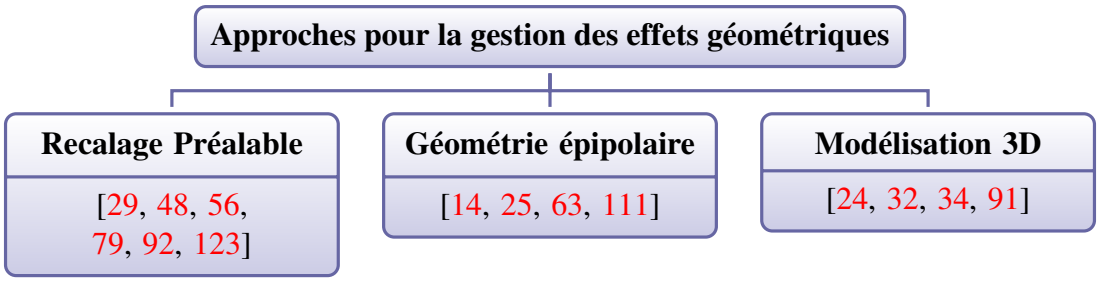


FIGURE 2.4 – *Cet arbre présente une taxonomie des différentes approches utilisées pour aborder le problème de la gestion des effets géométriques.*

Recalage préalable Une large majorité d'articles [29, 48, 56, 79, 92, 123] en détection de changements abordent le problème des effets géométriques en supposant qu'un recalage préalable des images a été effectué, et en se concentrant sur leur comparaison. Une telle hypothèse, posée par exemple par Clifton [29], est réaliste dans le contexte de l'imagerie satellitaire, car le point de vue et le relief générant les effets géométriques sont connus très précisément, par rapport à la distance d'observation. En revanche, dans le contexte de l'imagerie aérienne ou terrestre, le relief est beaucoup plus complexe et les variations de points de vue sont nettement plus importantes. Il devient donc nécessaire d'aborder explicitement le problème de la mise en correspondance.

Dans le cadre de vidéos acquises selon des trajectoires proches, par exemple par des caméras montées sur véhicules terrestres, l'approche par recalage permet d'obtenir de bons résultats. En effet, pour toute image de test il est possible de trouver une image de référence acquise selon un point de vue très proche. Par conséquent, un recalage relativement simple entre les deux images peut suffire à atténuer efficacement les effets géométriques. Ainsi, Primdahl et al. [92] utilisent un modèle planaire de la route, et commencent par convertir les deux images en ortho-images, à l'aide d'une homographie calculée en pré-traitement, la caméra étant fixée de manière rigide au véhicule. Un recalage grossier, combinant une translation et une rotation, est ensuite calculé à partir des méta-données (coordonnées GPS et focales des caméras). Finalement, une translation précise de recalage résiduel est estimée à l'aide de la transformée de Fourier.

Dans le cas de vidéos aériennes acquises selon des trajectoires arbitraires, il est plus rare que les images soient acquises selon des points de vue proches. En revanche, si la région observée contient peu de relief, ou que la distance de la caméra par rapport à la scène est grande

devant la taille des objets observés, les effets géométriques restent minimes et peuvent être atténués simplement. Mittal et Huttenlocher [79] proposent ainsi d'effectuer un recalage robuste de l'image courante de la vidéo de test avec une mosaïque de référence constituée des observations moyennes, afin d'éliminer les objets mobiles. Dans un premier temps, un recalage grossier est effectué, en estimant par moindres carrés une transformation affine grâce à un filtre KLT. Ce recalage grossier sert d'initialisation à un recalage projectif fin, estimé itérativement par l'algorithme de Levenberg-Marquardt, initialisation permettant d'accélérer significativement la convergence.

Plus généralement, Zitova et Flusser [123] présentent une revue de l'état de l'art des méthodes de recalage d'images. Ils identifient deux catégories d'approches, les méthodes par surface (*area-based* dans la littérature) et celles par primitives caractéristiques (*feature-based* dans la littérature), et quatre sous-problèmes orthogonaux. Ces problèmes sont l'extraction de caractéristiques, l'appariement de ces caractéristiques, l'estimation d'un modèle de transformation et enfin le ré-échantillonnage d'image. Il ressort de cette étude que, dans le cas général où les effets géométriques sont importants, les modèles de transformation paramétrique globale (e.g. transformation affine ou projective) sont peu adaptés. Il devient alors nécessaire d'utiliser des techniques d'appariement local, comme par exemple un recalage élastique [3, 30] ou un flot optique [6, 39].

Cependant, bien que des méthodes relativement sophistiquées existent pour le recalage générique d'images ayant subi des déformations complexes et arbitraires, ces méthodes sont généralement très coûteuses. Or, les déformations rencontrées dans le cadre de la détection de changements sont généralement bien spécifiques puisqu'elles proviennent d'effets géométriques relativement bien modélisés. Par conséquent, des approches dédiées ont été étudiées pour résoudre ce problème, en exploitant le comportement particulier des effets géométriques.

Exploitation de la géométrie épipolaire La géométrie épipolaire est un cadre théorique modélisant, de manière très précise, le comportement bi-dimensionnel des effets géométriques, c'est-à-dire leurs conséquences au niveau des intensités des images. Ce cadre théorique est introduit et décrit en détail dans les livres traitant de vision artificielle [41, 52]. L'exploitation des équations de la géométrie épipolaire dans une méthode de détection de changements [14, 25, 63, 111] permet donc d'aborder de manière appropriée les difficultés générées par les effets géométriques.

Par exemple, Kumar et al. [63] montrent que le recalage de deux vues d'une même scène, acquises selon des points de vue différents, comporte deux étapes. La première étape consiste à recaler les deux images par rapport à une surface paramétrique arbitraire, par exemple par rapport au plan dominant du sol, qui mène à un recalage à base d'homographie. Par la suite, il est démontré que les déviations résiduelles dues à la parallaxe peuvent être modélisées par un champ de vecteurs épipolaires. Ils présentent deux algorithmes, permettant la résolution séquentielle ou simultanée de ces deux étapes. L'approche séquentielle utilise une première exécution de l'algorithme itératif de Levenberg-Marquardt pour l'estimation du recalage par rapport au plan du sol, puis une seconde pour estimer le champ de vecteurs épipolaires. Cette approche séquentielle peut échouer dans le cas où le sol de la scène observée n'est pas plan (par exemple du fait d'un relief important). Dans de tels cas, ils proposent d'utiliser l'approche simultanée, qui consiste à estimer, en une seule exécution de l'algorithme de Levenberg-Marquardt, un plan du sol moyen et le champ de vecteurs épipolaires.

La méthode proposée par Watanabe et Miyajima [111] exploite également la géométrie épipolaire, et effectue la mise en correspondance de toits de bâtiments dans des images aériennes, en recherchant la forme de ces toits le long des droites épipolaires. Ces droites épipolaires sont calculées grâce à la matrice fondamentale, qui peut être estimée par appariement de points caractéristiques [52]. Cette méthode permet de gérer correctement les effets de parallaxe, mais en revanche elle n'aborde pas les éventuelles occultations.

[25] propose une approche de recalage d'images basée sur le Dynamic Time Warping, algorithme de programmation dynamique visant à la mise en correspondance optimale entre deux images. Cet algorithme, relativement coûteux, considère les intensités observées le long des droites épipolaires, et permet de mettre en correspondance les intensités issues de deux vues de la même scène malgré les occultations et les effets de parallaxe.

Les travaux réalisés au début de cette thèse ont mené au développement d'une technique de détection de changements [14], basée sur la géométrie épipolaire. Cette technique, qui intègre la contrainte épipolaire à un algorithme rapide de flot optique, sera présentée à la section 4.1. Cependant, il est rapidement apparu que cette approche, qui ne peut être appliquée qu'entre paires d'images, était inefficace dans le cadre de la détection de changements entre vidéos. Ceci a donc justifié l'adoption d'une approche de modélisation tri-dimensionnelle.

Modélisation tri-dimensionnelle Une autre approche pour la gestion des effets géométriques consiste à simuler leurs conséquences à l'aide de techniques, tels que l'algorithme de lancer de rayons, exploitant une modélisation tri-dimensionnelle. En effet, si la scène est modélisée en trois dimensions, il est possible de générer des images synthétiques par lancer de rayons en simulant les effets d'occultation et de parallaxe. Un certain nombre d'approches [24, 32, 34, 91] exploitent cette idée pour générer, à partir de modèles de référence, des images synthétiques acquises selon le même point de vue que l'image de test considérée, réduisant donc sensiblement l'impact des effets géométriques.

Par exemple, Buchanan [24] calcule, à l'aide de l'algorithme de *Structure From Motion* [52] deux modèles tri-dimensionnels de la scène observée à partir d'une vidéo de référence et d'une vidéo de test, ainsi que les trajectoires d'acquisition de ces vidéos. Ces deux modèles sont ensuite recalés, notamment afin d'obtenir un système de coordonnées compatibles entre le modèle de référence et la trajectoire de la vidéo de test. Par des techniques de rendu, il est alors possible de générer une image synthétique du modèle de référence alignée avec n'importe quelle image de la vidéo de test. Cela permet donc de comparer les contenus des deux vidéos en faisant abstraction des effets géométriques. En revanche, une limite de cette approche est qu'elle ne permet pas l'exploitation incrémentale d'une vidéo, c'est-à-dire au fur et à mesure qu'elle est acquise. D'autre part, il est généralement difficile de convertir un modèle tri-dimensionnel issu de l'algorithme de *Structure From Motion* en un modèle dense exploitable pour le rendu d'images.

Plus originaux, Pollard et Mundy [91] proposent d'estimer, à partir des observations de référence, une modélisation volumétrique de la scène observée, où chaque voxel contient une probabilité d'occultation et un modèle d'apparence. En supposant que le point de vue d'acquisition de l'image de test est connu, un algorithme de lancer de rayons permet de calculer une image synthétique du modèle de référence selon le même point de vue. Il est alors possible de comparer cette image avec l'image de test, afin de détecter les changements en minimisant l'impact des effets géométriques. Crispell et al. [33] étendent ces travaux, en proposant une méthode permettant l'estimation du point de vue de l'image de test à partir de son contenu et de la connaissance d'un modèle 3D de la scène observée. Cette extension permet donc de traiter une vidéo de test de manière incrémentale.

Enfin, une approche basée sur la modélisation tri-dimensionnelle d'apparence a été proposée [18] dans le cadre de cette thèse. Partant de la remarque selon laquelle, dans le cadre de l'imagerie aérienne, il est plus efficace de considérer la scène comme une surface que comme un volume, nous avons proposé une représentation de la scène basée sur une carte d'élévation. La carte d'élévation est une forme contrainte de modèle 3D, qui offre une bonne capacité de généralisation, par rapport aux points de vue non explorés lors de sa construction. Cette représentation sera présentée plus en détails au chapitre 4.

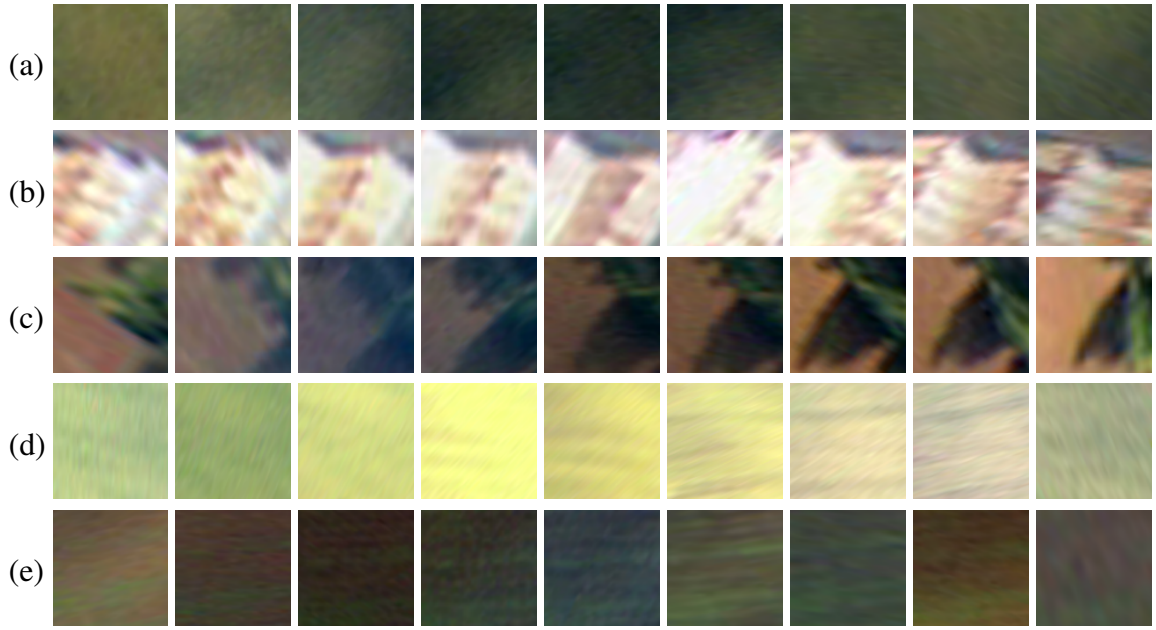


FIGURE 2.5 – *Cette figure illustre la manifestation des effets de l'illumination, en prenant l'exemple de cinq zones (a)-(e) observées à plusieurs instants différents. Ces manifestations incluent des effets divers tels que l'illumination directe ou voilée, ce qui atténue les frontières des ombres et modifie la saturation des couleurs, la projection d'ombres mobiles, qui modifie l'apparence aléatoirement, les réflexions spéculaires plus ou moins intenses, qui dépendent du point de vue, etc. Cette illustration montre la grande variabilité générée par ces effets de l'illumination.*

2.3.2 Variations d'illumination

Dans le cadre de la détection de changements, les variations d'illumination représentent une source classique de fausses alarmes. Ce problème est en particulier incontournable dans les images acquises en extérieur, et certains travaux [95] ont étudié les techniques permettant de réduire la sensibilité des algorithmes de détection de changements à ces variations d'illumination. Leurs manifestations dans les images peuvent prendre plusieurs formes : éclairage intense ou voilé de la scène selon que le temps est dégagé ou couvert, ombres plus ou moins dures selon l'intensité de l'éclairage, ombres orientées différemment selon l'heure du jour, ombres mobiles dues par exemple au déplacement de nuages ou inversement, zones éclairées du fait d'un trou dans la couverture nuageuse, et ainsi de suite. De plus, les variations d'illumination sont le plus souvent associées à de légères variations de couleur, pouvant générer par la suite des problèmes de comparaison d'image. Typiquement, la lumière du Soleil tend à apporter une légère touche de jaune aux objets qu'elle éclaire, et inversement les objets dans l'ombre apparaissent légèrement bleus. La figure 2.5 illustre les manifestations des variations d'illumination, en présentant plusieurs ensembles de vignettes correspondant à quelques emplacements d'une scène donnée vus sous différentes conditions d'illumination.

Nous avons vu précédemment, au sujet des effets géométriques, qu'il existait des modèles permettant la description précise de leurs conséquences. De la même manière, pour les variations d'illumination, de nombreux modèles physiques permettent de décrire les mécanismes mis en œuvre. Les approches correspondantes peuvent être classées en trois catégories : les approches génératives qui simulent les conditions d'illumination, les approches cherchant à détecter les effets de l'illumination pour ignorer les fausses alarmes correspondantes, et les approches basant la détection de changements sur une mesure invariante par rapport aux variations d'illumination. La figure 2.6 présente visuellement la taxonomie correspondant à ces catégories.

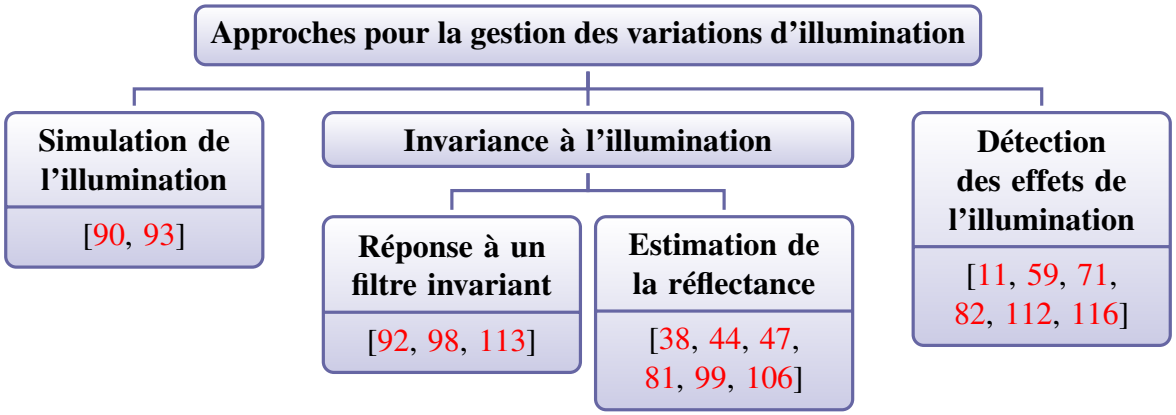


FIGURE 2.6 – *Cet arbre présente une taxonomie des différentes approches utilisées pour aborder le problème de la gestion des variations d'illumination.*

Simulation des conditions d'illumination Le problème posé par les variations d'illumination apparaît au niveau de la comparaison entre deux observations. Par conséquent, une première idée consiste à minimiser les différences de contenu entre les deux observations, en simulant dans l'une les conditions d'illumination présentes dans l'autre. Par exemple, en déterminant précisément les conditions d'illumination dans l'image de test, il peut être possible de les reporter dans l'image de référence pour permettre une comparaison fiable des images.

Suivant cette idée, Pollard [90] propose une extension de la méthode [91] se concentrant sur la gestion de différentes orientations de l'illumination. La technique suppose que la direction de l'illumination associée à chaque image est connue a priori, et quantifiée en un nombre non précisé de représentants. Il est alors possible de calculer un jeu de modèles d'apparence dédié à chaque pas de direction de l'illumination, en utilisant uniquement les images de référence ayant une direction de l'illumination proche. Par la suite, durant l'étape de comparaison, les changements sont détectés en utilisant le jeu de modèles d'apparence correspondant à la direction de l'illumination dans l'image de test considérée. Cependant, une limite évidente de cette approche est qu'elle nécessite beaucoup d'observations de référence, acquises dans des conditions d'illumination variées. L'auteur montre qu'il est possible de nuancer ce problème en utilisant une interpolation des modèles, dans le cas où quelques directions de l'illumination ne seraient pas représentées parmi les observations de référence.

Pritt et LaTourette [93] proposent une autre approche nécessitant également la connaissance de la direction de l'illumination. Leur technique exploite un modèle tri-dimensionnel de la scène et suppose connus les paramètres d'acquisition et la direction de l'illumination dans l'image de test. Il est alors possible, par des techniques de rendu, d'éclairer le modèle selon la même direction de l'illumination et de générer une image synthétique de la scène, acquise selon le même point de vue que l'image de test et sous les mêmes conditions d'illumination.

Cette approche de simulation des conditions d'illumination est assez adaptée lorsqu'un modèle tri-dimensionnel de la scène est disponible, car la simulation des effets de l'illumination nécessite la connaissance du relief. Cependant, deux limites communes aux approches de cette catégorie peuvent être identifiées. D'abord, la gamme des effets de l'illumination est extrêmement vaste et ne se limite pas à différentes directions de l'illumination. Or, l'utilisation d'un modèle global, permettant de simuler de manière réaliste tous les effets rencontrés dans les données d'observation aérienne, semble illusoire du fait de sa complexité. D'autre part, extraire automatiquement les conditions d'illumination de l'image de test est un problème délicat et mal posé, qui nécessite donc une connaissance a priori ou une intervention humaine.

Détection des effets de l'illumination Une autre catégorie d'approches [11, 59, 71, 82, 112, 116] vise à détecter, de manière non supervisée, les effets de l'illumination afin de les exclure

des résultats de détection de changements. Certaines méthodes se concentrent sur un type spécifique d'effets de l'illumination, comme les ombres ou les reflets, tandis que d'autres, plus rares, permettent d'en détecter plusieurs simultanément.

Ainsi, la méthode présentée par Black et al. [11] permet de détecter plusieurs types d'effets de l'illumination, à condition de disposer d'un modèle paramétrique a priori. Cette technique explique les variations d'intensités dans les images par une combinaison de ces modèles paramétriques, et maximise, par un algorithme d'Espérance-Maximisation (EM, pour *Expectation-Maximization* dans la littérature), la vraisemblance du modèle total, en estimant alternativement les coefficients de pondération et les paramètres de chaque modèle.

Parmi les approches se concentrant sur la détection des ombres, Watanabe et al. [112] proposent une technique permettant de distinguer les changements significatifs de ceux dus à un changement d'orientation de l'ombre d'un bâtiment. Pour cela, ils utilisent un modèle d'illumination basé sur la diffusion Lambertienne [1] et une source ponctuelle de lumière située à l'infini. Il est alors démontré, en supposant connue la direction de l'illumination, que le ratio des intensités des observations de référence et de test prend des valeurs spécifiques selon que le pixel considéré a subi un changement d'ombre et de structure, un changement d'ombre seulement, de structure seulement, ou aucun changement. En raisonnant sur la valeur de ce ratio, ils parviennent donc à exclure les fausses alarmes dues aux ombres des résultats de détection de changements.

Kaewtrakulpong et Bowden [59] développent une technique de détection des ombres applicable dans le cadre de la soustraction de fond. Cette technique, qui vise la rapidité, considère deux mesures, l'une caractérisant la distorsion de luminosité et l'autre la distorsion de couleur. Lors de la détection de changements, ces deux mesures sont comparées à des seuils pour décider s'il s'agit d'ombres ou de changements significatifs.

Nadimi et Bhanu [82] introduisent une méthode permettant la détection des ombres dans une séquence vidéo. Pour cela, ils mettent en œuvre une série de tests dédiés, visant à ne conserver que les pixels correspondant aux ombres et à éliminer les autres. La première étape consiste à analyser la luminosité des observations dans l'image de test, et à éliminer celles étant plus lumineuses que les observations de référence correspondantes et ne pouvant donc pas être dues aux ombres. Ensuite, les zones dans l'ombre ayant tendance à apparaître bleues, les pixels dont la teneur en bleu est trop faible sont éliminés. Les pixels restant sont alors regroupés en régions aux propriétés cohérentes, à l'aide d'un nouveau critère appelé rapport d'albédo. Enfin, la couleur dominante de chaque région est comparée à une couleur de référence dépendant du matériau attendu à cet endroit, et identifie la région à une ombre si ces couleurs sont compatibles. Cette approche nécessite donc non seulement une phase d'apprentissage supervisée afin de rassembler les couleurs de référence pour chaque matériau visible dans la scène, mais également une caractérisation fine des matériaux présents dans la scène, qui peuvent être difficilement accessible en pratique et en particulier dans le cas de l'observation aérienne.

Certaines approches s'intéressent également aux reflets de lumière. Par exemple, Mallick et al. [71] proposent une technique permettant d'éliminer les reflets de lumière dans des images ou des vidéos. Ils utilisent pour cela le cadre de la morphologie différentielle et décomposent les intensités observées par érosion afin d'estimer les composantes diffuse et spéculaire. Différents jeux d'équations aux dérivées partielles sont ainsi proposés pour répondre aux contraintes de différents scénarios : images avec ou sans texture, et vidéos.

La majorité des approches de cette catégorie se concentrent sur un type particulier d'effet de l'illumination, par exemple la détection des ombres ou des reflets. En revanche, dans le cas de plusieurs types de variations d'illumination, il est nécessaire d'adopter une approche similaire

1. Les objets qui suivent une loi de diffusion Lambertienne possèdent une apparence mate sans reflets, identique quelque soit le point de vue selon lequel ils sont observés. C'est par exemple le cas du béton ou de la terre, par opposition aux matériaux brillants ou réfléchissants tels que les peintures métallisées, les plastiques, etc.

à celle de Black et al. [11], qui est en pratique inutilisable dans le cadre de vidéos du fait de l'importante charge de calcul requise.

Invariance aux variations d'illumination Pour aborder les variations générales d'illumination, de manière efficace et non supervisée, la dernière catégorie d'approches propose d'effectuer la comparaison des données dans un espace invariant par changement d'illumination. En effet, s'il est possible de convertir les intensités des images de référence et de test, dans une représentation où les effets de l'illumination ont disparu, alors ces effets ne généreront aucune fausse alarme lors de la comparaison des images.

Réponse à un filtre invariant Une première possibilité consiste à transformer les images à l'aide d'un filtre invariant par changement d'illumination, tel que ceux faisant ressortir les contours des objets. En effet, les contours restent généralement bien discernables sous une large gamme de conditions d'illumination et leur position est très stable, permettant une comparaison fiable pour la détection de changements.

Ainsi, dans le but de représenter les images de manière invariante par changement d'illumination, Wiles et al. [113] utilisent la combinaison d'un filtre de dérivée seconde de gaussienne ainsi que sa transformée de Hilbert, pour détecter les arêtes selon quatre orientations. Dans la même optique, Rowe et Grewe [98] proposent d'utiliser l'algorithme de Canny pour la détection de contours. Le choix de cet algorithme est justifié par le fait qu'il permet d'obtenir des contours de longueur plus importante. Primdahl et al. [92] préfèrent employer le filtrage de Sobel, à cause des faibles performances de détection obtenues par l'algorithme de Canny dans le cas d'objets aux contours vagues. De plus, afin de réduire les temps de calcul, ils n'utilisent que la version verticale du filtre de Sobel, ce qui correspond à un a priori sur la manière dont la scène est observée.

Estimation de la réflectance Les modèles physiques décrivant les mécanismes de l'illumination font le plus souvent intervenir deux termes dans l'expression de l'intensité observée : l'illumination incidente et la réflectance de la scène. Ce terme de réflectance [2] décrit les propriétés intrinsèques de la scène observée, qui sont indépendantes de l'illumination incidente. Par conséquent, cette réflectance représente une mesure intéressante pour la comparaison d'image indépendamment des effets de l'illumination, et un certain nombre d'approches [38, 44, 47, 81, 99, 106] cherchent à l'estimer.

Par exemple, Toth et al. [106] proposent une méthode d'estimation de la réflectance de la scène à partir de l'intensité observée, égale au produit de la réflectance par la composante d'illumination. Cette méthode pose l'hypothèse que la fréquence spatiale associée à la composante d'illumination est faible devant celle de la composante de réflectance. Dans ce cas, en considérant le logarithme de l'intensité observée qui transforme les produits en sommes, il est alors possible de séparer la composante de réflectance du reste grâce à un filtre passe-haut. Cependant, l'hypothèse portant sur la différence de fréquence spatiale entre les deux composantes n'est pas valable dans le cas, courant, des ombres portées, puisque l'illumination peut varier très rapidement aux frontières de la zone d'ombre.

Elgammal et al. [38] proposent d'utiliser un espace approprié de représentation des couleurs, basé sur les coordonnées chromatiques classiques, afin d'atténuer simplement les effets de l'illumination. Ces coordonnées chromatiques permettent de représenter la couleur d'une observation en ignorant la notion de luminosité, ce qui est utile pour filtrer les effets de l'illumination mais a également pour effet de supprimer une partie de l'information liée à l'objet observé. Par exemple, dans cet espace de couleurs, il n'est plus possible de distinguer les objets

2. Cette notion de réflectance peut également être désignée par le terme albédo (*albedo* dans la littérature).

blancs des objets gris. Afin de ne pas perdre cette information, les auteurs utilisent une coordonnée supplémentaire pour représenter la luminosité, sur laquelle un seuil est utilisé pour limiter la sensibilité aux effets de l'illumination. Dans le cadre de cette thèse, plutôt que d'introduire un seuil arbitraire sur une grandeur peu fiable, nous avons proposé [18] de multiplier ces coordonnées chromatiques par l'intensité moyenne des observations de référence. Cette opération a pour effet d'exprimer à nouveau les observations dans l'espace de couleurs *RGB*, en les associant cette fois à des conditions d'illumination normalisées. Cette technique sera présentée plus en détails au chapitre 3.

Gevers et Smeulders [47] analysent huit espaces de couleur invariants par rapport aux changements d'illumination, dont l'espace des coordonnées chromatiques et l'espace Teinte-Saturation. La sensibilité de chacun de ces espaces est évaluée dans le cadre de la reconnaissance d'objets, et en présence de multiples effets de l'illumination, tels que la direction de l'éclairage, son intensité, sa couleur, la présence de zones sur-éclairées ou d'inter-réflexions. La conclusion de ces travaux est que l'espace classique de couleurs *RGB* n'est adapté que dans le cas où les conditions d'éclairage sont contrôlées (e.g. en intérieur). Dans le cas où l'éclairage est blanc et où il n'y a pas de zones sur-éclairées, alors l'espace des coordonnées chromatiques classiques (r,g,b) ou la variante définie par l'espace $C1C2C3$ sont les plus adaptés. En présence de zones sur-éclairées, il est recommandé de travailler avec la teinte (*hue* dans la littérature) ou dans la variante définie par l'espace $L1L2L3$. Finalement, dans le cas où aucune contrainte n'est posée sur la nature de l'éclairage ou sa couleur, alors l'espace $m_1m_2m_3$ est le plus adapté, mais celui-ci fait intervenir le voisinage des pixels et ne peut donc être calculé indépendamment en chaque pixel.

Enfin, Finlayson et al. [44] introduisent une méthode permettant de filtrer les effets de l'illumination dans une image. Dans un premier temps, il est démontré que dans le cas de la diffusion Lambertienne, d'une caméra ayant une sensibilité spectrale en forme de distribution de Dirac et d'une illumination suivant la loi de radiation des corps noirs [3], alors les coordonnées chromatiques logarithmiques $(\chi_R,\chi_G,\chi_B))$ peuvent s'exprimer linéairement par rapport à $\frac{1}{T}$, T étant la température de corps noir. Ils en déduisent alors qu'en connaissant la direction de variation linéaire de ces coordonnées chromatiques par rapport à la température d'illumination, il est possible de filtrer l'information dépendant de l'illumination, en effectuant une projection par rapport à la direction orthogonale.

De manière générale, les approches cherchant à estimer la réflectance ont cependant une limite majeure : la majorité des modèles physiques correspondants sont non-linéaires et prennent en compte le produit de la réflectance et de l'illumination. Il est donc difficile de traiter l'information liée à la scène de manière complètement indépendante de l'information liée à l'illumination. Par conséquent, ce type d'approche fait généralement un compromis entre fausses alarmes et non-détections. Ainsi, les approches parvenant à filtrer très efficacement les effets de l'illumination, comme celle de Finlayson et al. [44], filtrent en même temps une partie de l'information liée à la scène et peuvent donc ne pas détecter une partie des changements significatifs. Inversement, les approches capables de conserver une majorité des changements significatifs, comme celle de Elgammal et al. [38], conservent également une partie des effets de l'illumination, générant des fausses alarmes.

2.3.3 Autres sources de variabilité

Bien que les effets géométriques et les variations d'illumination soient les sources de variabilité les plus fréquemment abordées dans la littérature, nous avons vu dans le chapitre 1 qu'il en existait beaucoup d'autres. Quelques travaux se sont intéressés au traitement de certaines de ces

3. La loi de radiation des corps noirs est également appelée approximation de Wien de la loi de radiation thermique, qui est notamment valable pour la lumière du soleil.

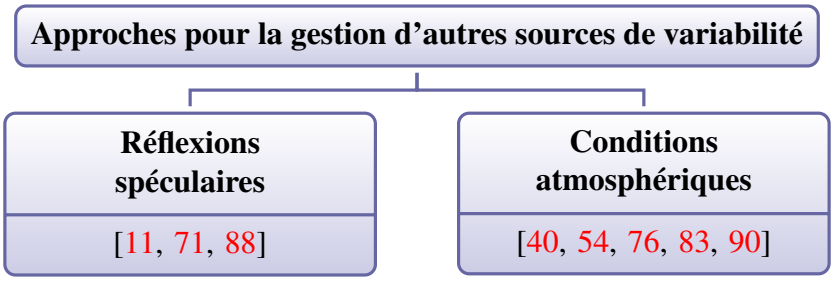


FIGURE 2.7 – *Cet arbre présente une taxonomie des sources de variabilités plus rarement abordées dans la littérature.*

sources dans le contexte de la détection de changements. La figure 2.7 propose une taxonomie de ces sources secondaires de variabilités et liste les approches correspondantes.

Réflexions spéculaires Les réflexions spéculaires sont des réflexions de lumière intense par des matériaux réfléchissants (e.g. métal, vitre, etc) et elles se manifestent par une saturation de l'intensité des pixels dans les images. Du fait de cette saturation, elles sont difficilement abordables par les techniques permettant d'atténuer les effets de l'illumination en général et nécessitent des techniques adaptées [11, 71, 88].

La formulation proposée par Black et al. [11], qui permet d'aborder de nombreuses variabilités à condition de fournir la modélisation associée, propose un modèle pour identifier les changements dus aux réflexions spéculaires. Ce modèle utilise une somme pondérée de trois vecteurs de base, un constant et deux linéaires selon les directions horizontale et verticale, dont les paramètres de pondération sont estimés à l'aide de l'algorithme EM [36].

Ping et al. [88] proposent une méthode permettant d'éliminer les reflets de lumière dans les images, à condition que la zone à traiter soit détourée manuellement. Cette méthode utilise ensuite une technique de diffusion anisotrope pour éliminer la composante spéculaire de l'intensité observée, et ne conserver que la composante diffuse. Cependant cette approche nécessite l'intervention de l'utilisateur pour détourer la zone à traiter, ce qui la rend peu adaptée pour le traitement de vidéos.

Pour aborder ce problème, Mallick et al. [71] ont proposé une technique alternative, qui permet d'effectuer l'élimination des reflets en détectant automatiquement les zones à traiter. Pour cela, leur technique se base sur la morphologie différentielle et, à l'aide de différents jeux d'équations aux dérivées partielles respectivement adaptés à différents scénarios, ils parviennent à séparer les composantes diffuse et spéculaire des images ou vidéos considérées.

Conditions atmosphériques Dans le cas d'observations acquises en extérieur, les conditions atmosphériques, telles que le brouillard ou la présence dans l'atmosphère d'aérosols dispersifs quelconques, peuvent provoquer des variations visuelles relativement importantes. Certains travaux [40, 54, 76, 83, 90] se sont intéressés à ce problème en essayant d'atténuer les conditions atmosphériques (*dehazing* dans la littérature).

Pour remédier à la perte de contraste provoquée par la présence de brouillard, Narasimhan et Nayar [83] proposent une technique permettant de restaurer ce contraste. La perte de contraste est décrite par la loi de Beer-Lambert, où l'atténuation dépend exponentiellement de la profondeur du point considéré. L'article décrit une méthode permettant d'estimer la carte de profondeur à l'aide de deux images issues d'une même caméra fixe, dans le cadre de la vidéosurveillance. Une fois connue cette carte des profondeurs, deux cas peuvent se présenter. Soit elle peut être segmentée en quelques ensembles de pixels de profondeurs identiques, et les auteurs montrent qu'il est alors possible de restaurer l'atténuation exponentielle du contraste, en considérant la moyenne des intensités sur chaque ensemble de pixels. Lorsque la carte de profondeur ne peut pas être segmentée en quelques ensembles de pixels de profondeurs identiques,

une seconde méthode est proposée, exploitant cette fois la connaissance de points noirs dans la scène et dont la luminosité apparente ne dépend donc que de l'atténuation due au brouillard. Pollard [90] propose d'utiliser une modélisation plus simple dans laquelle les intensités de deux images acquises dans des conditions atmosphériques différentes sont liées par une relation affine. Les deux coefficients de cette relation affine, identiques en chaque pixel, peuvent être estimés grâce au modèle d'apparence 3D utilisé par l'auteur, qui permet à la fois un appariement précis des pixels et une modélisation des apparences. Comme la précédente, cette approche nécessite plusieurs images pour pouvoir restaurer la perte de contraste due aux conditions atmosphériques.

Une autre catégorie d'approches s'est intéressée à ce problème en exploitant une image unique. Ainsi, la méthode introduite par Fattal [40] permet d'estimer la réflectance de la scène, sous l'hypothèse que l'illumination de la surface et la fonction de dispersion de la lumière dans l'atmosphère sont localement décorrélées. L'estimation de la carte d'atténuation du contraste est alors estimée à l'aide d'une analyse en composantes indépendantes. Cette approche peut cependant échouer lorsque les images sont bruitées ou de mauvaise qualité, ou en présence de brouillard dense. Pour aborder ce dernier problème, He et al. [54] ont proposé d'analyser les observations sombres sur de petits voisinages de chaque pixel, afin d'en déduire une estimation grossière de la carte d'atténuation du contraste. Cette estimation grossière est ensuite affinée via un algorithme de dé-cachage d'image (*image matting* dans la littérature), visant à estimer le premier-plan, le fond, et la fonction de composition. Une difficulté importante pour ces méthodes est la présence de bruit dans les images, qui est courante dans les images aériennes, et qui peut faire échouer la restauration de contraste du fait de l'ambiguïté inhérente du problème. Pour cela, Matlin et Milanfar [76] ont proposé une méthode de régression itérative et non-paramétrique, basée sur l'estimation alternée de la composante due à l'atmosphère et de celle due à la scène.

2.4 Comparaison d'une observation avec une référence

Dans le cadre de la détection de changements, la problématique de la comparaison d'une observation de test avec une référence est centrale et a été abordée par une myriade de méthodes. L'objectif consiste à estimer un masque de changements qui, à chaque pixel de l'image de test, associe une étiquette qui le plus souvent est simplement binaire et distingue les pixels ayant changé de ceux n'ayant pas changé. Plus généralement, les valeurs prises par ce masque de changements peuvent être discrètes, et les étiquettes associées peuvent représenter diverses catégories de changements identifiées directement à partir des observations radiométriques [4]. Par conséquent, le problème de l'estimation de ce masque de changements peut être formulé comme un problème de classification à deux classes ou plus, dont au moins une représente les zones n'ayant pas changé et au moins une autre représente les zones ayant changé. Selon les méthodes, les autres classes peuvent représenter différentes catégories de changements significatifs et/ou différentes catégories de changements non pertinents. Comme dans la majorité des problèmes de classification, la principale difficulté réside dans le fait que ces classes peuvent éventuellement se chevaucher, ce qui rend leur distinction délicate pour un système automatique.

Selon les applications visées et les choix effectués pour répondre aux autres problématiques indépendantes, le mécanisme d'affectation des classes peut bénéficier de différentes opportunités, menant à différentes techniques de classification. La figure 2.8 présente une taxonomie

4. Il est important de noter la distinction entre une méthode identifiant des catégories de changements entre deux observations, d'une autre identifiant des changements de catégories d'observation. Ce deuxième type de méthode, comme évoqué dans le chapitre 1, ne correspond pas à la détection de changements telle qu'entendue dans le cadre de cette thèse.

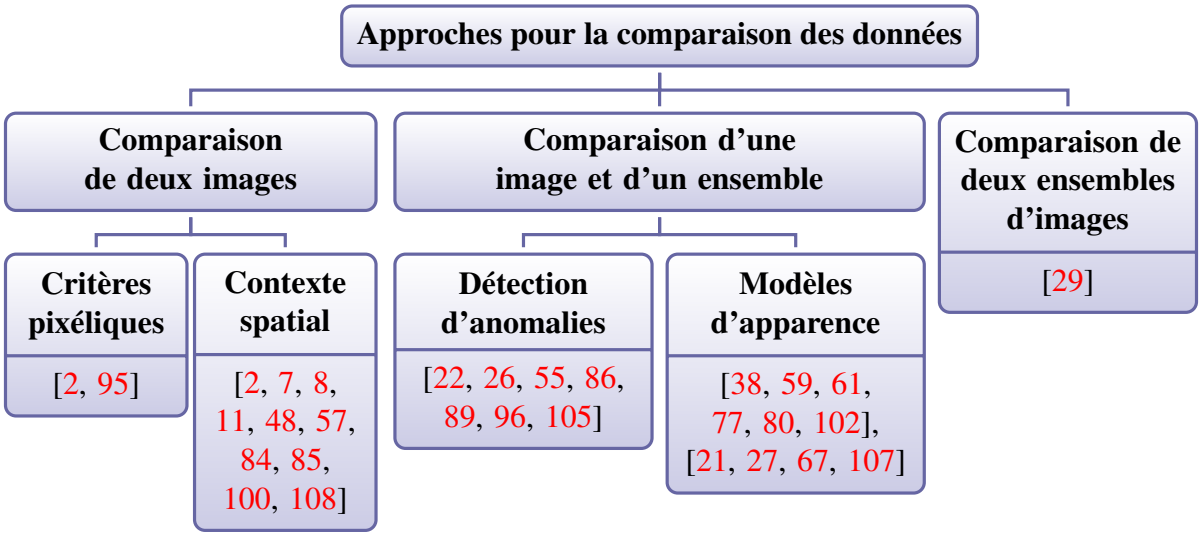


FIGURE 2.8 – *Cet arbre présente une taxonomie des différentes approches utilisées pour aborder le problème de la comparaison des données en détection de changements.*

des techniques rencontrées dans la littérature. Comme évoqué dans la section 2.2, dans certains cas cette affectation se fait uniquement sur la base d'une image de test et d'une image de référence, tandis que dans d'autres, il est possible de disposer d'une image de test et d'un ensemble d'images de référence. En poursuivant le raisonnement, la comparaison d'un ensemble d'images de test à un ensemble d'images de référence peut également être envisagée, point qui sera discuté à la fin de cette section bien que les méthodes correspondantes soient extrêmement rares dans la littérature.

Comparaison de deux images Une première catégorie d'approches s'intéresse à l'estimation du masque de changements sur la base d'une unique image de test et d'une unique image de référence. C'est par exemple le cas lorsque deux images satellitaires doivent être comparées, ou qu'une fusion de plusieurs images de référence a été effectuée pour en obtenir une unique, qui doit ensuite être comparée à une image de test.

Comparaison pixélique Dans de tels cas, l'approche la plus simple consiste à analyser indépendamment chaque pixel de l'image de test par rapport au pixel correspondant de l'image de référence. Une approche populaire parmi les premières méthodes de détection de changements [95] consistait à exploiter une mesure très simple sur les intensités des pixels de test et de référence, par exemple la différence ou le ratio, et à utiliser un seuil déterminé expérimentalement pour obtenir une classification binaire. Il est rapidement apparu qu'un tel seuil, déterminé sur quelques images, n'était pas forcément applicable tel quel en exploitation.

Par conséquent, les auteurs ont cherché à formaliser le choix de ce seuil dans le cadre de tests d'hypothèse statistique [60]. Ces approches s'intéressent généralement au problème de classification binaire, visant à distinguer les changements des non-changements, et exploitent la connaissance de la distribution de probabilité des valeurs de la mesure utilisée (différence ou ratio des intensités) pour déterminer un seuil optimal. La distribution de probabilité associée à la classe des non changements est généralement supposée suivre une loi gaussienne, car les variations d'intensités peuvent alors être assimilées à du bruit. Aach et al. [2] expliquent que, lorsque la distribution de probabilité associée à la classe des changements est inconnue, le seuil optimal peut être défini grâce à un test de signification (*significance test* dans la littérature) et à un objectif sur le taux de fausses alarmes. En revanche, lorsque cette distribution est connue, ce qui requiert d'avoir modélisé le comportement statistique de la mesure utilisée en présence

de changements, il est possible d'estimer plus précisément le seuil optimal grâce à un test du rapport des vraisemblances (*likelihood ratio test* dans la littérature).

Exploitation du contexte spatial Les techniques précédentes, qui analysent indépendamment les pixels des images de test et de référence, ont l'avantage d'être simples mais sont également très sensibles au bruit dans les images. Cela a généralement pour effet de rendre le masque de changements très bruité, c'est-à-dire que les classes estimées pour deux pixels voisins sont rarement cohérentes et donc peu fiables. Pour éviter cela, il est préférable d'introduire la notion de contexte spatial, et d'estimer la classe d'un pixel en exploitant les intensités des pixels voisins. Les approches exploitant cette idée sont très diverses [2, 7, 8, 11, 48, 57, 84, 85, 100, 108].

Par exemple, Hsu et al. [57] exploitent une modélisation quadratique par morceaux du signal image. Un test d'hypothèse statistique est ensuite utilisé pour décider si les coefficients polynomiaux, obtenus sur les images de référence et de test, correspondent ou non à un changement significatif. Ce test d'hypothèse permet donc d'estimer la classe de chaque pixel grâce à l'analyse des pixels voisins.

Sarkar et Boyer [100] remarquent que les changements significatifs s'accompagnent souvent d'une évolution du niveau d'organisation entre primitives visuelles. Ils développent donc une théorie basée sur les graphes relationnels pour décrire l'organisation spatiale de primitives ciblant les contours. Différentes mesures du changement basées sur les valeurs propres de la matrice représentant le graphe relationnel sont finalement combinées et une classification bayésienne est effectuée pour détecter les changements.

Black et al. [11] proposent une méthode permettant d'identifier plusieurs classes de variations des intensités, associées à des modèles paramétriques connus, et une classe de changements inconnus. Les modélisations paramétriques a priori des effets de chaque type de variation sont pondérées pour former une loi de probabilité décrivant la transition de l'image de référence vers l'image de test. L'algorithme EM [36] est ensuite utilisé pour maximiser la vraisemblance du modèle total, en estimant alternativement les coefficients de pondération et les paramètres de chaque modèle. Un procédé hiérarchique (*coarse-to-fine* dans la littérature) est également utilisé pour permettre l'estimation d'éventuels mouvements importants entre les deux images.

Tsai et Lai [108] proposent une méthode de comparaison entre une image de référence et une image de test basée sur l'Analyse en Composantes Indépendantes (ACI). Cette approche cherche à séparer l'image de test considérée en la somme de l'image de référence et d'une image statistiquement indépendante de l'image de référence, ne contenant donc que les changements. Cette indépendance statistique se traduit en théorie par la nullité de la différence de la distribution de probabilité jointe et du produit des distributions de probabilités marginales. En pratique, une mesure approchée de cette différence, basée sur des histogrammes, est minimisée par une méthode d'optimisation par essaim de particules (*particle swarm optimization* dans la littérature). Finalement, l'image statistiquement indépendante de l'image de référence est seuillée pour décider de la classe binaire de chaque pixel.

Par ailleurs, de nombreuses méthodes [95] ont utilisé le cadre théorique apporté par la notion de champ de Markov aléatoire caché (*hidden random Markov field* dans la littérature). Par exemple, Aach et al. [2] modélisent une image grâce à un tel champ de Markov connectant les pixels voisins horizontalement, verticalement et diagonalement. La notion de masque de changements idéal est définie a priori, via la définition d'une énergie potentielle appropriée, comme étant celui dans lequel le nombre de changements de classe entre pixels voisins est minimal. Ce masque de changements est estimé par une méthode itérative basée sur l'algorithme de relaxation *Iterative Conditional Mode* (ICM). Nava et al. [84] présentent une approche exploitant un champ de Markov similaire. Cependant, l'énergie potentielle à minimiser est composée d'un terme d'interaction, encourageant les pixels voisins à avoir la même classe, et d'un terme d'attache aux données, comparant la différence d'intensité à une valeur d'écart type différente pour

chacune des deux classes. Le masque de changements minimisant cette énergie potentielle est finalement estimé par une méthode de Graph Cuts [20].

Comparaison d'une image à un ensemble d'images Dans le cas où un ensemble d'images de référence est disponible, il est possible d'exploiter leur recouvrement pour modéliser la dynamique "normale" de la scène. Cette modélisation représente un apport d'information non négligeable par rapport aux méthodes n'utilisant qu'une unique image de référence, débouchant sur une estimation plus fiable du masque de changements.

Détection d'anomalies génériques Le problème de l'identification d'un individu anormal parmi une population, ou dans notre cas, d'une observation représentant un changement par rapport à un ensemble d'observations normales, a fait l'objet de recherches spécifiques dans le cadre du domaine de la détection d'anomalies [22, 26, 55, 86, 89, 96, 105]. La détection de changements dans des images est un cas applicatif particulier de la détection d'anomalies, qui en généralise beaucoup d'autres, tels que la détection de fraudes bancaires ou d'intrusions informatiques.

Chandola et al. [26] proposent une revue de la littérature correspondante, et identifient plusieurs grandes catégories d'approches : par classification, par comparaison aux plus proches voisins, par agglomération (*clustering* dans la littérature) ou par méthodes statistiques. Les approches par classification supervisée utilisent des exemples étiquetés pour apprendre à distinguer les individus normaux des individus anormaux. Or, obtenir des exemples étiquetés d'individus anormaux est généralement difficile, du fait de la rareté de leur apparition ou de la connaissance très floue des anomalies possibles. Il existe donc des approches dites de classification à une classe (*one-class classification* dans la littérature), qui nécessitent seulement des exemples étiquetés d'individus normaux. À ce sujet, il est par exemple possible de citer les travaux de Tax [105] ou ceux de Ratsch et al. [96].

Une autre famille d'approches exploite les plus proches voisins pour décider de manière non-supervisée si les individus sont normaux ou anormaux. Les méthodes les plus efficaces dans cette catégorie utilisent pour cela la densité locale des individus. Par exemple, Breunig et al. [22] introduisent le facteur local d'anormalité (*local outlier factor* dans la littérature), qu'ils définissent comme le rapport entre d'une part, la moyenne des densités locales autour des plus proches voisins d'un individu, et d'autre part, la densité locale autour de cet individu. La classification binaire entre individus normaux et anormaux peut finalement être obtenue par un seuil sur le facteur local d'anormalité. Une seconde méthode relativement similaire proposée par Papadimitriou et al. [86] définit l'anormalité d'un individu à l'aide du facteur de déviation à granularité multiple. Ce facteur représente, pour un individu et une granularité donnés, la déviation de la densité locale autour de cet individu par rapport à la moyenne des densités locales à la granularité considérée, c'est-à-dire les densités locales autour des individus présents dans un rayon donné autour de l'individu central. L'avantage de cette méthode est qu'elle fournit une valeur de seuil appropriée permettant d'obtenir la classification binaire entre individus normaux et anormaux. Ces deux méthodes posent cependant un problème en pratique : dans le cas d'un flux de données, un nouvel individu peut potentiellement modifier la densité locale autour de chaque individu déjà connu, entraînant une charge de calcul non-négligeable pour le traitement des nouveaux individus. Pour résoudre ce problème, Pokrajac et al. [89] proposent une extension incrémentale de la méthode de Breunig et al. [22], démontrant qu'en réalité il n'est pas nécessaire de mettre à jour la totalité des calculs de distance entre voisins.

Les approches par agglomération construisent, également de manière non-supervisée, des agglomérats à partir d'individus non-étiquetés, et déclarent anormaux ceux ne pouvant être agrégés à aucun agglomérat, ou formant de minuscules agglomérats. Dans cette optique, He et al. [55] proposent un facteur local d'anormalité basé sur les agglomérats, dont la définition dépend de la taille de l'agglomérat de l'individu considéré. Si cet individu appartient à un petit

agglomérat, son facteur d'anormalité est le produit de la taille de cet agglomérat par la distance entre cet individu et le centre du grand agglomérat le plus proche. En revanche, si l'individu considéré appartient à un grand agglomérat, son facteur d'anormalité est le produit de la taille de cet agglomérat par la distance entre cet individu et le centre de ce grand agglomérat. La classification binaire entre individus normaux et anormaux est finalement obtenue par un seuil sur le facteur local d'anormalité. En pratique, la performance des approches de ce type est généralement fortement liée à la qualité de la méthode d'agglomération, qui peut par ailleurs engendrer une charge de calcul importante selon la technique utilisée. De plus, les approches par agglomération peuvent mener à de très mauvaises performances si la distribution des individus normaux n'est pas adaptée, ce qui est rarement prévisible.

Malgré leur intérêt objectif, toutes les méthodes de détection d'anomalies génériques ne sont pas adaptées au problème de la détection de changements dans des images. Par exemple, les méthodes de classification qui nécessitent un apprentissage sur des données étiquetées ne sont pas très flexibles. En effet, en cas de légère modification d'objectif, impliquant l'application de ces méthodes sur des données non-prévues lors de la construction de la base d'apprentissage, il est alors nécessaire de la mettre à jour avec de nouvelles données étiquetées. Ce problème peut par exemple survenir pour prendre en compte des données contenant une variété d'effets de l'illumination, ou d'autres types de variations particulières. Ainsi, la maintenance d'une base d'apprentissage pertinente demande un effort important. Par ailleurs, la plupart des méthodes utilisant la densité locale des individus [22, 86, 89] requièrent de conserver ces individus en mémoire, afin de calculer leurs distances par rapport à un nouvel individu dont l'anormalité doit être analysée. Dans le cas de la détection de changements dans une vidéo, c'est rigoureusement impossible car cela nécessiterait de conserver, sans compression, les intensités de chaque pixel de chaque image de la vidéo considérée. Or, une heure de vidéo haute définition (1280×720 pixels par image) en couleur et acquise à 25 images par seconde, représenterait par exemple environ 250 giga-octets en mémoire. Par conséquent, les méthodes visant à détecter des changements dans des vidéos, en particulier les méthodes de soustraction de fond [19, 87], ont développé des techniques adaptées à cette problématique particulière, utilisant la modélisation des apparences. Ces approches consistent à construire incrémentalement un modèle permettant la compression des observations passées tout en autorisant par la suite la détection de changements.

Modèles d'apparence pixéliques Un certain nombre de techniques de modélisation d'apparence calculent des modèles indépendants en chaque pixel des images [38, 59, 61, 77, 80, 102]. Stauffer et Grimson [102] proposent par exemple un algorithme incrémental des K-moyennes permettant l'estimation en ligne de modèles par mélanges finis de gaussiennes. Chacun de ces modèles contient un nombre maximal fixé de distributions gaussiennes, qui sont triées de manière à favoriser celles ayant une faible variance et étant estimées à partir de nombreuses observations. Chaque nouvelle observation est incorporée dans la première distribution compatible rencontrée, c'est-à-dire pour laquelle l'intensité de la nouvelle observation est à une distance prédéfinie de la moyenne. Si aucune distribution n'est compatible, la moins probable de toutes est remplacée par une nouvelle distribution centrée sur la nouvelle observation. Sinon, la première distribution compatible est mise à jour avec la nouvelle observation à l'aide d'un taux d'apprentissage permettant l'oubli progressif des observations les plus anciennes. Finalement, l'observation courante est déclarée comme non-changement si elle est compatible avec l'une des deux premières distributions (nombre paramétrable), et comme changement sinon. Notons ici que l'oubli progressif, qui est mis en œuvre dans la plupart des techniques de soustraction de fond, représente une différence majeure entre ce domaine et celui de la détection de changements entre séquences. En effet, le premier vise à détecter les changements dans l'image la plus récente d'une séquence, par rapport aux images passées de la même séquence, et il est

alors intéressant d'oublier les observations les plus anciennes, de manière à pouvoir s'adapter aux variations lentes mais extrêmes dans la scène (e.g. cycle jour/nuit). Au contraire, celui de la détection de changements entre séquences cherche à comparer deux séquences acquises à des dates différentes, et il faut alors tenir compte de toutes les observations de la séquence de référence et ne pas oublier les plus anciennes.

D'autres techniques de soustraction de fond ont par la suite été proposées pour résoudre certains problèmes liés à la méthode de Stauffer et Grimson [102]. Ainsi, Kaewtrakulpong et Bowden [59] affirment que les équations utilisées par Stauffer et Grimson [102] génèrent une mise à jour trop lente des distributions et sont très sensibles aux conditions d'initialisation, ce qui peut mener à des difficultés de modélisation. Ils proposent donc d'utiliser de nouvelles équations de mise à jour, sans oubli progressif, durant la phase d'initialisation, puis passent à des équations avec oubli progressif lorsqu'un nombre suffisant d'observations ont été intégrées au modèle. Elgammal et al. [38] considèrent que l'utilisation de modèles paramétriques de distributions, en particulier de mélanges de gaussiennes, peuvent ne pas correspondre à la distribution sous-jacente réelle des observations. Ils proposent donc d'utiliser la technique d'estimation de densité par noyaux (*kernel density estimation* dans la littérature) afin de modéliser très précisément la distribution réelle des données. Dans ce cas, un seuil sur la probabilité d'apparition de l'observation est utilisé pour estimer le masque de changements. La distribution des observations est modélisée comme une somme pondérée de noyaux centrés sur un nombre fixé des dernières observations. Il n'est donc pas possible avec cette méthode de passer outre l'oubli progressif typique des méthodes de soustraction de fond, mais indésirable pour la détection de changements, à moins de conserver la totalité des observations en mémoire, ce que nous avons vu être impossible. Par ailleurs, la largeur de noyau utilisée pour modéliser la distribution réelle, qui est un paramètre de la méthode, est délicate à régler a priori. Pour contourner cette dernière difficulté, Mittal et Paragios [80] proposent une méthode permettant d'adapter cette largeur de noyau en fonction des observations passées et de l'observation courante. Afin d'optimiser les contraintes mémoire et les temps de calcul, Kim et al. [61] proposent d'utiliser un modèle par dictionnaires (*codebooks* dans la littérature). L'algorithme proposé agglomère incrémentalement les observations successives dans des agglomérats sensiblement équivalents aux gaussiennes utilisées dans les modèles par mélanges de gaussiennes, mais qui ont l'avantage d'éviter les calculs coûteux à base de fonctions exponentielles.

Modèles d'apparence contextuels De la même manière que pour la comparaison de deux images, il est préférable lors de la comparaison d'une image de test à un modèle d'exploiter le contexte spatial pour estimer le masque de changements [21, 27, 67, 107].

Dans cette optique, Toyama et al. [107] combinent des modèles d'apparence calculés indépendamment en chaque pixel, avec des traitements effectués à l'échelle de groupes de pixels et également de l'image entière. À l'échelle du pixel, un filtre prédictif de Wiener utilise les observations passées les plus récentes pour prédire la valeur de la prochaine observation, et toute déviation significative est considérée comme un changement. À l'échelle d'un groupe de pixels, le déplacement d'objets homogènes, causant généralement des problèmes de non-détection pour les pixels intérieurs, est détecté et les pixels intérieurs analysés plus finement. Enfin, à l'échelle de l'image, les changements brutaux sont détectés lorsque plus de 70% des pixels de l'image sont déclarés comme changements, et sont traités en maintenant plusieurs modèles de fond et en sélectionnant celui générant le nombre minimal de changements.

Les méthodes exploitant une modélisation hiérarchique ou multi-échelle consistent à adapter le mécanisme de comparaison utilisé en fonction de l'échelle considérée. Par exemple, une méthode de comparaison rapide peut être utilisée sur un ensemble de zones de l'image considérée afin d'identifier les zones candidates pouvant contenir des changements intéressants, puis une méthode plus lente mais plus précise peut être employée sur chaque zone candidate pour confirmer la présence de changements significatifs. Cette approche hiérarchique a par exemple

été employée par Chen et al. [27], qui combinent une méthode par bloc, rapide mais peu précise en termes de localisation des changements, avec la méthode pixélique de Stauffer et Grimson [102]. L'analyse de l'image courante est d'abord effectuée selon l'approche par bloc, à l'aide d'un descripteur de bloc d'image basé sur un histogramme de valeurs de contraste. Au sein de chaque quadrant d'un même bloc, les intensités des pixels sont comparées à l'intensité du pixel central, et les valeurs positives et négatives sont sommées indépendamment. Cette approche par bloc identifie un certain nombre de zones candidates pouvant contenir des changements. Chaque zone est ensuite considérée indépendamment et subit un traitement différent selon qu'elle est candidate pour la détection de changements ou non. Si la zone considérée n'est pas candidate, alors les modèles pixéliques d'apparence sont simplement mis à jour avec les intensités contenues dans la zone, mais aucune détection de changements pixélique n'est réalisée. Inversement, si la zone est candidate, alors la détection de changements pixélique est effectuée, mais les modèles d'apparence ne sont pas mis à jour. Cette approche hiérarchique permet de réduire les temps de calcul par rapport à une exécution des deux approches indépendamment, et permet de plus d'effectuer la détection de changements en exploitant le contexte, menant ainsi à une meilleure performance de détection.

Enfin, Li [67] propose une méthode de soustraction de fond exploitant une formulation incrémentale de l'Analyse en Composantes Principales (ACP), appliquée aux images formatées comme des vecteurs colonnes de très grande dimension. L'idée de cette technique consiste à interpréter les composantes principales comme des modes de la variation normale de la scène. Pour détecter les changements dans une image de test, le vecteur associé est projeté sur la base du sous-espace formé par les composantes principales, et le vecteur résiduel, défini comme la différence entre le vecteur d'origine et le résultat de cette projection, est comparé au vecteur des écart-types. Un certain nombre de traitements supplémentaires sont décrits, afin de permettre la prise en compte d'éventuelles données manquantes dans les images. Dans la même veine, Brand [21] détaille la formulation incrémentale de la décomposition en valeurs singulières (*singular value decomposition* dans la littérature). Cette formulation permet en particulier de prendre en compte à la fois des données manquantes ainsi que des a priori d'incertitude sur les observations. Ces deux dernières approches partagent une limite commune : la formulation incrémentale permettant leur tractabilité mènent à des approximations, qui, dans le résultat final, se traduisent par une reconstruction seulement partielle de la variance totale observée, à l'aide des vecteurs propres estimés.

Comparaison de deux ensembles d'images Le raisonnement consistant à exploiter le recouvrement des images de référence pour améliorer la connaissance de la scène avant apparition des changements, et donc les performances de détection de changements, est également applicable aux images de test. En effet, il semble naturel qu'une meilleure connaissance de la scène de test, c'est-à-dire de la scène après apparition des changements, permette de mieux caractériser les changements présents dans la scène.

Dans cette optique, Clifton [29] propose une méthode permettant de comparer un ensemble d'images de référence à un ensemble d'images de test. Pour cela, un classificateur à base de réseau de neurones est utilisé pour prédire les observations de test à partir des observations de référence, et inversement. Les déviations significatives des observations réelles par rapport aux prédictions sont considérées comme des changements potentiels. Ces changements potentiels sont par la suite confirmés ou infirmés selon qu'une quantité significative d'autres changements potentiels sont également présents, ou non, dans leur voisinage.

Cependant, en pratique, ce type d'approche est peu utilisé dans le contexte de vidéos aériennes, car il est généralement souhaitable de pouvoir détecter les changements de manière incrémentale, à mesure que les images de la vidéo sont réceptionnées. Nous verrons dans la section 2.5 qu'un certain nombre d'approches ont été proposées pour exploiter le recouvrement des images de test tout en répondant au besoin de traitement incrémental.

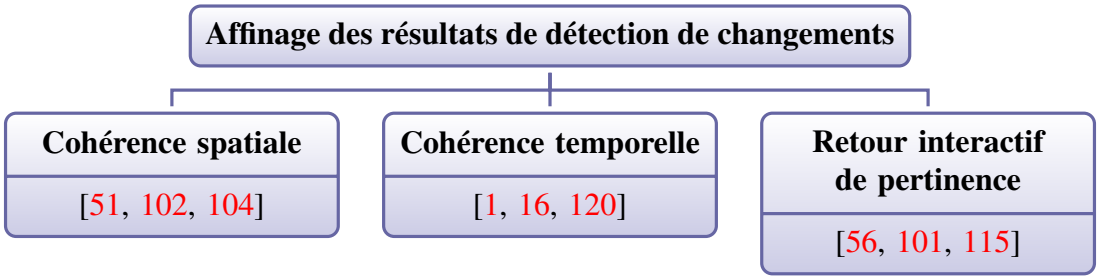


FIGURE 2.9 – *Cet arbre présente une taxonomie des techniques utilisées pour affiner les résultats de détection de changements.*

2.5 Affinage des résultats de détection de changements

Les résultats bruts de comparaison de données d'observation contiennent généralement de nombreuses imprécisions plus ou moins aléatoires, dues notamment au bruit dans les observations. Par conséquent, il est courant de procéder, après l'étape de comparaison à proprement parler, à une étape d'affinage des résultats. Cette étape prend généralement la forme de post-traitements appliqués au masque de changements, qui peuvent néanmoins être intégrés au mécanisme de traitement incrémental d'une vidéo de test. Une vaste gamme de techniques d'affinage peuvent être appliquées, avec des objectifs différents. Dans cette section, nous décrivons donc les différentes catégories de techniques indépendamment les unes des autres. La figure 2.9 présente une représentation visuelle de ces catégories.

Cohérence spatiale Une technique de post-traitement couramment rencontrée [95] en détection de changements consiste à améliorer la cohérence spatiale du masque de changements. En effet, le bruit dans les observations peut générer des variations de classes injustifiées par le contenu objectif de la scène observée. Par conséquent il peut être intéressant de lisser le masque de changements afin de le rendre plus fiable.

Haralick et Shapiro [51] présentent par exemple le filtre médian permettant d'éliminer les petits groupes de pixels dont la valeur est différente de la valeur dominante des voisins. Le filtre médian considère le voisinage d'un pixel donné pour remplacer son intensité par l'intensité médiane des voisins. Cette approche est couramment utilisée pour la réduction de bruit, et peut être également appliquée en détection de changements pour la réduction du bruit dans l'estimation des classes.

La méthode proposée par Stauffer et Grimson [102] génère également du bruit dans l'estimation des pixels ayant changé, résultant en de petits groupes de pixels isolés répartis uniformément dans les images. Pour filtrer ce bruit, les auteurs proposent d'utiliser un seuil sur le nombre de pixels formant ces groupes, afin d'éliminer ceux constitués d'un faible nombre de pixels.

Stringa [104] discute de plusieurs types de filtres morphologiques utilisables en détection de changements pour une large gamme d'applications. Les filtres morphologiques permettent non seulement d'éliminer les petits groupes isolés de pixels détectés comme changements, mais aussi de lisser les frontières des régions plus importantes.

Il est à noter que ces techniques ne peuvent remplacer l'exploitation du contexte faite par certaines méthodes décrites à la section 2.4, et donnent des résultats nettement moins fiables. Cependant, l'utilisation conjointe de ces deux types d'approches peut permettre d'affiner les résultats finaux.

Consolidation temporelle Nous avons vu dans la section 2.4 qu'il pouvait être intéressant d'exploiter le recouvrement éventuel des images, et de comparer un ensemble d'images de référence à un ensemble d'images de test. Cependant, dans le cadre du traitement de vidéos,

ce type de comparaison est incompatible avec le besoin d'analyse incrémentale. Un compromis permettant de consolider les résultats obtenus sur l'image courante peut néanmoins être trouvé, par exemple en combinant les résultats obtenus sur les images successives d'une vidéo.

Ainsi, Aach et Kaup [1] reconnaissent qu'une méthode itérative est peu adaptée pour la détection de changements dans des vidéos du fait de l'importante charge de calcul nécessaire. Ils remarquent cependant qu'il est possible d'éviter ces itérations en exploitant la similarité des images successives de la vidéo, qui se traduit par une similarité des masques de changements associés. Le calcul du masque de changements de l'image courante se base donc sur le masque de changements calculé à l'image précédente.

Dans la même optique, Yin et Collins [120] proposent une méthode exploitant la similarité des détections dans les images successives d'une vidéo pour consolider les résultats de détection d'objets mobiles. Cette méthode modélise le problème à l'aide d'un champ de Markov aléatoire (MRF) spatio-temporel, ce qui revient à considérer la vidéo comme une pile d'images où chaque pixel (excepté ceux des bords) possède six voisins : quatre spatialement, aux positions adjacentes dans la même image, et deux temporellement, à la même position dans l'image précédente et l'image suivante. L'algorithme de détection des objets mobiles exploite la différence des images successives à l'aide d'un mécanisme de propagation de croyance (*belief propagation* dans la littérature), dont les fonctions d'attache aux données et de transition entre états sont adaptées à l'application visée.

Dans le cadre de cette thèse, une méthode exploitant la similarité des images successives dans une vidéo a été développée pour la détection de changements [16]. Deux algorithmes ont été étudiés, et seront présentés dans le chapitre 5. Le premier, qui est rapide mais modérément efficace, utilise une moyenne temporelle des scores de détection. Le second, qui est précis mais plus coûteux en temps de calcul, effectue une optimisation à l'aide du mécanisme de propagation de croyance.

Retour interactif pour la détection d'objets Par ailleurs, certaines méthodes cherchent à affiner les résultats de détection de manière interactive, en exploitant un retour de l'utilisateur sur leur pertinence (*relevance feedback* dans la littérature). Cette approche, initialement introduite dans le domaine de la fouille de données, a aussi été appliquée avec succès par certains auteurs dans le cadre de la détection d'objets.

Par exemple, Sjahputera et al. [101] décrivent un mécanisme de retour de pertinence destiné à adapter les changements présentés à l'utilisateur selon ses besoins. Dans une liste de blocs d'images satellitaires détectés comme contenant des changements, l'utilisateur peut sélectionner les candidats pertinents et invalider ceux étant jugés inutiles. Le mécanisme de retour de pertinence mis en œuvre utilise une mesure de similarité entre blocs basée sur un descripteur de contenu d'image. Le système classe alors les blocs non annotés par ordre de similarité croissante par rapport aux blocs jugés pertinents par l'utilisateur et élimine ceux dont la similarité par rapport aux blocs jugés inutiles est trop grande.

Yao et al. [115] proposent une méthode d'assistance à l'annotation de données en détection d'objets. Cette méthode utilise un détecteur d'objets à base de forêts aléatoires de Hough, combinant la transformée de Hough généralisée et les arbres de décision aléatoires. Les auteurs analysent l'évolution du coût d'annotation par l'utilisateur, qui dépend du nombre de corrections effectuées par rapport aux prédictions automatiques, en fonction du nombre d'images annotées. Ils montrent que non seulement ce coût décroit en fonction du nombre d'images annotées, mais également qu'il est inférieur au coût obtenu par d'autres méthodes, telles que le suivi ou l'interpolation d'annotations, montrant ainsi l'intérêt des approches d'apprentissage semi-automatique.

Dans le cadre de cette thèse, nous avons également développé une méthode exploitant un retour interactif de pertinence. Cette méthode, qui sera présentée au chapitre 5, utilise un des-

		Algorithme	
		+	-
Vérité-terrain	+	Vrai Positif	Faux Négatif
	-	Faux Positif	Vrai Négatif

TABLE 2.1 – *Rappel de la définition des quatre comptes d'évaluation, selon la catégorie estimée par l'algorithme de détection et la catégorie réelle définie par la vérité-terrain.*

cripteur de régions dédié et un apprentissage à base de machine à vecteurs de support (SVM, pour *Support Vector Machine* dans la littérature) linéaire.

2.6 Évaluation des algorithmes de détection de changements

Afin de pouvoir comparer objectivement la performance de différentes approches, une évaluation quantitative des algorithmes de détection de changements est nécessaire. Cette évaluation requiert d'exécuter les algorithmes sur des données pour lesquelles le résultat idéal, souvent appelé vérité-terrain, est connu. L'évaluation quantitative consiste alors à calculer une mesure objective permettant la comparaison du résultat réel de l'algorithme considéré avec la vérité-terrain.

Dans le cadre plus général de la détection d'objets, les mesures de comparaison font quasiment toutes intervenir ce que nous désignerons dans la suite par comptes d'évaluation. Ces comptes déterminent le nombre d'occurrence des quatre cas possibles selon la décision de l'algorithme (détection ou non) et le contenu de la vérité-terrain (présence d'un objet ou absence d'objet) :

- Vrais positifs (VP) : détection par l'algorithme et présence d'un objet,
- Faux positifs (FP) : détection par l'algorithme et absence d'objet,
- Vrais négatifs (VN) : pas de détection par l'algorithme et absence d'objet,
- Faux négatifs (FN) : pas de détection par l'algorithme et présence d'un objet.

Pour résumer, on parle de positif lorsque l'algorithme fait une détection, et de négatif lorsqu'il n'en fait pas. On qualifie alors ces positifs et négatifs de vrais ou de faux selon que l'algorithme a raison ou tort. La table 2.1 présente ces définitions à l'aide d'une matrice à quatre entrées.

Ces comptes d'évaluation peuvent être déterminés de différentes manières selon ce qui intéresse les auteurs. Dans le cas de l'assistance à l'analyse vidéo, il peut par exemple être utile de s'intéresser au nombre d'images sur lesquelles une alerte, nécessitant une intervention de l'utilisateur, a été levée. Dans ce cas, les comptes d'évaluation peuvent être calculés à l'échelle d'une image, en comptant par exemple un vrai positif s'il y a au moins une détection dans une image contenant au moins un objet d'intérêt. Dans le cas où l'objectif est simplement de focaliser l'attention de l'analyste sur les zones d'intérêt, une localisation précise des objets n'est pas nécessaire et ces comptes peuvent alors être calculés en considérant des blocs ou régions de l'image. Ceci peut par exemple être effectué en comptant un vrai positif lorsqu'une détection intersecte un objet de la vérité-terrain. Enfin, dans le cas où une localisation précise des objets est souhaitée, ces comptes peuvent être calculés en considérant chaque pixel de l'image, en comptant alors un vrai positif pour chaque pixel d'une détection correspondant à un objet dans la vérité-terrain.

À partir des comptes d'évaluation, il est fréquent de calculer des taux permettant de comparer les algorithmes indépendamment du nombre effectif de détections. Par exemple, les taux les plus souvent employés sont les suivants :

- Taux de vrais positifs (également appelé sensibilité ou rappel), égal à $\frac{VP}{VP+FN}$: probabilité empirique de faire une détection lorsqu'un objet est présent, qui peut être notée $P($ détection $|$ objet $)$,
- Taux de faux positifs, égal à $\frac{FP}{FP+VN}$: probabilité empirique de faire une détection lorsque aucun objet n'est présent, qui peut être notée $P($ détection $|$ $\overline{\text{objet}}$ $)$,
- Précision, égal à $\frac{VP}{VP+FP}$: probabilité empirique de faire une détection pertinente, qui peut être notée $P($ objet $|$ détection $)$.

L'analyse des performances à l'aide des taux de vrais positifs et de faux positifs est le plus souvent utilisée pour mesurer la précision de localisation des changements. Dans ce cas, il est préférable que les changements apparaissent de manière fréquente, par exemple à raison de plusieurs changements par image, afin d'obtenir une mesure fiable de la précision de localisation. D'autre part, l'analyse des performances à l'aide des mesures de précision et de rappel est le plus souvent utilisée pour mesurer la pertinence des détections. Dans ce cas, il est préférable que les changements apparaissent de manière peu fréquente, afin de vérifier que les détections ne surviennent que lorsqu'un changement est effectivement présent.

Un certain nombre d'autres mesures utilisant les comptes d'évaluation sont mentionnées par Radke et al. [95], mais lorsqu'il est possible de faire varier certains paramètres de détection, l'évaluation quantitative se fait le plus souvent par comparaison des courbes ROC (pour *Receiver Operating Characteristic*) [32, 34, 90]. Ces courbes ROC tracent le taux de vrais positifs en fonction du taux de faux positifs, et permettent d'évaluer le compromis entre faux positifs et faux négatifs, qui est inévitable pour les méthodes de détection d'objets.

Outre des mesures de comparaison adaptées, l'évaluation quantitative et objective des algorithmes de détection de changements nécessite deux choses : des données et une vérité-terrain associée. D'abord, il est nécessaire de disposer de données sur lesquelles tester les algorithmes, si possible acquises selon des conditions variées afin de pouvoir analyser l'influence de facteurs divers. Ainsi, il peut être intéressant d'analyser l'impact des différences de point de vue, l'influence des conditions d'illumination sur les performances, etc. Cela nécessite de disposer de données adaptées, qui, dans l'idéal, ne peuvent être obtenues qu'à l'aide de campagnes d'acquisition longues (à différentes heures du jour, voire différents jours de l'année) et pouvant nécessiter d'importants moyens, notamment dans le cas des vidéos d'observation aérienne. Ce problème est encore amplifié dans le cadre de la détection de changements, car il est nécessaire de disposer de plusieurs observations de la même zone, dont une partie contient des changements significatifs et pertinents par rapport à l'autre partie. À moins de disposer de données acquises sur théâtre d'opération, qui sont excessivement rares car souvent confidentielles, cela nécessite de disposer de moyens au sol pour mettre en œuvre des changements à détecter. Or, dans un tel scénario, la pertinence des changements mis en œuvre est forcément limitée (déplacements de véhicules, de personnes ou d'objets) car il est par exemple impensable de détruire un bâtiment ou de provoquer un feu de forêt pour l'occasion.

Le second point concerne l'obtention de la vérité-terrain associée aux données considérées. Dans le cadre de la détection de changements, cette vérité-terrain est généralement établie par un analyste expert qui doit fournir un travail d'annotation long et fastidieux. Radke et al. [95] discutent des variabilités engendrées par ce travail d'annotation. Par exemple, du fait de la définition relativement floue de ce qui constitue un changement significatif, différents analystes experts peuvent établir des vérités-terrain différentes, rendant ainsi l'évaluation des algorithmes subjective. De plus, la pénibilité de ce travail d'annotation fait qu'un même expert peut produire différentes vérités-terrain à deux instants différents. Cette difficulté d'obtenir la vérité-terrain est largement accrue dans le cas des données vidéo, car le volume de travail requis pour l'annotation de plusieurs heures de vidéos devient prohibitif. Certaines approches ont été proposées pour réduire l'effort d'annotation de données vidéo [115], par exemple en utilisant un apprentissage statistique combiné à un mécanisme de retour de pertinence.

Les articles de la littérature utilisent différentes approches pour contourner ces deux problèmes. Par exemple, certains auteurs présentent uniquement des résultats d'évaluation qualitatifs [92, 103], consistant le plus souvent en une superposition du masque de changements et de l'image d'origine. D'autres présentent des résultats quantitatifs obtenus sur des données artificielles [90, 91], générées par simulation ou synthèse d'images, pour lesquelles il est souvent possible d'extraire automatiquement la vérité-terrain. Cependant, les modèles de simulation utilisés pour la génération de données artificielles sont souvent plus simples que les processus physiques mis en œuvre dans la nature (e.g. absence de bruit, de perturbations diverses, etc). Par conséquent, une critique courante argue que ces données artificielles sont plus faciles à traiter par un algorithme automatique, introduisant donc un biais dans l'évaluation. Enfin, certains auteurs évaluent leurs algorithmes dans le cadre de problèmes connexes, comme par exemple la soustraction de fond [33]. Cela n'est pas toujours souhaitable, car les hypothèses de travail de ces problèmes connexes peuvent différer de celles de la détection de changements (e.g. oubli progressif dans le cas de la soustraction de fond, voir section 2.4).

Afin d'évaluer l'approche de détection de changements développée dans le cadre de cette thèse, nous avons proposé [18] d'insérer des changements virtuels dans des données réelles par réalité augmentée. Outre l'utilisation de vidéo réelles, cette technique, qui sera décrite plus en détails au chapitre 6, présente l'avantage considérable de permettre une extraction simple et rapide de la vérité-terrain.

2.7 Motivations

Pour conclure cet examen de l'état de l'art, cette section synthétise les principaux points contraignant la conception d'une approche globale pour la détection de changements dans des vidéos aériennes.

Modélisation incrémentale asymétrique Pour commencer, les méthodes permettant d'exploiter un ensemble d'observations de référence pour la détection de changements sont relativement nombreuses. Cependant, comme discuté dans la section 2.4, celles adaptées aux contraintes issues du traitement de données vidéo sont peu nombreuses. Par exemple, les vidéos de référence pouvant être longues, il est impossible de conserver en mémoire toutes les observations pour calculer un modèle, fidèle, seulement une fois qu'elles sont toutes disponibles. Il est donc crucial que l'algorithme utilisé permette la compression des observations successives et que la représentation utilisée puisse être mise à jour de manière incrémentale, c'est-à-dire à chaque fois qu'une nouvelle observation est disponible. De plus, comme mentionné dans la section 2.6, la notion de changement est floue et les exemples pertinents sont rares et difficiles à obtenir. Par conséquent, l'algorithme de détection de changements doit mettre en œuvre une modélisation asymétrique des apparences. En d'autres termes, il doit être capable de distinguer les observations contenant des changements significatifs de celles n'en contenant pas, cela en n'ayant analysé au préalable que des observations sans changements. Ces contraintes sont très proches de celles rencontrées dans le cadre de la soustraction de fond, faisant des techniques associées de bonnes candidates pour notre problème. Une adaptation est cependant nécessaire, en particulier afin d'éliminer l'oubli progressif, mentionné à la section 2.4, et pour gérer correctement les éventuelles données manquantes, dues aux occultations dans le cas de changements de points de vue.

Exploitation de la redondance D'autre part, l'utilisation de données vidéo engendre une grande redondance dans les observations considérées, qu'il est à la fois bénéfique et nécessaire de prendre en compte. En effet, ne pas prendre en compte le recouvrement des images de référence, en particulier pour la constitution d'une référence correspondant à une image de

test donnée (voir section 2.2), rendra l'approche globale lente et inefficace. Inversement, l'exploitation de ce recouvrement permet d'isoler une partie importante des calculs dans une étape hors-ligne, rendant le traitement de l'image de test nettement plus rapide. De plus, les méthodes développées dans le cadre de la soustraction de fond [67, 102] ont montré que l'exploitation de cette redondance dans la vidéo de référence, grâce à la modélisation des apparences, permet une plus grande robustesse des algorithmes vis-à-vis des fréquentes variations d'apparence (voir section 2.4). De manière similaire, les méthodes exploitant la redondance dans la vidéo de test [120] permettent de combiner les observations successives pour améliorer la précision des résultats de détection (voir section 2.5). Ainsi, l'exploitation de la redondance sous toutes ses formes est un point essentiel pour un système de détection de changements, en particulier lorsqu'il vise des données vidéo.

Représentation tri-dimensionnelle Par ailleurs, l'approche par modélisation des apparences nécessite une représentation de la scène pour l'organisation des modèles d'apparence. Or, les méthodes utilisant une représentation tri-dimensionnelle [24, 32, 34, 91] sont mieux adaptées que les méthodes bi-dimensionnelles [63, 79] au problème de la détection de changements dans des vidéos aériennes acquises selon des trajectoires arbitraires. En effet, l'approche par modélisation tri-dimensionnelle possède deux avantages majeurs. D'une part, le modèle 3D permet une gestion simplifiée et plus précise des effets géométriques (e.g. parallaxe et occultations) évoqués à la section 2.3.1. Cela débouche sur une modélisation plus fidèle des apparences de la scène, qui mène à de meilleures performances en détection de changements. D'autre part, la représentation tri-dimensionnelle de la scène utilise une structure indépendante de la vidéo de référence qui peut être établie en coordonnées réelles. Cette représentation peut donc servir pour la fusion ou l'exploitation jointe de plusieurs données de référence éventuellement hétérogènes (e.g. ensemble de vidéos aériennes et/ou d'images satellitaires). Par opposition, les représentations bi-dimensionnelles, par exemple la mosaïque [63, 79], sont souvent établies dans un système de coordonnées lié à la vidéo considérée, et sont par conséquent moins facilement ré-utilisables pour de nouvelles données. En revanche, le principal problème de l'approche tri-dimensionnelle est qu'elle nécessite souvent une connaissance précise du relief dans la scène. Nous montrons dans le chapitre 6 que l'utilisation d'une structure 3D adaptée permet de minimiser l'impact d'une connaissance approximative du relief. Nous montrons notamment qu'en milieu faiblement urbain, la donnée d'un modèle numérique de terrain (MNT) est suffisante pour l'obtention de bonne performances en détection de changements.

Gestion des variations d'illumination Nous avons vu dans la section 2.3.2 que la gestion des variations d'illumination était un problème délicat. En effet, les méthodes génératives, qui tentent de simuler les conditions d'illumination, sont rapidement limitées par le manque de réalisme ou par la complexité des modèles. Les méthodes de détection des effets de l'illumination manquent de généralité quant aux effets détectés. Enfin, les approches par représentation invariante font un compromis entre les fausses alarmes, générées par les variations d'illumination, et les non-détections, générées par l'atténuation incorrecte de changements pertinents. Par conséquent, il semble intéressant d'aborder ce problème à l'aide d'une méthode composite, qui combine plusieurs mécanismes pour améliorer la distinction entre les variations d'illumination et les changements significatifs. Par conséquent, nous avons choisi de combiner quatre mécanismes pour éliminer les fausses alarmes dues aux effets de l'illumination. Nous utilisons ainsi une technique de représentation invariante, visant à éliminer une part significative des variations d'illumination, et une technique de modélisation d'apparence, dont la capacité d'apprentissage permet d'apprendre à reconnaître les variations d'illumination résiduelles. Les deux derniers mécanismes sont l'exploitation de connaissance a priori, et plus généralement, l'utilisation d'un mécanisme de retour interactif de pertinence, afin d'éliminer les dernières fausses alarmes.

Approche semi-automatique Les approches semi-automatiques [101, 115], par opposition aux approches complètement automatiques, mettent ponctuellement l'utilisateur à contribution pour rendre les traitements plus robustes. Dans le cadre d'un système d'assistance à l'analyse, il n'est pas aberrant d'implémenter ce type d'approche, puisque l'utilisateur participe déjà à l'analyse des données. En revanche, il est essentiel de garantir que la mise à contribution de l'utilisateur ne rende pas le système lourd et fastidieux d'utilisation. Les interventions de l'utilisateur peuvent par exemple permettre d'adapter les résultats à ses besoins, ou d'améliorer la robustesse des traitements dans le cas de données difficiles. En particulier, il peut être intéressant d'employer une telle approche semi-automatique dans le cadre d'une approche composite visant à distinguer les variations non pertinentes des changements significatifs. En effet, cela autoriserait l'utilisateur à spécifier lui-même la distinction qui l'intéresse, permettant ainsi de se passer d'une base de données d'apprentissage appropriée, qui aurait été délicate à constituer et à maintenir.

Les chapitres 3 à 5 présentent les contributions techniques réalisées dans le cadre de cette thèse. Ils décrivent l'ensemble des méthodes et des algorithmes développés pour la détection de changements dans des vidéos aériennes, en prenant en compte les points mentionnés ci-dessus et en visant à exploiter au maximum le potentiel des données disponibles.

Chapitre 3

Pré-traitement des données

La notion de pré-traitement est relativement vague et peut désigner trois types d'opération. Premièrement, elle peut désigner les opérations effectuées hors-ligne et préalablement à d'autres opérations effectuées en ligne. Par exemple, dans le cas de la vidéo-surveillance, une opération en ligne, effectuée pour chaque image à mesure qu'elles sont reçues, peut consister à détecter des bagages abandonnés, tandis que les opérations hors-ligne peuvent consister à calculer au préalable un modèle 3D de la zone observée. Deuxièmement, dans le cadre de la détection de changements, la notion de pré-traitement peut également désigner les opérations effectuées sur les données de référence, par opposition à celles effectuées sur les données de test. Enfin, plus généralement, cette notion peut désigner les opérations effectuées avant d'autres opérations, implicitement considérées comme principales. C'est ce que nous entendons ici par pré-traitement, en considérant comme traitement principal les opérations dédiées à la détection de changements, qui seront abordés dans le chapitre suivant.

Le présent chapitre détaille donc les opérations de pré-traitement, étudiées dans le cadre de cette thèse, et relatives aux tâches les plus critiques parmi les opérations envisageables. Ces opérations sont de deux types : d'une part, la géo-localisation des vidéos considérées, visant à estimer les trajectoires d'acquisition et les paramètres de calibration associés, et d'autre part, la conversion des intensités des images dans une représentation invariante aux variations d'illumination. Ces opérations sont appliquées à la fois aux vidéos de référence et aux vidéos de test, à l'aide d'algorithmes adaptés aux contraintes liées à chaque type de vidéo.

La suite de ce chapitre est organisée en deux parties pour les deux types d'opération étudiés. La section 3.1 présente les algorithmes de géo-localisation des vidéos aériennes, selon que le traitement doit être effectué hors-ligne (section 3.1.1) ou en ligne (section 3.1.2). La section 3.2 présente ensuite différents algorithmes dédiés à la gestion des variations d'illumination.

Sommaire

3.1 Géo-localisation des vidéos aériennes

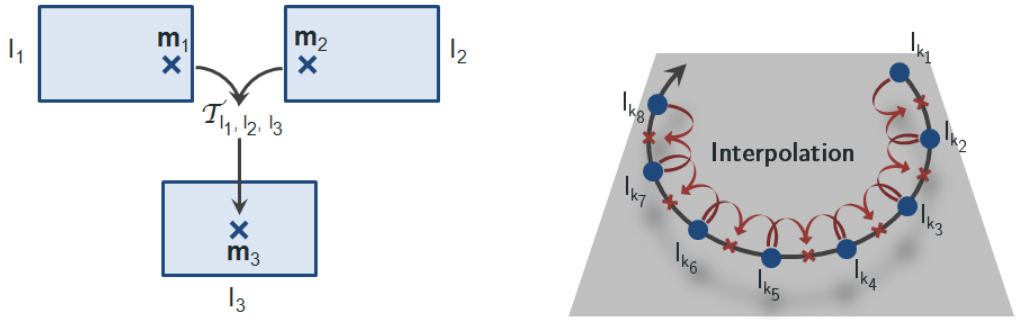
Il est crucial, pour une méthode de détection de changements basée sur l'approche par modélisation 3D des apparences, de mettre précisément en correspondance les intensités de l'image avec les modèles d'apparence. En effet, ces modèles d'apparence doivent représenter fidèlement les différentes observations des points qui leur sont associés dans la scène, afin de permettre une analyse précise des changements. Cependant, même si de nos jours une majorité des plates-formes d'observation aérienne (e.g. avions, hélicoptères, drones, ballons, etc) disposent d'une localisation GPS, ces données ne sont généralement pas suffisamment précises pour permettre une mise en correspondance directement utilisable. D'autre part, d'un point de vue plus technique, il est rare que ces données soient synchronisées avec la séquence d'images ou qu'elles contiennent les paramètres de calibration de la caméra. Or, ces paramètres de calibration ne peuvent pas toujours être estimés une fois pour toutes avant l'acquisition de la vidéo. En effet, de nombreux dispositifs d'acquisition effectuent une stabilisation de la séquence d'images, procédé permettant de modifier les paramètres de calibration en temps réel afin d'éliminer de la vidéo finale les vibrations dues au mouvement de la plate-forme.

Nous proposons donc ici deux algorithmes de géo-localisation des vidéos aériennes. Le premier, présenté à la section 3.1.1 propose un mécanisme semi-automatique utilisable uniquement de manière hors-ligne, et permettant de géo-localiser les vidéos de référence. Cet algorithme nécessite que l'utilisateur calibre manuellement quelques images de la vidéo considérée, en mettant manuellement en correspondance des points physiques de la scène avec les pixels correspondants dans les images. Ensuite, l'algorithme propage automatiquement ces annotations aux images restantes de la vidéo, à l'aide d'un mécanisme d'interpolation utilisant le tenseur trifocal [52, chapitre 15]. Le second algorithme, présenté à la section 3.1.2 permet une géo-localisation en ligne et automatique de la vidéo de test, en utilisant la technique de l'asservissement visuel. Cet algorithme exploite une technique de recalage quelconque ainsi que le modèle 3D d'apparence, qui est alors disponible, pour estimer les paramètres d'acquisition en analysant la correspondance entre l'image réelle et le modèle 3D.

3.1.1 Calibration et interpolation des paramètres d'acquisition

Une méthode classique pour estimer les paramètres d'acquisition associés à une image donnée consiste à localiser, dans l'image considérée, un certain nombre d'amers facilement repérables dans la zone observée (e.g. coins de bâtiments, objets ponctuels et inamovibles, etc). S'il est possible de déterminer la position tri-dimensionnelle de ces amers, par exemple via leurs coordonnées GPS, alors un algorithme de calibration de caméra (*camera resectioning* dans la littérature [52, algorithme 7.1]), basé sur une minimisation de l'erreur géométrique de projection, permet de retrouver les paramètres d'acquisition de l'image considérée. Ce travail de localisation manuelle d'amers dans les images, que nous désignerons dans la suite de cette section par travail d'annotation, est fastidieux mais est réalisable pour quelques images. En revanche, envisager de le faire pour toutes les images d'une vidéo est irréaliste, car cela nécessiterait un effort trop important.

Par conséquent, nous avons développé [18] un algorithme permettant de propager automatiquement les annotations déterminées manuellement sur un petit nombre d'images aux autres images d'une vidéo. Ce mécanisme de propagation utilise une interpolation basée sur l'estimation du tenseur trifocal [52, chapitre 15], qui exprime les contraintes liant trois vues quelconques d'une même scène. Ce tenseur trifocal possède la remarquable propriété de permettre la localisation très précise d'un point de la scène dans l'une des trois images, à condition de connaître les positions de ce point dans les deux autres images (voir l'illustration de la figure 3.1a). À condition d'être estimé précisément, le tenseur trifocal permet donc de propager les annotations, saisies manuellement pour deux images de la même scène, à une troisième image. Comme le



(a) **(b)**

FIGURE 3.1 – *Cette figure illustre l'utilisation du tenseur trifocal pour l'interpolation des paramètres d'acquisition des images d'une vidéo. La figure de gauche (a) illustre la propriété du tenseur trifocal, entre trois images I_1, I_2 et I_3, à permettre la localisation précise d'un point physique donné à la position $\mathbf{m}_3$ dans l'image I_3, à condition de connaître les positions $\mathbf{m}_1$ et $\mathbf{m}_2$ de ce point dans les deux autres images. L'image de droite (b) montre une application possible de cette propriété pour l'interpolation des paramètres d'acquisition. Il est ainsi possible d'utiliser les localisations d'amers, fournies manuellement pour un faible nombre d'images-clés (ronds bleus), afin de localiser automatiquement ces amers dans les autres images (croix rouges) de la vidéo considérée.*

montre la figure 3.1b, ce mécanisme peut alors être utilisé pour interpoler l'estimation des paramètres d'acquisition dans une vidéo. Pour cela, nous définissons un ensemble d'images-clés, réparties uniformément dans une vidéo donnée. Le travail d'annotation sera effectué manuellement pour ces images-clés, qui serviront ensuite à propager les annotations aux autres images de la vidéo. Le reste de cette section décrit la méthode employée en termes techniques. Le pseudo-code de cette méthode est fourni par l'algorithme 3.1.

Détails de l'algorithme d'interpolation Soit une séquence d'images désignée par $\{I_t\}_{t\in\mathbb{N}}$ et soit un ensemble restreint d'images-clés $\{I_k\}_{k\in K}$, $K \subset \mathbb{N}$. La première étape de notre méthode consiste à localiser manuellement un ensemble d'amers dans les images-clés. L'utilisateur doit donc définir un ensemble d'amers, dont les coordonnées 3D sont désignées par $\{\mathbf{M}_i\}_{i\in\mathbb{N}}$. Ces coordonnées 3D, qui peuvent être approximatives, peuvent être définies à l'aide des coordonnées GPS, mais doivent être converties en grandeurs linéaires, par opposition aux grandeurs angulaires telles que la longitude et la latitude. Par ailleurs, notons que les coordonnées GPS ainsi que l'altitude de ces amers peuvent être obtenues aisément et gratuitement à l'aide d'une variété d'outils internet [1]. Les projections $\{\mathbf{m}_{i,k}\}_{i\in\mathbb{N},k\in K}$ de ces amers, lorsqu'ils sont visibles, dans les images-clés sont ensuite définies manuellement par l'utilisateur. Il est important que ces projections des amers dans les images-clés soient définies le plus précisément possible, car leur précision influe directement sur la précision d'estimation des paramètres d'acquisition.

À l'aide des correspondances entre points 3D et projections 2D, il est alors possible de calculer une première estimation des paramètres d'acquisition des images-clés, à l'aide de l'algorithme de calibration de caméra [52, algorithme 7.1]. Les coordonnées 3D des amers pouvant être approximatives et leurs projections, issues d'une annotation manuelle, pouvant être légèrement incohérentes, cette première estimation des paramètres d'acquisition est ensuite affinée par la méthode d'ajustement de faisceaux (*bundle adjustment* dans la littérature, [52, § 18.1]). Nous obtenons donc finalement une estimation des matrices de projection $\{\widehat{\mathrm{P}}_k\}_{k\in K}$, contenant

1. Google Maps et son API fournissent ainsi un moyen rapide d'extraire les coordonnées GPS et l'altitude d'un point donné. Le site `www.daftlogic.com/sandbox-google-maps-find-altitude.htm` propose une implémentation expérimentale de ces fonctionnalités. Le site `www.gpsvisualizer.com/convert` permet également d'obtenir les altitudes correspondant à des coordonnées GPS données.

Entrées : Ensemble d'images $\{I_t\}_{t\in\mathbb{N}}$, indices des images-clés $K \subset \mathbb{N}$ et ensemble d'amers $\{\mathbf{M}_i\}_{i\in\mathbb{N}}$
Sorties : Matrices de projection $\widehat{\mathrm{P}_n}$ pour chaque image I_n (clé ou non-clé)

1: **Pour** $k \in K$ **Faire** ▷ Étape d'Annotation
2: Localiser les projections $\{\mathbf{m}_{i,k}\}_{i\in\mathbb{N}}$ des amers dans l'image-clé I_k
3: Estimer la matrice de projection $\widehat{\mathrm{P}_k}$ associée à l'image-clé I_k ▷ [52], Alg 7.1
4: **Fin Pour**
5: Affiner les matrices de projection $\{\widehat{\mathrm{P}_k}\}_{k\in K}$ par ajustement de faisceaux ▷ [52], § 18.1
. . . . . .
6: **Pour** $t \in \mathbb{N}\backslash K$ **Faire** ▷ Étape d'Interpolation
7: $k_1 \leftarrow \sup\{k \in K \mid k < t\}$
8: $k_2 \leftarrow \inf\{k \in K \mid k > t\}$
9: Déterminer les correspondances entre les points SIFT extraits de I_t, I_{k_1} et I_{k_2}
10: Estimer le tenseur trifocal $\widehat{\mathcal{T}}_{t,k_1,k_2}$ à l'aide des correspondances entre les 3 vues ▷ [52], Alg 16.4
11: Trouver les projections $\{\mathbf{m}_{i,t}\}_{i\in\mathbb{N}}$ des amers, transférées de I_{k_1} et I_{k_2} vers I_t ▷ [52], § 15.3.2
12: Estimer la matrice de projection $\widehat{\mathrm{P}_t}$ par l'algorithme de calibration ▷ [52], Alg 7.1
13: **Fin Pour**

ALGORITHME 3.1 – *Algorithme hors-ligne d'interpolation des paramètres d'acquisition associés aux images d'une vidéo.*

la position, l'orientation et les paramètres de calibration, pour l'ensemble des images-clés. Une factorisation QR permet alors d'extraire la matrice de calibration de cette matrice de projection, afin de séparer les paramètres d'acquisition intrinsèques et extrinsèques.

La dernière étape consiste alors à estimer les matrices de projections pour les images non-clé $\{I_t\}_{t\in\mathbb{N}\backslash K}$. Étant donnée une image non-clé I_t, $t \in \mathbb{N}\backslash K$, les images-clés I_{k_1} et I_{k_2}, encadrant au plus près l'image I_t dans la séquence considérée, sont sélectionnées. Plus formellement, la sélection des images-clés I_{k_1} et I_{k_2} est telle que :

$$\begin{aligned} k_1 &= \sup\{k \in K \mid k < t\} \\ k_2 &= \inf\{k \in K \mid k > t\} \end{aligned} \tag{3.1}$$

Une méthode alternative pour la sélection des images-clés pourrait consister à analyser le contenu des images pour trouver celles correspondant le mieux à l'image non-clé considérée. Cependant, la méthode exposée ci-dessus est nettement plus rapide et donne de bons résultats lorsque les images-clés sont suffisamment nombreuses (typiquement, 1% du nombre total d'images). Une fois les images-clés sélectionnées, le tenseur trifocal $\widehat{\mathcal{T}}_{t,k_1,k_2}$ entre les trois images I_t, I_{k_1} et I_{k_2} est alors automatiquement estimé à partir d'appariements de points SIFT [69], à l'aide de l'algorithme RANSAC+GoldStandard [52, algorithme 16.4]. Nous avons mentionné la propriété du tenseur trifocal $\widehat{\mathcal{T}}_{t,k_1,k_2}$ à permettre le transfert précis de points correspondants de I_{k_1} et I_{k_2} vers I_t. Il est donc possible d'estimer les projections $\{\widehat{\mathbf{m}}_{i,t}\}_{i\in\mathbb{N}}$ des amers dans l'image I_t, grâce à la connaissance du tenseur $\widehat{\mathcal{T}}_{t,k_1,k_2}$ et des projections $\{\mathbf{m}_{i,k_1}\}_{i\in\mathbb{N}}$ et $\{\mathbf{m}_{i,k_2}\}_{i\in\mathbb{N}}$ dans I_{k_1} et I_{k_2}. Disposant à nouveau de correspondances entre points 3D et projections 2D dans l'image I_t, il est alors possible d'estimer la matrice de projection $\widehat{\mathrm{P}_n}$ associée, à l'aide de l'algorithme de calibration [52, algorithme 7.1]. Le même mécanisme peut être appliqué à toutes les images non-clé, menant à l'estimation des paramètres d'acquisition pour chaque image de la séquence.

Il faut noter ici que la précision d'estimation du tenseur trifocal est cruciale pour garantir que les paramètres d'acquisition estimés pour l'image I_t soient corrects et utilisables. Or, cette précision dépend de la qualité et du nombre des appariements de points SIFT dans les trois images I_t, I_{k_1} et I_{k_2}, qui dépendent à leur tour de la similarité entre ces trois images. Cela sous-entend que, pour maximiser la précision de l'interpolation des paramètres d'acquisition, ces trois images doivent être les plus proches possibles dans la séquence d'images. Ce raisonnement

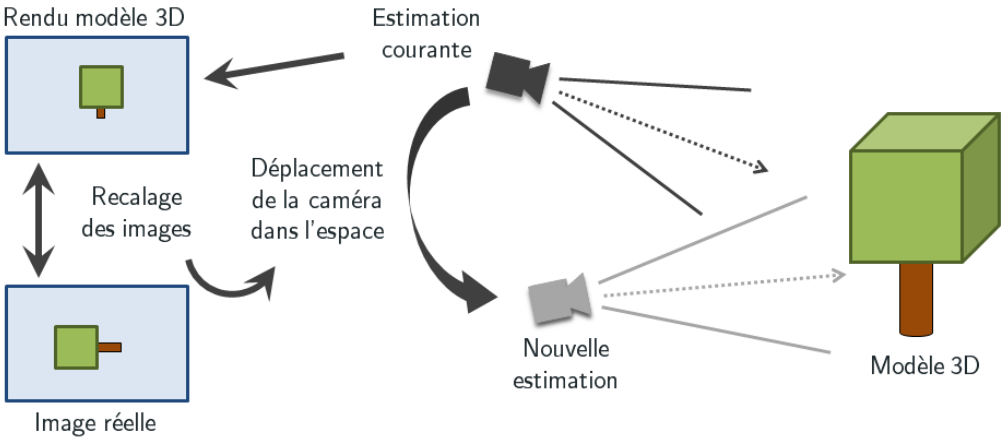


FIGURE 3.2 – *Cette figure illustre le principe de l'asservissement visuel. Pour commencer, un rendu du modèle 3D est effectué selon l'estimation courante des paramètres d'acquisition. Le recalage entre ce rendu et l'image réelle est ensuite utilisé pour déplacer la caméra dans l'espace, résultant en une nouvelle estimation des paramètres d'acquisition. L'itération de ce mécanisme permet d'obtenir une estimation précise des paramètres d'acquisition de l'image considérée.*

débouche sur la conclusion, qui correspond à l'intuition, qu'asymptotiquement, la précision maximale est obtenue lorsque la totalité des images de la séquence sont calibrées manuellement. C'est bien sûr irréalisable en pratique, et cela mène ainsi à un compromis à trouver entre effort d'annotation et précision des estimations.

D'autre part, la complexité algorithmique de cette méthode d'estimation des paramètres d'acquisition est linéaire avec le nombre d'images à traiter. En pratique, elle est toutefois relativement lente, le temps de traitement pour une image donnée étant dominé par l'algorithme d'estimation du tenseur trifocal, qui est lui-même relativement lent.

Ainsi, cette méthode nécessite que l'ensemble des images de la séquence soient disponibles, mais elle n'est en revanche basée que sur de l'information fournie par l'utilisateur. Par conséquent, elle est tout à fait adaptée au traitement des vidéos de référence, mais n'est pas applicable pour la vidéo de test, qui doit être traitée de manière incrémentale. Les résultats d'évaluation de cette méthode sont présentés et analysés à la section 6.2.1.

3.1.2 Asservissement visuel des paramètres d'acquisition

Afin de pouvoir estimer les paramètres d'acquisition d'une vidéo de manière incrémentale, c'est-à-dire à mesure que ses images sont acquises, il est préférable de ne pas employer la technique de calibration basée sur des correspondances entre points 3D et projections 2D. En effet, pour atteindre un bon niveau de précision, cette méthode requiert soit un important effort d'annotation manuelle pour chaque image, qui rendrait l'approche globale inefficace, soit un modèle tri-dimensionnel extrêmement précis de la scène, ce qui constitue une hypothèse peu réaliste dans le cadre de l'observation aérienne. Cependant, un modèle tri-dimensionnel approximatif de la scène peut suffire à estimer de manière très précise les paramètres d'acquisition d'une image donnée, à l'aide de la technique d'asservissement visuel.

L'idée de cette méthode est de guider l'estimation des paramètres d'acquisition, à l'aide du recalage entre l'image considérée et le rendu d'un modèle 3D de la scène, généré grâce à l'estimation courante des paramètres d'acquisition. En d'autres termes, les paramètres d'acquisition sont ajustés itérativement jusqu'à ce que le modèle 3D de la scène soit parfaitement aligné avec l'image considérée (voir figure 3.2). En pratique, cette méthode est constituée de deux étapes. La première étape consiste à obtenir une estimation grossière des paramètres d'acquisition de

l'image considérée. Pour cela, nous avons développé [15] un algorithme de prédiction, qui estime les paramètres d'acquisition de l'image courante en exploitant ceux estimés pour l'image précédente. La seconde étape consiste à corriger l'estimation des paramètres d'acquisition de l'image courante, en exploitant un modèle 3D de la scène observée. Pour cela, nous utilisons un algorithme de correction guidé par une matrice Jacobienne particulière [15], dont la dérivation est présentée en annexe.

Le reste de cette section décrit la méthode employée en termes techniques. Le pseudo-code de cette méthode, appliquée à l'estimation des paramètres d'acquisition d'une séquence d'images, est fourni par l'algorithme 3.2.

Notations dans le cas restreint Pour commencer, nous définissons les notations communes aux deux étapes de la méthode, dans le cas restreint où les paramètres intrinsèques de calibration (distances focales et coordonnées 2D du point d'intersection entre l'axe optique et le plan image de la caméra, également appelé point principal) sont supposés connus. Soient deux caméras $\mathcal{C}_1$ et $\mathcal{C}_2$, associées aux images I_1 et I_2, observant une scène contenant un plan π (par exemple le plan dominant du sol), caractérisé par sa normale $\mathbf{n}_\pi$ et sa distance à l'origine d_π. Nous désignons par $\mathbf{x}_1^e$ et $\mathbf{x}_2^e$ les vecteurs contenant les paramètres extrinsèques (position et orientation) de chaque caméra, et par $\mathbf{x}_1^i$ et $\mathbf{x}_2^i$ les vecteurs contenant les paramètres intrinsèques de calibration. Nous désignons alors par $\mathbf{x}_1 = [\mathbf{x}_1^e\ \mathbf{x}_1^i]$ et $\mathbf{x}_2 = [\mathbf{x}_2^e\ \mathbf{x}_2^i]$ la concaténation des paramètres extrinsèques et intrinsèques pour chaque caméra. Enfin, nous désignons par $d\mathbf{x}^e$ les paramètres extrinsèques de la caméra $\mathcal{C}_2$, exprimés dans le système de coordonnées défini par $\mathbf{x}_1$. En d'autres termes, $d\mathbf{x}^e$ représente le changement de position et d'orientation subi par $\mathcal{C}_2$ par rapport à la position et à l'orientation de $\mathcal{C}_1$. Ces vecteurs sont exprimés comme suit :

$$\begin{aligned}
\mathbf{x}_1^e &= \begin{bmatrix} \psi_1 & \theta_1 & \phi_1 & x_1 & y_1 & z_1 \end{bmatrix}^T \\
\mathbf{x}_2^e &= \begin{bmatrix} \psi_2 & \theta_2 & \phi_2 & x_2 & y_2 & z_2 \end{bmatrix}^T \\
d\mathbf{x}^e &= \begin{bmatrix} d\psi & d\theta & d\phi & dx & dy & dz \end{bmatrix}^T \\
\mathbf{x}_1^i &= \begin{bmatrix} fx_1 & fy_1 & ox_1 & oy_1 \end{bmatrix}^T \\
\mathbf{x}_2^i &= \begin{bmatrix} fx_2 & fy_2 & ox_2 & oy_2 \end{bmatrix}^T
\end{aligned} \tag{3.2}$$

Dans ces expressions, $(\psi_1, \theta_1, \phi_1)$, $(\psi_2, \theta_2, \phi_2)$ et $(d\psi, d\theta, d\phi)$ représentent les orientations selon la convention ZYX d'Euler (encore appelée convention de Tait-Bryan, fréquemment utilisée pour représenter les orientation de véhicules aériens). (x_1, y_1, z_1), (x_2, y_2, z_2) et (dx, dy, dz) représentent les positions linéaires 3D. Enfin, (fx_1, fy_1) et (fx_2, fy_2) représentent les distances focales horizontales et verticales, et (ox_1, oy_1) et (ox_2, oy_2) représentent les coordonnées des points principaux.

Soit $\mathrm{H}_{2\leftarrow 1}$ l'homographie recalant l'image I_2 par rapport à l'image I_1 selon le plan π. La notation $2 \leftarrow 1$ est utilisée ici pour rappeler que $\mathrm{H}_{2\leftarrow 1}$ transforme les coordonnées 2D de l'image I_1 vers les coordonnées 2D de l'image I_2. Par ailleurs, notons que dans le reste de cette section, toutes les homographies considérées effectueront un recalage selon le plan π, même si cela n'est pas mentionné explicitement. Il a été démontré dans [52, § 13.1] que $\mathrm{H}_{2\leftarrow 1}$ pouvait être exprimée analytiquement en fonction de $\mathbf{n}_\pi$, d_π et des matrices de projections des deux images. Cependant, l'expression associée ne fait pas apparaître clairement les paramètres d'acquisition de chaque image. Nous avons donc déterminé une expression analytique de l'homographie $\mathrm{H}_{2\leftarrow 1}$ recalant I_2 par rapport à I_1, que nous noterons $\mathrm{H}^f_{2\leftarrow 1}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{n}_\pi, d_\pi)$, en fonction des paramètres d'acquisition des deux images (voir l'expression analytique démontrée en annexe, section A.1.1). Dans la suite, nous désignerons par $\mathrm{vec}(\cdot)$ la fonction transformant une matrice 3×3 en un vecteur colonne [2] contenant tous les éléments de la matrice sauf l'élément constant d'indice $(3,3)$. Alors, sous l'hypothèse que les paramètres d'acquisition $\mathbf{x}_1$ et $\mathbf{x}_2$ sont

2. Ce vecteur colonne n'est composé que de 8 lignes, du fait que dans la représentation canonique des homographies, le neuvième élément est constant égal à 1 et qu'il n'y a donc pas lieu de le prendre en compte.

proches, il est possible de linéariser l'expression de vec $\left(\mathrm{H}^{f}_{2\leftarrow 1}\right)$ par rapport à $d\mathbf{x}$:

$$\mathrm{vec}\left(\mathbf{H}^{f}_{2\leftarrow 1}(\mathbf{x}_1,\mathbf{x}_2,\mathbf{n}_\pi,d_\pi)\right) \approx \mathbf{h}_{Id}(\mathbf{x}^i_1,\mathbf{x}^i_2) + \mathrm{J}_H(\mathbf{x}_1,\mathbf{x}^i_2,\mathbf{n}_\pi,d_\pi)\cdot d\mathbf{x} \tag{3.3}$$

Les expressions analytiques du terme constant de la linéarisation $\mathbf{h}_{Id}(\mathbf{x}^i_1,\mathbf{x}^i_2)$ ainsi que de la matrice jacobienne $\mathrm{J}_H(\mathbf{x}_1,\mathbf{x}^i_2,\mathbf{n}_\pi,d_\pi)$ sont données en annexe, section A.1.2.

Notations dans le cas général Dans le cas général, les paramètres de calibrations ne sont pas connus et peuvent également être estimés par la méthode d'asservissement visuel. Nous désignons toujours par $\mathbf{x}_1$ et $\mathbf{x}_2$ les vecteurs contenant l'ensemble des paramètres d'acquisition (position et orientation, distances focales et coordonnées du point principal) de chaque caméra. Ces vecteurs sont exprimés comme suit :

$$\begin{aligned}\mathbf{x}_1 &= \begin{bmatrix} \psi_1 & \theta_1 & \phi_1 & x_1 & y_1 & z_1 & fx_1 & fy_1 & ox_1 & oy_1 \end{bmatrix}^T \\ \mathbf{x}_2 &= \begin{bmatrix} \psi_2 & \theta_2 & \phi_2 & x_2 & y_2 & z_2 & fx_2 & fy_2 & ox_2 & oy_2 \end{bmatrix}^T\end{aligned} \tag{3.4}$$

Les quantités scalaires intervenant dans ces expressions sont définies de la même manière que pour le cas restreint. Sous l'hypothèse que les paramètres d'acquisition $\mathbf{x}_1$ et $\mathbf{x}_2$ sont proches, il est possible de linéariser vec $\left(\mathrm{H}^{f}_{2\leftarrow 1}\right)$ par rapport à $d\mathbf{x}$:

$$\mathrm{vec}\left(\mathrm{H}^{f}_{2\leftarrow 1}(\mathbf{x}_1,\mathbf{x}_2,\mathbf{n}_\pi,d_\pi)\right) \approx \mathrm{vec}(\mathrm{I_D}) + \mathrm{J}_H(\mathbf{x}_1,\mathbf{n}_\pi,d_\pi)\cdot d\mathbf{x} \tag{3.5}$$

où $\mathrm{I_D}$ représente la matrice identité. L'expression analytique de la nouvelle matrice jacobienne $\mathrm{J}_H(\mathbf{x}_1,\mathbf{n}_\pi,d_\pi)$ est donnée en annexe, table A.2.

Il est intéressant de remarquer que les équations 3.3 et 3.5 définissent une relation entre d'une part, les variations de poses $d\mathbf{x}$ entre les caméras $\mathcal{C}_1$ et $\mathcal{C}_2$, et d'autre part, l'homographie de recalage $\mathrm{H}_{2\leftarrow 1}$ de I_2 par rapport à I_1. En d'autres termes, connaissant les variations de paramètres d'acquisition entre deux caméras, il est possible de calculer de manière très précise l'homographie de recalage entre les images correspondantes. Mieux, il est à l'inverse possible, connaissant l'homographie de recalage entre les deux images, d'estimer les variations de paramètres d'acquisition entre les deux caméras. Notons toutefois que cela nécessite la connaissance des paramètres du plan dominant π, qui peuvent être calculés si un modèle 3D de la scène est connu. Cette relation est au cœur de notre méthode d'asservissement visuel, dont les détails des étapes de prédiction et de correction sont donnés ci-dessous.

Prédiction Considérons maintenant que les paramètres d'acquisition $\mathbf{x}_1$ de la caméra $\mathcal{C}_1$ sont connus et cherchons à estimer ceux de la caméra $\mathcal{C}_2$. Les deux images étant disponibles, il est possible d'estimer l'homographie $\widehat{\mathrm{H}}_{2\leftarrow 1}$ recalant I_2 par rapport à I_1 à l'aide d'un algorithme de recalage [123]. En pratique, nous estimons cette homographie à l'aide d'un algorithme RANSAC [45] appliqué sur des appariements de points SURF [5] extraits des deux images [3]. Nous souhaitons alors déterminer la variation de paramètres d'acquisition $d\mathbf{x}$ expliquant au mieux l'homographie mesurée $\widehat{\mathrm{H}}_{2\leftarrow 1}$. Cela revient à estimer les paramètres d'acquisition $\widehat{\mathbf{x}}_2$ pour lesquels la fonction $\mathbf{x} \mapsto \mathrm{vec}\left(\mathrm{H}^{f}_{2\leftarrow 1}(\mathbf{x}_1,\mathbf{x},\mathbf{n}_\pi,d_\pi)\right) - \mathrm{vec}\left(\widehat{\mathrm{H}}_{2\leftarrow 1}\right)$ s'annule. Ce problème peut donc être résolu grâce à la méthode de Newton pour les fonctions non-linéaires à plusieurs variables, algorithme d'estimation itératif mis en œuvre dans l'étape Prédiction de l'algorithme 3.2.

Cette étape de prédiction est très rapide et est en pratique limitée par le temps nécessaire pour mesurer l'homographie de recalage $\widehat{\mathrm{H}}_{2\leftarrow 1}$ entre les deux images. En revanche, l'erreur d'estimation dépend directement de la précision avec laquelle les paramètres d'acquisition $\mathbf{x}_1$

3. Les points SURF sont des points d'intérêt similaires aux points SIFT, bien que moins discriminants, mais qui présentent l'avantage de pouvoir être extraits plus rapidement, ce qui est souhaitable pour une méthode de traitement en ligne.

Entrées : Images courante I_n et précédente I_{n-1}, estimation des paramètres d'acquisition précédents $\widehat{\mathbf{x}}_{n-1}$ et modèle 3D de la scène, choix entre méthode de correction rapide ou précise
Sorties : Estimation des paramètres d'acquisition courants $\widehat{\mathbf{x}}_n$

Déterminer les paramètres $\mathbf{n}_\pi$ et d_π du plan dominant dans le modèle 3D par moindres carrés
. . . . . .
Trouver l'homographie $\widehat{\mathrm{H}}_{n\leftarrow n-1}$ recalant I_n par rapport à I_{n-1} ▷ Étape de Prédiction
$\mathbf{x} \leftarrow \widehat{\mathbf{x}}_{n-1}$
Faire
    Calculer $\mathrm{H}^f_{n\leftarrow n-1}(\widehat{\mathbf{x}}_{n-1}, \mathbf{x}, \mathbf{n}_\pi, d_\pi)$
    Résoudre (SVD) par rapport à $d\mathbf{x}$: $J_H(\widehat{\mathbf{x}}_{n-1}, \mathbf{n}_\pi, d_\pi) \cdot d\mathbf{x} = \mathrm{vec}\left(\widehat{\mathrm{H}}_{n\leftarrow n-1}\right) - \mathrm{vec}\left(\mathrm{H}^f_{n\leftarrow n-1}(\widehat{\mathbf{x}}_{n-1}, \mathbf{x}, \mathbf{n}_\pi, d_\pi)\right)$
    Mettre à jour $\mathbf{x}$ grâce au décalage $d\mathbf{x}$ (accumulation des paramètres linéaires, composition des matrices de rotation)
Jusqu'à convergence ($d\mathbf{x}$ proche de zéro)
$\widetilde{\mathbf{x}}_n \leftarrow \mathbf{x}$
. . . . . .
$\mathbf{x} \leftarrow \widetilde{\mathbf{x}}_n$ ▷ Étape de Correction
Si Méthode de correction précise **Alors**
    Faire
        Générer l'image $I_r(\mathbf{x})$ de rendu du modèle 3D selon $\mathbf{x}$
        Trouver l'homographie $\widehat{\mathrm{H}}_{n\leftarrow r}(\mathbf{x})$ recalant I_n par rapport à $I_r(\mathbf{x})$
        Résoudre (SVD) par rapport à $d\mathbf{x}$: $J_H(\mathbf{x}, \mathbf{n}_\pi, d_\pi) \cdot d\mathbf{x} = \mathrm{vec}\left(\mathrm{I_D}\right) - \mathrm{vec}\left(\widehat{\mathrm{H}}_{r\leftarrow n}(\mathbf{x})\right)$
        Mettre à jour $\mathbf{x}$ grâce au décalage $d\mathbf{x}$ (accumulation des paramètres linéaires, composition des matrices de rotation)
    Jusqu'à convergence ($d\mathbf{x}$ proche de zéro)
Sinon
    Générer l'image $I_r(\widetilde{\mathbf{x}}_n)$ de rendu du modèle 3D selon $\widetilde{\mathbf{x}}_n$
    Trouver l'homographie $\widehat{\mathrm{H}}_{n\leftarrow r}(\widetilde{\mathbf{x}}_n)$ recalant I_n par rapport à $I_r(\widetilde{\mathbf{x}}_n)$
    Faire
        Calculer $\mathrm{H}^f_{n\leftarrow r}(\widetilde{\mathbf{x}}_n, \mathbf{x}, \mathbf{n}_\pi, d_\pi)$
        Résoudre (SVD) par rapport à $d\mathbf{x}$: $J_H(\widetilde{\mathbf{x}}_n, \mathbf{n}_\pi, d_\pi) \cdot d\mathbf{x} = \mathrm{vec}\left(\widehat{\mathrm{H}}_{n\leftarrow r}(\widetilde{\mathbf{x}}_n)\right) - \mathrm{vec}\left(\mathrm{H}^f_{n\leftarrow r}(\widetilde{\mathbf{x}}_n, \mathbf{x}, \mathbf{n}_\pi, d_\pi)\right)$
        Mettre à jour $\mathbf{x}$ grâce au décalage $d\mathbf{x}$ (accumulation des paramètres linéaires, composition des matrices de rotation)
    Jusqu'à convergence ($d\mathbf{x}$ proche de zéro)
Fin Si
$\widehat{\mathbf{x}}_n \leftarrow \mathbf{x}$

ALGORITHME 3.2 – *Algorithme d'asservissement visuel pour l'estimation incrémentale des paramètres d'acquisition des images d'une vidéo, à l'aide d'un modèle 3D de la scène.*

sont connus. Or, dans la plupart des cas, nous ne disposerons que d'une estimation approximative $\widehat{\mathbf{x}}_1$ de ces paramètres. Par conséquent, dans le cas d'une utilisation séquentielle de cet algorithme, par exemple lors de l'estimation des paramètres d'acquisition de chaque image d'une vidéo, ce problème risque d'engendrer une accumulation de l'erreur d'estimation.

Correction Pour éviter cette accumulation d'erreurs, nous utilisons l'approche basée sur l'asservissement visuel pour corriger la prédiction des paramètres d'acquisition à l'aide du modèle 3D d'apparence, qui sert alors de référence pour la localisation. Soit $\widetilde{\mathbf{x}}_2$ une estimation grossière des paramètres d'acquisition de l'image I_2, en pratique fournie par l'étape de prédiction ci-dessus. Désignons par $I_r(\mathbf{x})$ l'image obtenue par rendu du modèle 3D selon les paramètres d'acquisition $\mathbf{x}$, et soit $\widehat{\mathrm{H}}_{r\leftarrow 2}(\mathbf{x})$ l'homographie recalant l'image $I_r(\mathbf{x})$ par rapport à l'image I_2, estimée par la méthode évoquée plus haut (points SURF [5] + RANSAC [45]). Nous souhaitons déterminer les paramètres d'acquisition $\mathbf{x}_2$ tels que l'homographie recalant l'image réelle I_2 par rapport à l'image $I_r(\mathbf{x}_2)$, rendue selon $\mathbf{x}_2$, soit égale à la matrice identité. Cela revient à estimer les paramètres d'acquisition $\widehat{\mathbf{x}}_2$ pour lesquels la fonction $\mathbf{x} \mapsto \mathrm{vec}\left(\mathrm{H}^f_{r\leftarrow 2}(\mathbf{x}_2, \mathbf{x}, \mathbf{n}_\pi, d_\pi)\right) -$

vec $\left(\text{ID}\right)$ s'annule. Cependant, l'évaluation de cette fonction nécessite la connaissance des vrais paramètres d'acquisition $\mathbf{x}_2$, que nous cherchons à déterminer. Par conséquent, nous remplaçons l'expression analytique $\mathrm{H}^{f}_{r\leftarrow 2}(\mathbf{x}_2, \mathbf{x}, \mathbf{n}_\pi, d_\pi)$ par l'homographie $\hat{\mathrm{H}}_{r\leftarrow 2}(\mathbf{x})$ mesurée empiriquement, ce qui est possible à condition que le plan π décrive effectivement le plan dominant dans la scène. Comme précédemment, ce problème peut alors être résolu grâce à la méthode de Newton pour les fonctions non-linéaires à plusieurs variables, dont l'algorithme est mis en œuvre dans l'étape Correction de l'algorithme 3.2.

Remarquons que cette étape de correction est assez lente puisqu'elle nécessite à chaque itération un rendu du modèle 3D et une estimation de l'homographie de recalage entre I_2 et I_r. Néanmoins, en pratique, l'étape de prédiction employée au préalable permet de trouver une valeur d'initialisation très proche de la valeur optimale et par conséquent, le nombre d'itérations requises pour la convergence de l'étape de correction est généralement faible. Toutefois, une approche alternative, plus rapide mais moins précise, peut consister à appliquer une seconde fois l'étape de prédiction, non plus à l'aide de l'image précédente mais à l'aide cette fois d'un rendu du modèle 3D, généré depuis l'estimation du point de vue courant (voir l'algorithme 3.2). Dans tous les cas, il est crucial d'appliquer une correction à l'estimation issue de l'étape de prédiction, afin d'éviter la divergence de l'erreur associée.

Pour finir, notons que les étapes de prédiction et de correction présentées ci-dessus reposent sur la résolution d'un système linéaire d'équations, qui est sur-contraint, dans le cas restreint où les paramètres de calibrations sont connus, et sous-contraint, dans le cas général où ces paramètres sont inconnus. Par conséquent, dans les deux cas, la résolution est effectuée de manière approchée à l'aide de la méthode par décomposition en valeurs singulières (SVD). Cette méthode de résolution permet d'obtenir le décalage $d\mathbf{x}$ approchant au mieux les contraintes exprimées par le système linéaire, au sens des moindres carrés. Dans le cas restreint, où le système est sur-contraint, ceci permet d'obtenir la solution permettant de faire le meilleur compromis entre toutes les contraintes disponibles. Dans le cas général, où le système est sous contraint, ceci permet de trouver le vecteur $d\mathbf{x}$ de norme minimale permettant de satisfaire les contraintes disponibles. Cependant dans ce dernier cas, le nombre insuffisant de contraintes peut avoir pour conséquence de faire diverger l'estimation de l'algorithme d'asservissement visuel. En pratique, nous avons pu limiter cette divergence en effectuant un amortissement sur les paramètres de calibration [4].

Comme pour l'algorithme de géo-localisation présenté à la section précédente, cette méthode par asservissement visuel a une complexité algorithmique linéaire par rapport au nombre d'image à traiter. Cependant, elle est un peu plus rapide en pratique, ce qui est intéressant pour le traitement en ligne de la vidéo de test. Les résultats d'évaluation, relatifs à la précision d'estimation de cette méthode, sont présentés et analysés à la section 6.2.1.

3.2 Invariance aux variations de l'illumination

Nous avons vu dans le chapitre 2 que les variations d'illumination pouvaient entraîner un grand nombre de faux positifs. Pour éviter cela, nous avons choisi de combiner différentes techniques permettant d'atténuer l'impact de ces variations d'illumination. La première de ces technique intervient en pré-traitement des données et consiste à convertir les observations dans un espace d'intensités invariant aux effets de l'illumination. Les autres techniques employées pour cela par notre approche de détection de changements seront présentées dans les chapitres suivants.

4. Cet amortissement est mis en œuvre en introduisant un facteur d'atténuation dans la mise à jour des paramètres intrinsèques de calibration (lignes 16 et 24 de l'algorithme 3.2) : $f = f + \zeta \cdot df$, avec $\zeta \in]0,1[$.

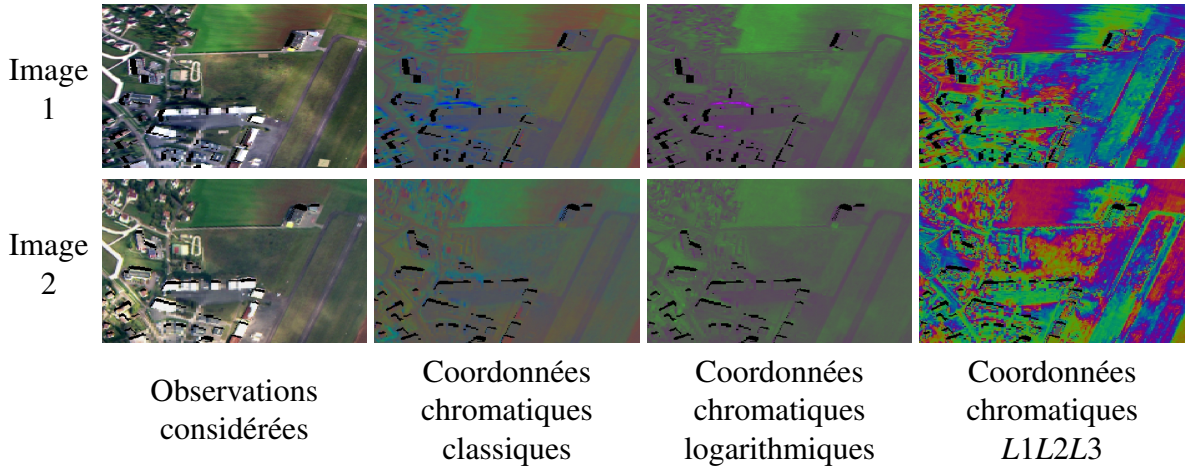


FIGURE 3.3 – *Cette figure illustre les techniques de représentation invariante aux variations d'illumination, et montre la représentation brute associée à deux images acquises sous des conditions d'illumination différentes. La première colonne montre les observations initiales considérées. Les seconde, troisième et quatrième colonnes montrent les représentations brutes associées à ces deux images et obtenues respectivement à l'aide des coordonnées chromatiques classiques, des coordonnées chromatiques logarithmiques et des coordonnées chromatiques L1L2L3.*

Le choix d'appliquer un pré-traitement consistant à transformer les observations dans un espace invariant aux effets de l'illumination est justifié par les raisons suivantes. D'une part, bien que ces méthodes de représentation invariante ne soient pas parfaites (compromis entre faux positifs et faux négatifs), elles ont l'avantage d'être extrêmement flexibles et malgré tout assez efficaces. D'autre part, notre objectif final étant la détection de changements et non la détection des effets de l'illumination, nos travaux se sont naturellement concentrés sur l'étude des méthodes permettant de comparer des données d'observation selon des conditions d'illumination normalisées. Ce type d'approches présente en effet l'intérêt de permettre la comparaison des vidéos de référence avec celles de test, et cela sans se préoccuper de localiser les effets de l'illumination avec précision. Pour cela, nous avons cherché à exploiter les multiples observations disponibles dans les vidéos de référence, afin de déterminer des conditions d'illumination normalisées. Ces considérations ont mené vers la notion de chromaticité, qui a été employée par un certain nombre de méthodes [38, 44, 47] pour représenter des observations de manière invariante par rapport aux variations d'illumination.

Dans cette section, nous présentons donc différentes méthodes permettant l'atténuation des effets de l'illumination. La section 3.2.1 décrit les différentes représentations employées pour éliminer les conditions d'illumination des observations considérées. Ces représentations sont basées sur la notion de chromaticité, qu'il est possible de définir de différentes manières. Les sections suivantes présentent en détail ces différentes mesures de la chromaticité. La section 3.2.2 présente la technique basée sur les coordonnées chromatiques classiques. La section 3.2.3 présente la technique, introduite par Finlayson et al. [44], basée sur les coordonnées chromatiques logarithmiques. Enfin, la section 3.2.4 décrit la technique basée sur les coordonnées chromatiques $L1L2L3$, qui ont été introduites par Gevers et Smeulders [47].

3.2.1 Représentations invariantes

La chromaticité d'une observation donnée est une grandeur qualifiant la couleur de cette observation de manière indépendante de sa luminosité. Il existe un grand nombre de mesures permettant de quantifier la chromaticité d'une observation, comme par exemple le couple Teinte / Saturation de l'espace TSV (*HSV* dans la littérature), le couple (a,b) de l'espace CIELAB ou encore le couple (u,v) de l'espace CIELUV. Il est souvent suffisant de représenter la chroma-

ticité à l'aide de deux variables indépendantes. Cependant pour une meilleure correspondance vis-à-vis des illustrations affichées en trois couleurs, nous désignerons par un triplet (r,g,b) (en minuscules) la chromaticité d'une observation (R,G,B), où la troisième coordonnée chromatique peut être directement exprimée en fonction des deux premières.

Représentation brute Les trois techniques de mesure de la chromaticité, présentées plus en détail dans les sections 3.2.2 à 3.2.4, exploitent des rapports entre les couleurs observées afin d'ignorer la luminosité de l'observation, dans le but final d'aboutir à une représentation invariante aux variations d'illumination. L'élimination de la luminosité est une bonne chose pour garantir l'invariance par rapport aux conditions d'illumination, cependant, cela a pour conséquence indésirable de générer des faux négatifs. En effet, la luminosité d'une observation n'est pas uniquement liée à l'illumination de la scène. Par exemple, en considérant un objet blanc et un objet gris côte-à-côte et subissant donc la même illumination, les observations issues de l'objet blanc seront toujours plus lumineuses que celles issues de l'objet gris, et ce, quelque soit l'illumination de la scène. En d'autres termes, le fait d'ignorer la luminosité des observations implique la perte d'une partie de l'information liée à la scène, ce qui peut générer des faux négatifs.

La figure 3.3 présente les résultats obtenus en appliquant, sur deux images acquises sous des conditions d'illumination différentes, chacune des mesures de chromaticité décrites dans les sections suivantes. Ces résultats montrent que la représentation brute obtenue à l'aide de chacune de ces techniques permet bien d'atténuer les variations d'illumination. Cependant, une bonne partie de l'information liée à la scène a également disparu. Par exemple, dans ces représentations, les bâtiments blancs sont indistinguables des pistes goudronnées situées à proximité. Cette perte d'information peut avoir des conséquences néfastes sur les performances de détection de changements, et peut en particulier générer des faux négatifs.

Représentation normalisée Afin de minimiser l'impact de cette perte d'information, nous avons proposé [18], pour un point physique donné, de transformer à nouveau le résultat de ces méthodes de représentation invariante, en exploitant l'observation moyenne $(\overline{R}_{ref},\overline{G}_{ref},\overline{B}_{ref})$ dans les vidéos de référence au point physique considéré. Cette approche peut être justifiée par deux arguments. Premièrement, le fait d'exploiter la moyenne des observations pour un point physique donné est une technique efficace pour atténuer la dépendance aux effets non stationnaires de l'illumination [78]. Deuxièmement, le fait de toujours se ramener à l'observation moyenne sur les vidéos de référence peut être vu comme une manière de comparer les données de référence et les données de test dans des conditions d'illumination normalisées. En pratique, l'expression analytique de la représentation normalisée $(R_{norm},G_{norm},B_{norm})$ d'une observation (R,G,B) dépend de la technique de chromaticité utilisée et sera donc détaillée dans les sections suivantes.

La seconde ligne de la figure 3.4 montre la représentation normalisée associée à une petite zone de l'image 1 de la figure 3.3, pour chacune des techniques étudiées. Il est clair que cette représentation normalisée permet d'augmenter le pouvoir discriminant par rapport à la représentation brute, ce qui permet de réduire le nombre de faux négatifs, sans pour autant réintroduire de variabilité due à l'illumination. En revanche, la normalisation des conditions d'illumination a également pour conséquence d'augmenter le nombre de faux positifs. Plus précisément, les trois mesures de chromaticité deviennent instables pour les observations faiblement lumineuses, puisqu'elles utilisent des rapports entre les couleurs considérées, qui impliquent alors une division par un terme qui tend vers zéro. Ce problème apparaît plus particulièrement avec les coordonnées chromatiques classiques, comme illustré à la figure 3.4 (flèches noires), et se manifeste par des zones de couleur très saturée. Ainsi, de manière générale, la représentation normalisée d'une observation faiblement lumineuse (e.g. issue d'une zone ombragée) peut être instable et donc source de faux positifs.

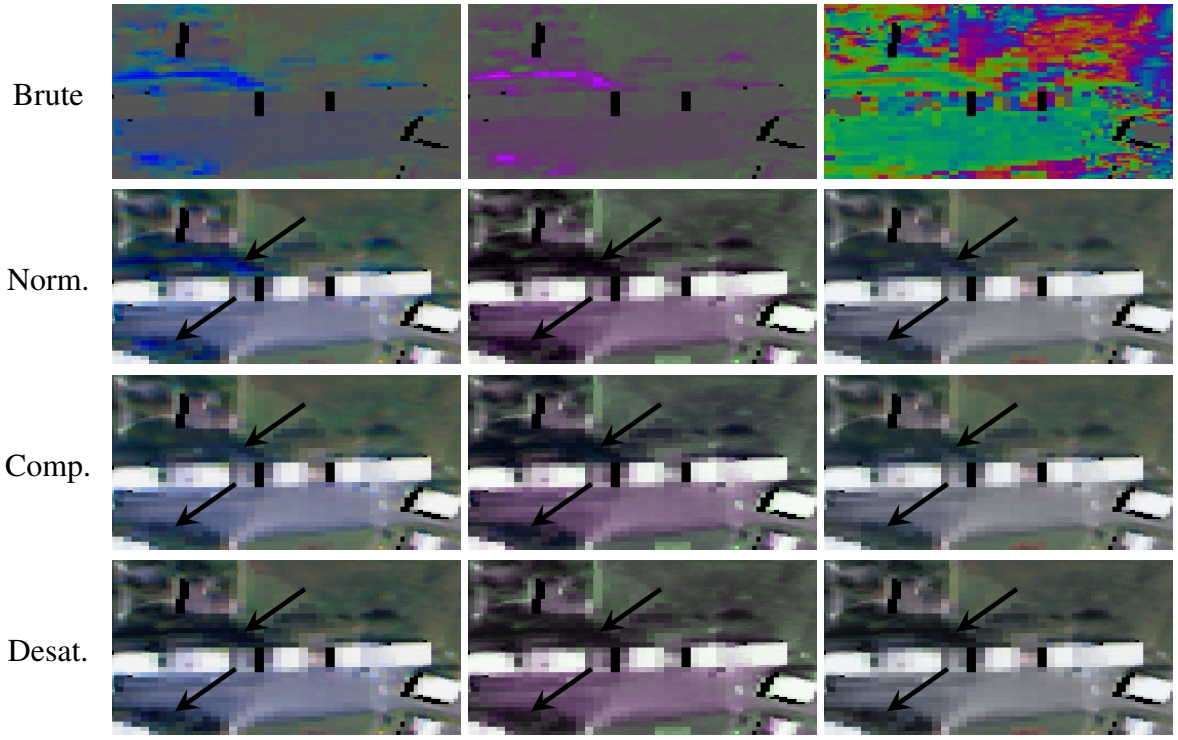


FIGURE 3.4 – *Cette figure illustre les résultats de la normalisation des conditions d'illumination, sur une petite zone de l'image 1 présentée à la figure 3.3, et montre que les observations résultantes sont plus facilement exploitables. Les première, seconde, troisième et quatrième lignes montrent respectivement les représentations brutes, normalisées, compensées et désaturées de l'image en fonction des mesures de chromaticité utilisées. Les première, seconde et troisième colonnes montrent respectivement les résultats obtenus avec les coordonnées chromatiques classiques, les coordonnées chromatiques logarithmiques et les coordonnées chromatiques L1L2L3. Les différences entre les diverses représentations sont mises en évidences par des flèches noires.*

Représentation compensée Pour contourner ce problème, nous proposons de compenser de manière progressive les observations faiblement lumineuses à l'aide de l'observation de référence moyenne. Plus précisément, en gardant les notations introduites plus haut, la représentation compensée est exprimée comme suit :

$$\begin{aligned} R_{comp} &= (1-\beta)\cdot R_{norm} + \beta \cdot \overline{R}_{ref} \\ G_{comp} &= (1-\beta)\cdot G_{norm} + \beta \cdot \overline{G}_{ref} \\ B_{comp} &= (1-\beta)\cdot B_{norm} + \beta \cdot \overline{B}_{ref} \\ \text{avec} \qquad \beta &= \left(1 - \frac{R+G+B}{3}\right)^5 \end{aligned} \tag{3.6}$$

Il est à noter que cette formulation présente un inconvénient, car elle remplace l'information présente dans l'image de test considérée par l'information issue des vidéos de référence. Ceci peut donc générer des erreurs de détection, par exemple dans le cas où l'image de test contiendrait un changement sombre, tel qu'un bâtiment noir. En effet, la zone correspondante serait alors remplacée par l'observation moyenne dans les vidéos de référence, qui par définition ne contiennent pas le changement. Par conséquent, le bâtiment disparaîtrait de l'observation de test et ne pourrait donc plus être détecté, augmentant ainsi le nombre de faux négatifs.

La troisième ligne de la figure 3.4 illustre les résultats obtenus avec la représentation compensée et permet de voir les différences avec la représentation normalisée. Dans le cas des coor-

données chromatiques classiques, le résultat montre que les zones de couleur saturées, sources de faux positifs, sont correctement éliminées.

Représentation désaturée Une alternative à la représentation compensée, permettant également de contourner le problème des faux positifs générés par la représentation normalisée, consiste à compenser de manière progressive les observations faiblement lumineuses à l'aide de la luminosité de l'observation de test. Plus précisément, en gardant les notations introduites plus haut, la représentation désaturée est exprimée comme suit :

$$\begin{aligned} R_{desat} &= (1-\beta)\cdot R_{norm} + \beta \cdot \frac{R+G+B}{3} \\ G_{desat} &= (1-\beta)\cdot G_{norm} + \beta \cdot \frac{R+G+B}{3} \\ B_{desat} &= (1-\beta)\cdot B_{norm} + \beta \cdot \frac{R+G+B}{3} \\ \text{avec} \qquad \beta &= \left(1 - \frac{R+G+B}{3}\right)^5 \end{aligned} \tag{3.7}$$

Cette représentation permet bien de conserver l'information présente dans l'image de test considérée. En revanche, pour des raisons différentes de la représentation compensée, cette formulation est également problématique. En effet, elle réintroduit la variabilité due à l'illumination dans le résultat final. Prenons l'exemple d'un bâtiment dont l'ombre portée aurait changé de direction. Le fait d'utiliser la luminosité $\frac{R+G+B}{3}$ dans la formulation de la représentation désaturée a pour conséquence que l'ombre sera toujours présente dans le résultat final. Cette ombre sera donc détectée comme changement, augmentant ainsi le nombre de faux positifs. Ceci montre ainsi que le choix entre la représentation compensée et la représentation désaturée dépendra là encore du compromis souhaité entre faux positifs et faux négatifs.

La quatrième ligne de la figure 3.4 illustre les résultats obtenus avec la représentation désaturée et permet de voir les différences avec la représentation normalisée. Ces résultats montrent que, comme dans le cas de la représentation compensée, les zones de couleur saturées sont correctement éliminées sans influer sur les autres parties de l'image. Cependant, les ombres sont plus sombres et plus marquées que dans le cas de la représentation compensée.

Pour une image donnée, chacune des méthodes présentées ci-dessus traite les pixels de manière indépendante les uns des autres. Par conséquent, la complexité algorithmique de ces méthodes est en $O(N_{\text{images}} N_{\text{pixels}})$, où N_{images} et le nombre d'images à traiter et N_{pixels} le nombre de pixels par image. Les performances en détection de changements de ces différentes représentations, en fonction de la technique de chromaticité utilisée, seront analysées au chapitre 6.

3.2.2 Invariance via les coordonnées chromatiques classiques

Dans le cas où les observations considérées sont représentées dans l'espace *RGB*, la chromaticité d'une observation peut être obtenue très simplement grâce aux coordonnées chromatiques classiques. Ces coordonnées chromatiques classiques, utilisées notamment par Elgammal et al. [38], possèdent des propriétés d'invariance intéressantes vis-à-vis de l'illumination. En effet, Gevers et Smeulders [47] montrent que, sous l'hypothèse que la scène observée est éclairée par une source lumineuse blanche et que les objets suivent une loi de diffusion Lambertienne, ces coordonnées chromatiques sont invariantes par changement de point de vue, par changement de la direction de l'illumination et par changement de l'intensité de l'illumination. Ces grandeurs restent intéressantes même lorsque les hypothèses mentionnées ne sont pas rigoureusement vérifiées, mais cela peut alors générer des faux positifs dans certains cas. En particulier, lorsque la source lumineuse n'est pas blanche, des variations de couleur peuvent apparaître localement, par exemple dans les zones sur-éclairés ou ombragées, générant ainsi des faux positifs.

Pour une observation (R,G,B) donnée, les coordonnées chromatiques classiques sont obtenues de la façon suivante :

$$(r,g,b) = \left(\frac{R}{R+G+B}, \frac{G}{R+G+B}, \frac{B}{R+G+B}\right) \tag{3.8}$$

Notons que la troisième coordonnée est redondante par rapport aux deux autres, puisque $b = 1-r-g$, et est donc fréquemment ignorée. La seconde colonne de la figure 3.3 montre le résultat de la représentation brute de deux images à l'aide des coordonnées chromatiques classiques.

Afin de minimiser la perte d'information liée au fait que les coordonnées chromatiques (r,g,b) ignorent la luminosité de la scène, nous utilisons l'observation moyenne sur les vidéos de référence $(\overline{R}_{ref}, \overline{G}_{ref}, \overline{B}_{ref})$ pour reconvertir les coordonnées chromatiques vers l'espace de couleur *RGB* classique. La représentation normalisée est exprimée de la manière suivante :

$$\begin{aligned}
R_{norm} &= r \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) = \frac{R}{R+G+B} \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) \\
G_{norm} &= g \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) = \frac{G}{R+G+B} \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) \\
B_{norm} &= b \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) = \frac{B}{R+G+B} \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref})
\end{aligned} \tag{3.9}$$

3.2.3 Invariance via les coordonnées chromatiques logarithmiques

Finlayson et al. [44] proposent une méthode intéressante dans le cas où l'objectif consiste à filtrer la quasi-totalité de l'information liée à l'illumination. Pour cela, ils analysent en détails le principe d'acquisition d'une image par une caméra, et proposent un modèle valable sous certaines hypothèses concernant la caméra et la scène observée. D'abord, il est nécessaire que les sensibilités spectrales des capteurs (R,G,B) de la caméra soient assimilables à des distributions de Dirac, c'est-à-dire que les grandeurs R, G et B représentent l'intensité lumineuse à trois longueurs d'onde précises (par opposition à trois gammes de longueurs d'onde). D'autre part, ce modèle est valable pour les objets suivant une loi de diffusion Lambertienne et pour les sources lumineuses suivant la loi de radiation des corps noirs. Ce modèle est donc valable dans le cas de sources lumineuses colorées.

Sous ces conditions, il est démontré que les coordonnées chromatiques logarithmiques, associées à une observation (R,G,B) donnée, peuvent s'exprimer de la façon suivante :

$$(\chi_R, \chi_G, \chi_B) = \left(log(\frac{R}{\sqrt[3]{R \cdot G \cdot B}}), log(\frac{G}{\sqrt[3]{R \cdot G \cdot B}}), log(\frac{B}{\sqrt[3]{R \cdot G \cdot B}})\right) = \boldsymbol{\rho} + \frac{s}{T} \cdot \mathbf{V}_{illum}(\xi) \tag{3.10}$$

où T est la température de corps noir associée à la source lumineuse, $\boldsymbol{\rho}$ et s sont des constantes par rapport à T et $\mathbf{V}_{illum}(\xi)$ est le vecteur, caractérisé par le paramètre angulaire ξ, représentant la direction de variation des coordonnées chromatiques logarithmiques en fonction de l'illumination. L'équation précédente montre que les coordonnées chromatiques logarithmiques s'expriment linéairement par rapport à la grandeur $\frac{1}{T}$. Par conséquent, en disposant d'une valeur estimée de la direction de variation $\mathbf{V}_{illum}(\xi)$, il est possible de s'affranchir des conditions d'illumination en projetant les coordonnées chromatiques logarithmiques de manière orthogonale à $\mathbf{V}_{illum}(\xi)$, procédé qui débouche sur de nouvelles coordonnées chromatiques logarithmiques désignées par $(\chi_R^\perp, \chi_G^\perp, \chi_B^\perp)$. La section A.2 en annexe détaille l'établissement de la matrice de projection, fonction du paramètre angulaire ξ, permettant d'éliminer la composante d'illumination dans une observation. Finalement, il est possible d'exprimer une chromaticité linéaire, et non plus logarithmique, comme suit :

$$(r,g,b) = \left(exp(\chi_R^\perp), exp(\chi_G^\perp), exp(\chi_B^\perp)\right) \tag{3.11}$$

La troisième colonne de la figure 3.3 montre le résultat de la représentation brute de deux images à l'aide des coordonnées chromatiques logarithmiques. Afin de minimiser la perte d'information liée au fait que les coordonnées chromatiques (r,g,b) ignorent la luminosité des objets la scène, nous utilisons l'observation moyenne sur les vidéos de référence $(\overline{R}_{ref},\overline{G}_{ref},\overline{B}_{ref})$ pour reconvertir les coordonnées chromatiques vers l'espace de couleur *RGB* classique. La représentation normalisée est exprimée de la manière suivante :

$$\begin{aligned} R_{norm} &= r\cdot\sqrt[3]{\overline{R}_{ref}\cdot\overline{G}_{ref}\cdot\overline{B}_{ref}} = exp(\chi_R^{\perp})\cdot\sqrt[3]{\overline{R}_{ref}\cdot\overline{G}_{ref}\cdot\overline{B}_{ref}} \\ G_{norm} &= g\cdot\sqrt[3]{\overline{R}_{ref}\cdot\overline{G}_{ref}\cdot\overline{B}_{ref}} = exp(\chi_G^{\perp})\cdot\sqrt[3]{\overline{R}_{ref}\cdot\overline{G}_{ref}\cdot\overline{B}_{ref}} \\ B_{norm} &= b\cdot\sqrt[3]{\overline{R}_{ref}\cdot\overline{G}_{ref}\cdot\overline{B}_{ref}} = exp(\chi_B^{\perp})\cdot\sqrt[3]{\overline{R}_{ref}\cdot\overline{G}_{ref}\cdot\overline{B}_{ref}} \end{aligned} \tag{3.12}$$

3.2.4 Invariance via les coordonnées chromatiques L1L2L3

La définition de l'espace de couleurs *L1L2L3*, introduit par Gevers et Smeulders [47], est inspirée de la notion de teinte et permet de représenter les observations de manière invariante par rapport à l'illumination. D'après les auteurs, sous l'hypothèse que la scène observée est éclairée par une source lumineuse blanche, cette représentation est invariante par changement de point de vue, par changement de la direction de l'illumination, par changement de l'intensité de l'illumination et en présence ou non de zones sur-éclairées. En revanche, elles sont sensibles aux éventuels changements de couleur de l'illumination. Le point intéressant de cet espace de couleur est qu'il ne suppose plus que les objets de la scène suivent une loi de diffusion Lambertienne.

Pour une observation (R,G,B) donnée, les coordonnées chromatiques *L1L2L3* s'expriment comme suit :

$$\begin{aligned} r &= \frac{(R-G)^2}{(R-G)^2+(R-B)^2+(G-B)^2} \\ g &= \frac{(R-B)^2}{(R-G)^2+(R-B)^2+(G-B)^2} \\ b &= \frac{(G-B)^2}{(R-G)^2+(R-B)^2+(G-B)^2} \end{aligned} \tag{3.13}$$

La quatrième colonne de la figure 3.3 montre le résultat de la représentation brute de deux images à l'aide des coordonnées chromatiques *L1L2L3*.

Afin de minimiser la perte d'information liée au fait que les coordonnées chromatiques (r,g,b) ignorent la luminosité de la scène, nous utilisons l'observation moyenne sur les vidéos de référence $(\overline{R}_{ref},\overline{G}_{ref},\overline{B}_{ref})$ pour reconvertir les coordonnées chromatiques vers l'espace de couleur *RGB* classique. Nous proposons d'utiliser les expressions suivantes pour la représentation normalisée :

$$\begin{aligned} R_{norm} &= \frac{(\overline{R}_{ref}+\overline{G}_{ref}+\overline{B}_{ref})+\Delta_{R-G}+\Delta_{R-B}}{3} \\ G_{norm} &= \frac{(\overline{R}_{ref}+\overline{G}_{ref}+\overline{B}_{ref})+\Delta_{G-B}-\Delta_{R-G}}{3} \\ B_{norm} &= \frac{(\overline{R}_{ref}+\overline{G}_{ref}+\overline{B}_{ref})-\Delta_{G-B}-\Delta_{R-B}}{3} \end{aligned} \tag{3.14}$$

avec

$$\begin{aligned} \Delta_{R-G} &= \mathrm{sgn}(R-G)\cdot\sqrt{r\cdot\left[(\overline{R}_{ref}-\overline{G}_{ref})^2+(\overline{R}_{ref}-\overline{B}_{ref})^2+(\overline{G}_{ref}-\overline{B}_{ref})^2\right]} \\ \Delta_{R-B} &= \mathrm{sgn}(R-B)\cdot\sqrt{g\cdot\left[(\overline{R}_{ref}-\overline{G}_{ref})^2+(\overline{R}_{ref}-\overline{B}_{ref})^2+(\overline{G}_{ref}-\overline{B}_{ref})^2\right]} \\ \Delta_{G-B} &= \mathrm{sgn}(G-B)\cdot\sqrt{b\cdot\left[(\overline{R}_{ref}-\overline{G}_{ref})^2+(\overline{R}_{ref}-\overline{B}_{ref})^2+(\overline{G}_{ref}-\overline{B}_{ref})^2\right]} \end{aligned}$$

Dans ces équations, la fonction $\mathrm{sgn}(z)$ retourne le signe de z, et (R,G,B) est l'observation correspondant aux coordonnées chromatiques (r,g,b).

Chapitre 4

Détection de changements

La problématique centrale de cette thèse, présentée en détail dans le chapitre 1, consiste à proposer une méthode permettant de détecter les changements entre une vidéo de test et un ensemble de vidéos de référence, vidéos acquises par des plates-formes aux trajectoires arbitraires. Nous avons évoqué le fait que cette problématique soulève de nombreuses questions. Par exemple, il faut aborder le problème des effets géométriques (e.g. parallaxe et occultations) et des faux positifs qu'ils peuvent générer. Il faut aussi aborder le problème des changements, dus à une apparence variable (e.g. feuilles d'arbres, eau, etc) ou à des objets mobiles, qui peuvent survenir entre les différentes vidéos de référence utilisées. Nous avons vu dans le chapitre 2 qu'une modélisation tri-dimensionnelle des apparences de la scène permet d'aborder de manière efficace une bonne partie de ces problèmes. Cette approche très générale laisse cependant la place à de nombreux choix de conception, en particulier vis-à-vis de l'organisation des données dans la base de référence.

Le présent chapitre présente donc l'approche adoptée dans le cadre de cette thèse, qui consiste à organiser les observations de référence dans une base de données tri-dimensionnelle, créée hors-ligne grâce aux vidéos de référence, et exploitable en ligne pour traiter la vidéo de test. Notre approche est notamment intéressante parce qu'elle permet de séparer le problème géométrique du problème de modélisation d'apparence. Par conséquent, notre approche est capable d'utiliser et de comparer différentes techniques de modélisation d'apparence, si elle dispose des implémentations correspondantes.

La suite de ce chapitre est organisé de la manière suivante. Pour commencer, la section 4.1 détaille les travaux réalisés autour d'une technique bi-dimensionnelle de détection de changements, exploitant des paires d'images, ainsi que les difficultés rencontrées dans le cas de vidéos aériennes qui justifient l'utilisation d'une technique tri-dimensionnelle. La section 4.2 présente l'organisation des observations de référence dans notre base de données tri-dimensionnelle et décrit les mécanismes liés à l'indexation et aux requêtes. Enfin, la section 4.3 décrit les techniques de modélisation des apparences implémentées.

Sommaire

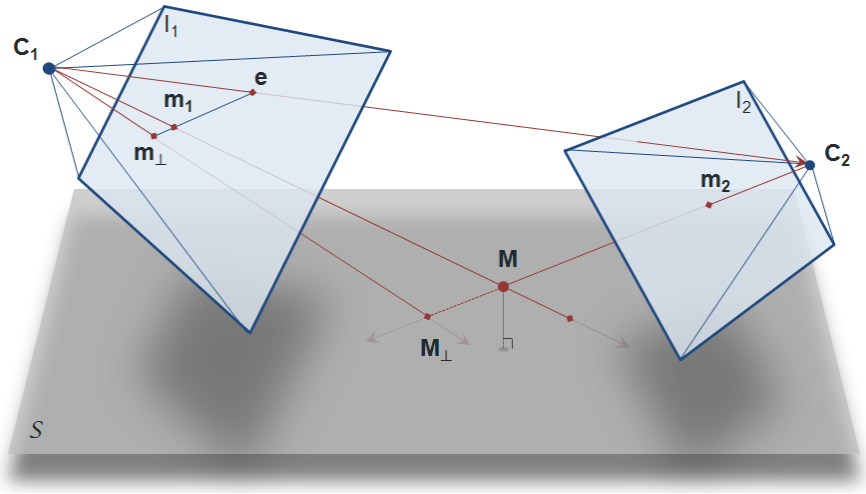


FIGURE 4.1 – *Ce schéma étaie le raisonnement montrant que le champ de vecteurs de parallaxe résiduel, après recalage de deux images selon une surface paramétrique, est un champ épipolaire. Dans cet exemple, pour recaler correctement I_1 et I_2, il faut aligner les points 2D $\mathbf{m}_1$ et $\mathbf{m}_2$ correspondant au point physique $\mathbf{M}$. Cependant, le recalage par rapport au plan $\mathcal{S}$ aligne $\mathbf{m}_2$ avec $\mathbf{m}_\perp$, donc le vecteur de recalage résiduel est le vecteur $\mathbf{m}_1 - \mathbf{m}_\perp$.*

4.1 Approche bi-dimensionnelle

Cette section décrit la technique bi-dimensionnelle développée dans le cadre de cette thèse [14] pour la détection de changements entre paires d'images correspondantes. Cette méthode explore l'idée de combiner un algorithme de flot optique avec la contrainte issue de la géométrie épipolaire, afin de détecter les changements entre deux images aériennes de manière rapide et robuste aux effets de parallaxe.

Cette idée se base sur un constat théorique, démontré par exemple par Kumar et al. [62], valable pour une scène ne contenant pas de mouvements. Ce constat concerne le champ de vecteurs permettant de corriger un recalage approximatif, que nous désignerons dans la suite par champ de vecteurs résiduels de recalage. Il affirme que, après recalage de deux vues d'une même scène selon une surface paramétrique, le champ de vecteurs résiduels de recalage, qui est dû aux effets de parallaxe en supposant qu'aucun mouvement n'est présent dans la scène, est un champ épipolaire. En d'autres termes, après un recalage approximatif de deux images observant la même scène, la correction à appliquer à un point donné, mal recalé du fait des effets de parallaxe, est une translation dans la direction de la droite épipolaire en ce point. Ce constat peut être démontré par un raisonnement très simple, étayé par le schéma de la figure 4.1. Soient deux images I_1 et I_2 observant la même scène et respectivement acquises depuis les positions $\mathbf{C}_1$ et $\mathbf{C}_2$. Supposons que l'image I_2 a été recalée par rapport à I_1 selon une surface paramétrique $\mathcal{S}$ donnée, par exemple le plan dominant du sol, et considérons un point $\mathbf{M}$ n'appartenant pas à cette surface. Ce point $\mathbf{M}$ est projeté dans l'image I_1 à la position $\mathbf{m}_1$ et dans l'image I_2 à la position $\mathbf{m}_2$. Comme $\mathbf{M}$ n'appartient pas à la surface $\mathcal{S}$, le recalage de I_2 par rapport à I_1 selon la surface $\mathcal{S}$ ne va pas correctement aligner $\mathbf{m}_2$ avec $\mathbf{m}_1$ mais va aligner $\mathbf{m}_2$ avec $\mathbf{m}_\perp$, la projection dans I_1 du point d'intersection $\mathbf{M}_\perp$ entre le rayon de direction $\mathbf{M} - \mathbf{C}_2$ et la surface $\mathcal{S}$. Par ailleurs, comme les points $\mathbf{C}_2$, $\mathbf{M}$ et $\mathbf{M}_\perp$ sont alignés dans l'espace, leurs projections $\mathbf{e}$, $\mathbf{m}_1$ et $\mathbf{m}_\perp$ sont également alignées dans l'image I_1. Par conséquent, le vecteur $\mathbf{m}_1 - \mathbf{m}_\perp$ est colinéaire au vecteur $\mathbf{e} - \mathbf{m}_\perp$ et, puisque $\mathbf{e}$ est également l'épipole de I_1 par rapport à I_2, le vecteur $\mathbf{m}_1 - \mathbf{m}_\perp$ (ou vecteur résiduel de parallaxe pour le point $\mathbf{m}_1$) est dans la direction de la droite épipolaire au point $\mathbf{m}_\perp$.

FIGURE 4.2 – *Cette figure présente les deux images utilisées pour illustrer le fonctionnement de notre technique bi-dimensionnelle de détection de changement entre paires d'images. Le changement présent dans l'image de test, à droite, est mis en évidence par un cercle.*

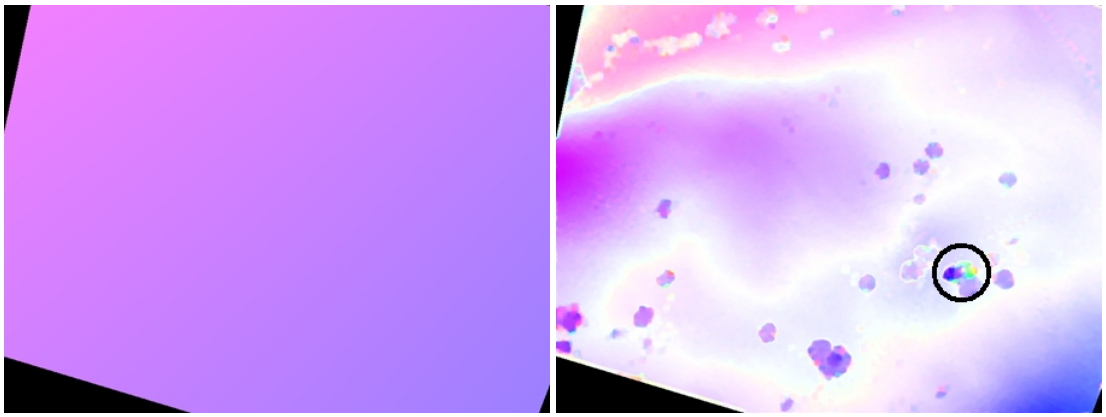


FIGURE 4.3 – *Cette figure compare le champ de vecteurs épipolaires avec le champ vectoriel estimé par un algorithme de flot optique, pour les images de la figure 4.2. La direction et la norme du vecteur en un point donné sont codées dans l'espace de couleur TSV respectivement grâce à la teinte et à la saturation du point, la valeur étant constante égale à* 255.

Ce constat est intéressant, car il signifie que la mise en correspondance de deux images, observant la même scène depuis deux points de vue différents, peut se faire en effectuant en premier lieu un recalage simple de ces images, puis en cherchant les points correspondants de manière colinéaire aux droites épipolaires, qui peuvent être déterminées grâce aux images. D'autre part, dans le contexte de la détection de changements, le fait qu'un changement soit survenu dans l'une des deux images signifie que la zone correspondante n'apparaît que dans une seule des deux images, par conséquent la mise en correspondance de cette zone est impossible. Cela fournit donc un critère permettant de distinguer les changements des effets de parallaxe.

La technique bi-dimensionnelle de détection de changements que nous avons développée pour les paires d'images exploite donc ce principe et utilise un algorithme de flot optique pour estimer le champ vectoriel résiduel de recalage. La figure 4.3 compare les directions de ce champ vectoriel résiduel de recalage, estimé ici par l'algorithme de flot optique proposé par Farneback [39], avec celles du champ épipolaire, pour les images présentées à la figure 4.2. Pour cela, la direction d'un vecteur en un point donné est codée grâce à la teinte (i.e. la chromaticité de la couleur) de ce point, la saturation (i.e. le caractère plus ou moins pâle de la couleur) permettant ici de coder la norme du vecteur. Cette figure montre ainsi que les deux images sont relativement semblables, les différences étant principalement dues à la saturation des couleurs,

c'est-à-dire à la norme des vecteurs [1]. À l'exception de quelques erreurs au niveau des bordures des arbres, cela signifie que l'algorithme de flot optique permet presque directement d'estimer le champ de vecteurs résiduels de parallaxe. Pour éliminer les dernières fausses alarmes, nous avons proposé [14] de contraindre l'algorithme de flot optique à effectuer la recherche dans la direction de la droite épipolaire. La suite de cette section fournit les détails techniques relatifs à la mise en œuvre de notre approche bi-dimensionnelle.

Soit I_1 et I_2 les deux images considérées après recalage par rapport à une surface paramétrique donnée. L'algorithme de flot optique proposé par Farneback [39] calcule le vecteur résiduel de recalage, en un point $\mathbf{m}_0$ donné dans les images, grâce à une expansion polynomiale du signal image effectuée sur un voisinage de $\mathbf{m}_0$. Soit $\mathrm{Poly}_{1,\mathbf{m}_0}(\mathbf{m})$ et $\mathrm{Poly}_{2,\mathbf{m}_0}(\mathbf{m})$ les expansions polynomiales bi-dimensionnelles de I_1 et I_2 sur un voisinage de $\mathbf{m}_0$. Ces polynômes s'expriment de la façon suivante :

$$\begin{aligned}\mathrm{Poly}_{1,\mathbf{m}_0}(\mathbf{m}) &= \mathbf{m}^T \cdot \mathrm{A}_{1,\mathbf{m}_0} \cdot \mathbf{m} + \mathbf{b}_{1,\mathbf{m}_0}^T \cdot \mathbf{m} + c_{1,\mathbf{m}_0} \\ \mathrm{Poly}_{2,\mathbf{m}_0}(\mathbf{m}) &= \mathbf{m}^T \cdot \mathrm{A}_{2,\mathbf{m}_0} \cdot \mathbf{m} + \mathbf{b}_{2,\mathbf{m}_0}^T \cdot \mathbf{m} + c_{2,\mathbf{m}_0}\end{aligned}$$

où les coefficients $\mathrm{A}_{i,\mathbf{m}_0}$, $\mathbf{b}_{i,\mathbf{m}_0}$ et $c_{i,\mathbf{m}_0}$, pour $i \in \{1,2\}$, sont calculés par interpolation des images sur un voisinage de $\mathbf{m}_0$ de taille paramétrable (voir [39]).

Dans le cas idéal, le vecteur résiduel de recalage en $\mathbf{m}_0$ affecte la totalité de son voisinage de manière homogène. Les deux expansions polynomiales peuvent par conséquent être identifiées à une translation près : $\mathrm{Poly}_{2,\mathbf{m}_0}(\mathbf{m}) = \mathrm{Poly}_{1,\mathbf{m}_0}(\mathbf{m} - \mathbf{d}_{\mathbf{m}_0})$. Cette relation permet alors d'en déduire l'expression du vecteur de recalage résiduel $\mathbf{d}_{\mathbf{m}_0}$:

$$\begin{aligned}\mathrm{Poly}_{2,\mathbf{m}_0}(\mathbf{m}) = \mathrm{Poly}_{1,\mathbf{m}_0}(\mathbf{m} - \mathbf{d}_{\mathbf{m}_0}) \quad &\Rightarrow \quad \begin{cases} \mathrm{A}_{1,\mathbf{m}_0} = \mathrm{A}_{2,\mathbf{m}_0} \\ \mathbf{b}_{2,\mathbf{m}_0} = \mathbf{b}_{1,\mathbf{m}_0} - 2 \cdot \mathrm{A}_{1,\mathbf{m}_0} \cdot \mathbf{d}_{\mathbf{m}_0} \end{cases} \\ &\Rightarrow \quad \mathrm{A}_{1,\mathbf{m}_0} \cdot \mathbf{d}_{\mathbf{m}_0} = -\frac{1}{2}\left(\mathbf{b}_{2,\mathbf{m}_0} - \mathbf{b}_{1,\mathbf{m}_0}\right) \qquad (4.1)\end{aligned}$$

Cette expression permet donc de calculer $\mathbf{d}_{\mathbf{m}_0}$ directement en fonction de $\mathbf{b}_{1,\mathbf{m}_0}$, $\mathbf{b}_{2,\mathbf{m}_0}$ et $\mathrm{A}_{1,\mathbf{m}_0}$. Cependant, pour rendre l'algorithme robuste au bruit présent dans les images, il est préférable d'utiliser la moyenne entre $\mathrm{A}_{1,\mathbf{m}_0}$ et $\mathrm{A}_{2,\mathbf{m}_0}$, menant à l'expression suivante :

$$\begin{aligned}\mathrm{A}_{\mathbf{m}_0} \cdot \mathbf{d}_{\mathbf{m}_0} &= \Delta\mathbf{b}_{\mathbf{m}_0} \qquad (4.2) \\ \text{avec} \qquad \mathrm{A}_{\mathbf{m}_0} &= \frac{1}{2} \cdot \left(\mathrm{A}_{1,\mathbf{m}_0} + \mathrm{A}_{2,\mathbf{m}_0}\right) \\ \Delta\mathbf{b}_{\mathbf{m}_0} &= -\frac{1}{2} \cdot \left(\mathbf{b}_{2,\mathbf{m}_0} - \mathbf{b}_{1,\mathbf{m}_0}\right)\end{aligned}$$

Par ailleurs, cette expression correspond au cas idéal où le vecteur de recalage résiduel affecte la totalité de son voisinage de manière homogène, ce qui n'est pas toujours le cas en pratique. Par conséquent, pour augmenter la robustesse par rapport aux déviations à ce cas idéal, $\mathbf{d}_{\mathbf{m}_0}$ est estimé à l'aide d'une minimisation aux moindres carrés, en considérant l'ensemble des expansions polynomiales sur un voisinage $\mathcal{V}(\mathbf{m}_0)$. On obtient alors les expressions suivantes pour $\mathbf{d}_{\mathbf{m}_0}$ et $\varepsilon_{\mathbf{m}_0}$, qui désigne l'erreur d'estimation :

$$\begin{aligned}\widehat{\mathbf{d}}_{\mathbf{m}_0} &= \arg\min_{\mathbf{d}} \sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \left\| \mathrm{A}_{\mathbf{m}_v} \cdot \mathbf{d} - \Delta\mathbf{b}_{\mathbf{m}_v} \right\|^2 = \left(\sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \left(\mathrm{A}_{\mathbf{m}_v}^T \cdot \mathrm{A}_{\mathbf{m}_v}\right) \right)^{-1} \cdot \sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \left(\mathrm{A}_{\mathbf{m}_v}^T \cdot \Delta\mathbf{b}_{\mathbf{m}_v}\right) \qquad (4.3) \\ \widehat{\varepsilon}_{\mathbf{m}_0} &= \sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \left(\Delta\mathbf{b}_{\mathbf{m}_v}^T \cdot \Delta\mathbf{b}_{\mathbf{m}_v}\right) - \mathbf{d}_{\mathbf{m}_0}^T \cdot \sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \left(\mathrm{A}_{\mathbf{m}_v}^T \cdot \Delta\mathbf{b}_{\mathbf{m}_v}\right)\end{aligned}$$

1. La norme des vecteurs résiduels de parallaxe est liée à la distance du point par rapport à la surface paramétrique utilisée. Dans notre cas, nous utilisons le plan du sol, et la norme du vecteur décrit donc l'altitude du point au dessus du sol. Dans l'image de droite de la figure 4.3, le relief de la scène observée peut ainsi être discerné, à l'aide de la saturation des couleurs.

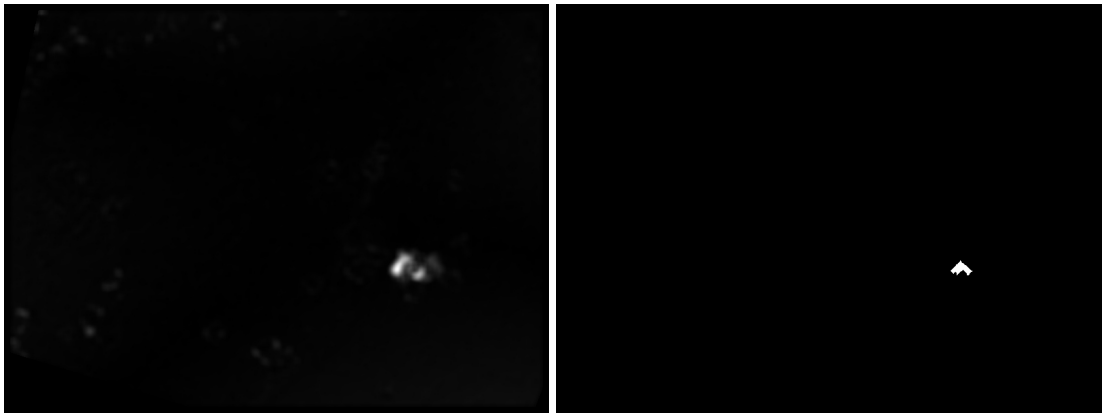


FIGURE 4.4 – *Cette figure présente à gauche, la carte des erreurs d'estimations définies à l'équation 4.4 et estimées grâce à la technique exploitant le flot optique contraint par la géométrie épipolaire, et à droite, le masque de changements issu de la vérité-terrain.*

L'équation 4.3, qui en pratique est employée de manière itérative selon une approche à résolution progressive (*coarse-to-fine* dans la littérature), permet d'estimer le vecteur résiduel de recalage $\widehat{\mathbf{d}}_{\mathbf{m}_0}$ sans contrainte de direction. Dans le cadre de la détection de changements, le fait de contraindre la direction du vecteur résiduel de recalage permet de réduire le nombre de fausses alarmes [14]. Ceci peut être fait en introduisant dans les équations précédentes la direction $\mathbf{V}_e(\mathbf{m}_0)$ de la droite épipolaire en $\mathbf{m}_0$, et en estimant la norme d du vecteur par moindres carrés :

$$\begin{aligned}
\widehat{d} &= \arg\min_d \sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \| d \cdot \mathrm{A}_{\mathbf{m}_v} \cdot \mathbf{V}_e(\mathbf{m}_0) - \Delta\mathbf{b}_{\mathbf{m}_v} \|^2 \\
&= \left(\sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \left[\mathbf{V}_e(\mathbf{m}_0)^T \cdot \mathrm{A}_{\mathbf{m}_v}^T \cdot \mathrm{A}_{\mathbf{m}_v} \cdot \mathbf{V}_e(\mathbf{m}_0) \right] \right)^{-1} \cdot \sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \left[\mathbf{V}_e(\mathbf{m}_0)^T \cdot \mathrm{A}_{\mathbf{m}_v}^T \cdot \Delta\mathbf{b}_{\mathbf{m}_v} \right] \qquad (4.4) \\
\widehat{\varepsilon}_{\mathbf{m}_0} &= \sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \left[\Delta\mathbf{b}_{\mathbf{m}_v}^T \cdot \Delta\mathbf{b}_{\mathbf{m}_v} \right] - \widehat{d} \cdot \mathbf{V}_e(\mathbf{m}_0)^T \cdot \sum_{\mathbf{m}_v \in \mathcal{V}(\mathbf{m}_0)} \left[\mathrm{A}_{\mathbf{m}_v}^T \cdot \Delta\mathbf{b}_{\mathbf{m}_v} \right]
\end{aligned}$$

La figure 4.4 présente le masque de changements estimé en appliquant cette technique sur les images de la figure 4.2.

Pour définir une valeur adéquate pour le seuillage des erreurs d'estimation et leur conversion en un masque de changements binaire, nous avons formulé un test d'hypothèse statistique basé sur le rapport des vraisemblances. Soit m une variable aléatoire représentant un point dans l'image de test considérée et soit Y la variable aléatoire binaire représentant le fait que m corresponde ($Y = 1$, hypothèse $\mathcal{H}_1$) ou non ($Y = 0$, hypothèse $\mathcal{H}_0$) à un changement. Désignons alors les distributions de probabilité de l'erreur $\widehat{\varepsilon}_m$, conditionnellement à $\mathcal{H}_1$ et $\mathcal{H}_0$, respectivement par $p_{\mathcal{H}_1}(m) = p(\widehat{\varepsilon}_m | Y = 1)$ et $p_{\mathcal{H}_0}(m) = p(\widehat{\varepsilon}_m | Y = 0)$. Le test du rapport des vraisemblances permet de prendre une décision en faveur de l'hypothèse $\mathcal{H}_1$ ou $\mathcal{H}_0$ selon que le signe de $\text{vraisemblance}(m) - \tau$ est respectivement positif ou négatif, avec :

$$\text{vraisemblance}(m) = \frac{p_{\mathcal{H}_1}(m)}{p_{\mathcal{H}_0}(m)} \quad \text{et} \quad \tau = \frac{p(Y=0)}{p(Y=1)} \qquad (4.5)$$

où les distributions de probabilité de l'erreur, conditionnellement aux hypothèses $\mathcal{H}_0$ et $\mathcal{H}_1$, ont été modélisées respectivement par une distribution exponentielle et une distribution de Rayleigh.

L'évaluation de cette technique sur une base de données de synthèse [14] montre de bonnes performances ainsi qu'une bonne robustesse à la présence de fort relief dans la scène et aux

changements de points de vue. Cependant, un certain nombre de limites importantes rendent cette approche inefficace dans les cas d'application réels.

Pour commencer, les algorithmes de flot optique ont été conçus pour estimer le recalage entre images successives d'une séquence d'images [6]. Par conséquent, ils ne sont pas adaptés pour traiter des images issues de séquences différentes et dont les conditions d'acquisition peuvent être très différentes (e.g. bruit, apparences de la scène, illumination, météo, etc), ce qui est précisément le but en détection de changements. D'autre part, dans le cadre de cette thèse nous considérons des données vidéo, qui génèrent un fort taux de recouvrement entre les images de référence. Or, cette technique ne permet de comparer que deux images entre elles, ce qui nécessite de trouver ou de construire une unique image de référence correspondant aux nombreuses images candidates pour la comparaison. Cela introduit donc dans l'approche globale une opération de sélection et/ou de fusion, qui la rend inefficace.

Comme l'a montré le chapitre 2, l'exploitation d'un modèle tri-dimensionnel et d'une modélisation des apparences permet d'apporter une solution appropriée à ces problèmes. La suite de ce chapitre présente donc les travaux réalisés pour la détection de changements à l'aide d'une base de données tri-dimensionnelle. La section 4.2 décrit l'organisation tri-dimensionnelle de cette base de données et les techniques de modélisation d'apparence sont détaillées à la section 4.3.

4.2 Base de données tri-dimensionnelle

Notre approche de détection de changements entre vidéos aériennes utilise une base de données tri-dimensionnelle pour organiser spatialement, c'est-à-dire de manière cohérente par rapport à la géométrie de la scène observée, d'une part les images de référence et d'autre part les modèles d'apparence.

Vis-à-vis de l'organisation des modèles d'apparence, certaines approches de la littérature ont choisi d'employer une modélisation volumique de la scène [32, 90]. Or, dans un contexte où la plate-forme d'acquisition reste à une distance importante de la scène observée, il semble plus approprié de modéliser la scène comme une surface plutôt que comme un volume. En effet, l'amélioration de la précision d'une modélisation volumique est souvent anecdotique, mais se paie par un coût de stockage bien plus important, qui peut devenir prohibitif pour de vastes scènes. Nous avons donc proposé [18] de représenter la géométrie de la scène par une surface avec altitude variable. Notre base de données tri-dimensionnelle exploite donc une représentation, que nous désignerons dans la suite par Quad-Tree augmenté, combinant une carte d'élévation avec une structure arborescente bi-dimensionnelle appelée Quad-Tree [43]. L'utilisation d'un Quad-Tree est une approche appropriée au compromis entre finesse de modélisation et complexité du modèle, car elle permet d'ajuster la finesse des cellules là où la résolution au sol des observations est la plus fine, tout en utilisant une grosse maille de modélisation pour les zones non observées ou correspondant à des observations peu résolues. De plus, la structure arborescente du Quad-Tree, qui peut être vue comme un système d'indexation spatiale, permet également un accès efficace aux cellules lors de requêtes spatiales, telles que celles utilisées lors d'un algorithme de lancer de rayon.

Cette structure de Quad-Tree augmenté est ainsi compatible avec la modélisation des apparences, qui, comme nous l'avons vu au chapitre 2, permet d'exploiter les données de référence au maximum de leur potentiel pour la détection de changement automatique. Cependant, l'objectif étant d'assister un analyste image dans sa tâche, il est également important de conserver et de permettre la navigation dans les images initiales, qui sont beaucoup plus informatives pour l'analyste que les modèles d'apparence. Par conséquent, notre base de données tri-dimensionnelle organise également les images de référence brutes par rapport à la géométrie

de la scène, grâce à une structure de R-Tree [50], qui permet également un accès efficace lors de requêtes spatiales.

Plus généralement, notre base de données tri-dimensionnelle est utilisable selon deux modes de fonctionnement. Le premier mode est un mode d'ingestion hors-ligne, qui permet d'ingérer des vidéos de référence pour effectuer la modélisation et la compression des observations associées. Le second mode est un mode de requête en-ligne, qui permet de comparer, à mesure que la vidéo de test est acquise, les observations de test avec les modèles d'apparence générés à partir des vidéos de référence. La section 4.2.1 détaille les traitements relatifs à l'indexation des observations de référence lors de l'ingestion d'une nouvelle vidéo. La section 4.2.2 décrit les mécanismes permettant la mise en correspondance efficace des données de test et de référence, étape cruciale à la fois pour la modélisation des observations de référence et la détection de changements dans l'image de test.

4.2.1 Indexation spatiale des données

Cette section détaille les traitements, mis en œuvre lors de l'ingestion d'une nouvelle vidéo de référence dans la base de données, relatifs d'une part à l'indexation des images de référence grâce à la construction d'un R-Tree, et d'autre part à l'indexation des observations de référence grâce à un Quad-Tree augmenté. Ces traitements supposent qu'un modèle numérique de surface (MNS), décrivant la surface du sol uniquement (on parle alors de modèle numérique de terrain, MNT) ou du sol et du sur-sol tel que la végétation ou les bâtiments (on parle alors de modèle numérique d'élévation, MNE), correspondant à la région observée est disponible [2] et que les paramètres d'acquisition de chaque image sont connus (voir la section 3.1.1 pour une méthode d'estimation).

4.2.1.1 Indexation spatiale des images de référence

L'intérêt d'indexer les images de référence est de pouvoir rapidement retrouver celles montrant la même zone qu'une image de test donnée, de façon à permettre à l'analyste image d'effectuer une comparaison visuelle des images réelles. Pour cela, la base de données indexe les images de référence par rapport à leur empreinte au sol grâce à une structure de R-Tree. Lors de l'ingestion d'une nouvelle vidéo dans la base de données, deux étapes successives sont mises en œuvre : le calcul des empreintes au sol de chaque image et la génération du R-Tree indexant spatialement ces empreintes au sol.

Calcul des empreintes au sol L'empreinte au sol de chaque image de la vidéo de référence considérée peut être calculée grâce à la connaissance des paramètres d'acquisition de l'image (voir section 3.1.1) ainsi que du MNS correspondant à la région observée.

Étant donné un pixel localisé sur la bordure de l'image considérée, nous utilisons un algorithme de lancer de rayon (*ray casting* dans la littérature) pour déterminer le point de l'espace représentant l'intersection entre la surface définie par le MNS et le rayon lumineux issu de la caméra au pixel considéré. Ce procédé permet de déterminer une séquence de points dans l'espace, qui représente l'empreinte au sol de l'image courante. En pratique, cette séquence peut être constituée de nombreux points alignés, ce qui peut alors alourdir inutilement les calculs. Par conséquent, nous utilisons un mécanisme de filtrage qui n'insère un nouveau point dans la séquence que lorsque ce point génère un angle important par rapport au dernier segment de la séquence de points (voir l'algorithme 4.1). Enfin, la boite englobante de cette empreinte au sol est déterminée grâce aux bornes inférieures et supérieures des coordonnées des points de

2. Aujourd'hui, les données d'altitude du sol (MNT) sont disponibles pour n'importe quelle région du monde. Par exemple, la NASA fournit gratuitement les relevés altimétriques issus des Shuttle Radar Topography Missions (SRTM) à l'adresse suivante `http://dds.cr.usgs.gov/srtm/version2_1`.

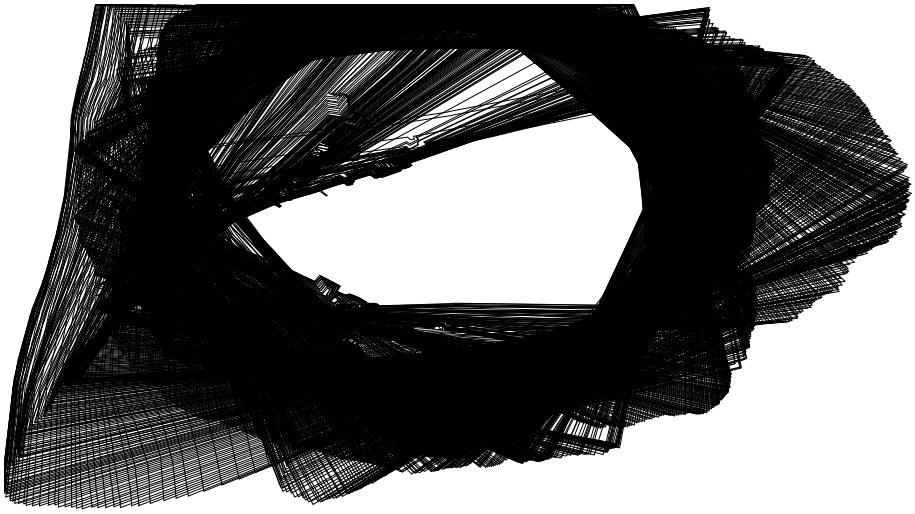


FIGURE 4.5 – *Cette figure présente les empreintes au sol des images de la vidéo* Aérodrome 2 *utilisée comme référence pour les évaluations du chapitre 6.*

Entrées : Paramètres d'acquisition de l'image considérée I, MNS de la région observée, Distance minimale d_{min} entre deux points successifs, Angle minimum θ_{min} entre deux segments successifs
Sorties : Empreinte au sol E de l'image I

1: Déterminer l'ensemble Q de pixels en bordure de l'image I
2: $E \leftarrow \varnothing$
3: **Pour** Chaque pixel $q \in Q$ **Faire**
4: Calculer la direction du rayon r_q issu de la caméra au pixel q
5: Déterminer le point d'intersection $\mathbf{M}_q$ entre la surface du MNS et le rayon r_q
6: **Si** $|E| > 1$ et Distance $\big(\mathbf{M}_q, \text{DernierPoint}(E)\big) > d_{min}$ **Alors**
7: **Si** $|E| > 2$ et Angle $\Big[\text{Segment}\big(\mathbf{M}_q, \text{DernierPoint}(E)\big), \text{DernierSegment}(E)\Big] > \theta_{min}$ **Alors**
8: $E \leftarrow E \cup \{\mathbf{M}_q\}$
9: **Sinon**
10: **Si** $|E| > 2$ **Alors**
11: $E \leftarrow E \backslash \big\{\text{DernierPoint}(E)\big\} \cup \{\mathbf{M}_q\}$
12: **Sinon**
13: $E \leftarrow E \cup \{\mathbf{M}_q\}$
14: **Fin Si**
15: **Fin Si**
16: **Sinon**
17: **Si** $|E| = 0$ **Alors**
18: $E \leftarrow E \cup \{\mathbf{M}_q\}$
19: **Fin Si**
20: **Fin Si**
21: **Fin Pour**

ALGORITHME 4.1 – *Algorithme de calcul de l'empreinte au sol d'une image donnée.*

la séquence, selon les deux axes horizontaux Ox et Oy. Le pseudo-code de cette méthode est fourni par l'algorithme 4.1. La figure 4.5 présente l'exemple des empreintes au sol obtenues sur la vidéo *Aérodrome 2*, utilisée comme référence pour les évaluations du chapitre 6. Cette figure montre que ces empreintes suivent le relief du sol, et y compris celui des bâtiments lorsque ceux-ci sont présents dans le MNS.

Génération de la structure arborescente Pour l'indexation spatiale d'un ensemble d'empreintes au sol dans un R-Tree, nous utilisons l'algorithme Sort-Tile-Recursive (STR) [66], choisi pour sa simplicité d'implémentation et ses bonnes performances d'indexation. De plus, cet algorithme étant extrêmement rapide, la structure du R-Tree peut être recalculée à chaque fois qu'une nouvelle vidéo de référence est ingérée.

Le principe de l'algorithme STR est de construire la structure arborescente itérativement en groupant les nœuds jusqu'à obtenir un nœud unique, racine de l'arbre final. Soit N_{fils} le nombre maximal de descendants par nœud, chaque nœud étant caractérisé par un rectangle englobant tous ses descendants. Cet algorithme est initialisé en affectant un nœud feuille à chaque empreinte au sol et à son rectangle englobant. À chaque itération i de l'algorithme, l'ensemble des $N^i_{\text{nœuds}}$ nœuds disponibles est trié selon l'axe Ox, en utilisant les coordonnées des centres des rectangles englobants. Cet ensemble trié est ensuite partitionné en $N^i_{\text{tranches}} = \lceil \sqrt{\frac{N^i_{\text{nœuds}}}{N_{\text{fils}}}} \rceil$ tranches verticales, chacune constituées de $N^i_{\text{tranches}} \cdot N_{\text{fils}}$ nœuds successifs. Les nœuds de chaque tranche verticale sont ensuite triés selon l'axe Oy et chaque groupe de N_{fils} nœuds successifs est utilisé pour définir les descendants d'un nouveau nœud qui est ajouté à un nouvel ensemble de nœuds disponibles pour l'itération suivante. Enfin, lorsque le nombre de nœuds disponibles est inférieur ou égal à N_{fils}, la racine de l'arbre est créée et les nœuds restants sont définis comme ses descendants. Le fort taux de recouvrement des images, comme l'illustre la figure 4.5, engendre un fort recouvrement des nœuds du R-Tree, ce qui rend la visualisation de la structure difficile. Cependant, l'utilisation du R-Tree est particulièrement importante lorsque la scène observée est vaste et que les images n'en observent qu'une petite partie à la fois.

Notons qu'une fois le R-Tree créé, sa racine contient le rectangle englobant l'ensemble des empreintes au sol des données de référence ingérées dans la base. Cela permet donc de définir l'étendue géographique des données de référence, qui pourra être utilisée par la suite, notamment pour la définition du Quad-Tree augmenté.

4.2.1.2 Organisation spatiale des modèles d'apparence

Pour l'indexation des observations de référence, ou en d'autres termes, l'organisation des modèles d'apparence, la base de données tri-dimensionnelle utilise un Quad-Tree augmenté, combinant un Quad-Tree avec une carte d'élévation. Le Quad-Tree permet de définir une subdivision bi-dimensionnelle de la scène en cellules dont la finesse est adaptable en fonction des données disponibles. Cette subdivision représente la scène selon le plan horizontal Oxy, représentation qui est augmentée par l'association, typique d'une représentation par carte d'élévation, d'une altitude à chaque cellule.

Lors de l'ingestion d'une nouvelle vidéo dans la base de données, la structure utilisée pour organiser les modèles d'apparence doit être initialisée ou mise-à-jour. Cela met en œuvre trois étapes successives : la définition de la racine du Quad-Tree en cohérence avec l'étendue géographique de la scène observée, l'adaptation aux nouvelles observations de la finesse de la subdivision du Quad-Tree et enfin le calcul des élévations associées aux nouvelles cellules de la subdivision.

Définition de la racine La première étape pour l'indexation des observations d'une nouvelle vidéo dans la base de données consiste à positionner la racine du Quad-Tree de manière à englober la région observée. Pour cela, deux cas sont possibles : soit le Quad-Tree n'existe pas

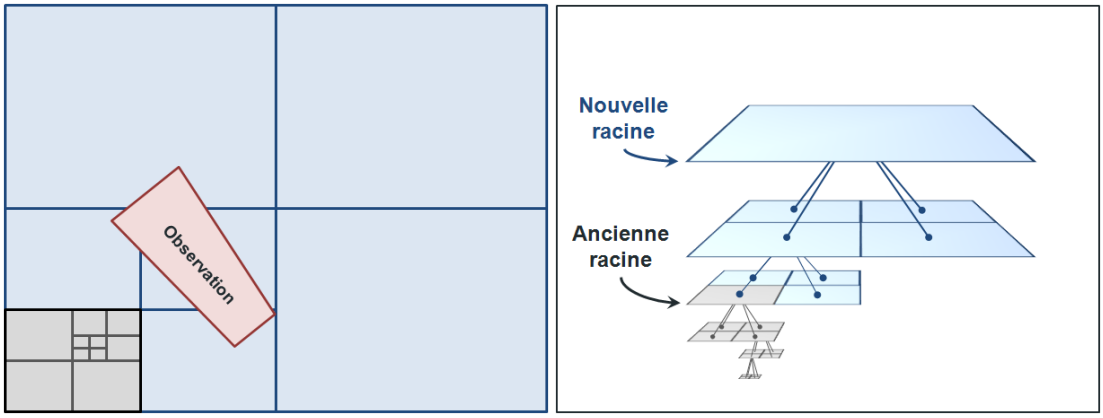


FIGURE 4.6 – *Cette figure illustre la procédure permettant de définir une nouvelle structure de Quad-Tree (en bleu) compatible à la fois avec une nouvelle observation (en rouge) et une structure existante (en gris). L'image de gauche (a) montre l'évolution de la subdivision bidimensionnelle du Quad-Tree, et le schéma de droite (b) montre l'évolution de la structure arborescente associée.*

encore et il est alors nécessaire de le créer, soit il existe déjà et il faut le mettre à jour. Dans le premier cas, il suffit simplement de créer un Quad-Tree vide dont la racine englobe exactement l'étendue géographique définie par le R-Tree, dont la construction est présentée à la section précédente.

Dans le second cas, le Quad-Tree existe déjà et il est alors nécessaire de définir une nouvelle racine qui englobe à la fois l'étendue géographique correspondant à la nouvelle vidéo et celle du Quad-Tree existant. En effet, afin d'éviter de devoir traiter à nouveau l'ensemble des observations passées, il est nécessaire de conserver la structure du Quad-Tree existant. Cela peut être fait en développant le Quad-Tree vers le haut, comme illustré à la figure 4.6. Notons que le fait d'utiliser une structure arborescente, dont la résolution est adaptative contrairement à une simple grille, permet de rendre ce procédé très peu coûteux, car n'engendrant ni gâchis de mémoire ni copie ou re-traitement des données existantes.

Adaptation de la résolution Une fois que la racine du Quad-Tree est définie de manière à englober la région observée, sa structure arborescente est adaptée aux données disponibles. Plus précisément, l'objectif consiste à ajuster la finesse des cellules du Quad-Tree là où la résolution au sol des observations est la plus fine, tout en utilisant une grosse maille de modélisation pour les zones non observées ou correspondant à des observations peu résolues.

Pour cela, nous avons développé un algorithme permettant d'adapter rapidement la résolution du Quad-Tree (ou en d'autres termes, sa profondeur), sans analyser chaque pixel de chaque image considérée [65]. Pour cela, notre algorithme vérifie de manière récursive que la structure arborescente du Quad-Tree permet de décrire avec suffisamment de précision chaque image de la vidéo à ingérer, et lorsque ce n'est pas le cas, la structure arborescente est développée vers le bas jusqu'à atteindre la précision de description requise. En particulier, nous utilisons trois cas d'arrêt permettant d'éviter de parcourir la totalité de la structure pour chaque image. Le premier cas d'arrêt vérifie qu'il y a bien intersection entre la cellule courante du Quad-Tree et l'image considérée, afin de ne pas explorer les branches inutiles par rapport à l'image considérée. Le second cas d'arrêt compare la résolution minimale exigée, l_{min}, qui est un paramètre de l'algorithme, avec la résolution maximale des descendants de la cellule courante. Cela permet d'éviter d'explorer une branche lorsqu'elle ne pourra de toute façon pas être développée plus. Le dernier cas d'arrêt compare l'aire de la projection de la cellule courante dans l'image considérée avec la résolution de l'image, ce qui permet d'éviter d'explorer une branche lorsqu'il n'est pas nécessaire de la développer plus. Lorsque ces trois cas d'arrêt autorisent l'exploration

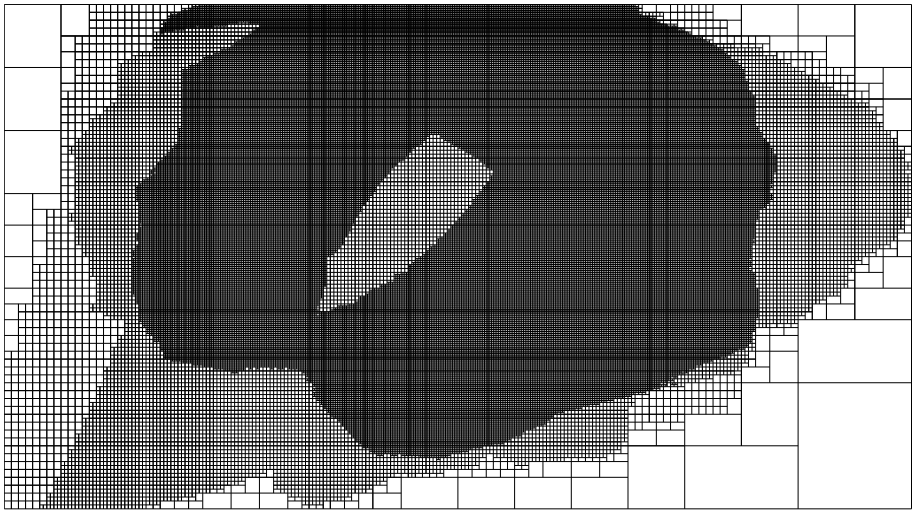


FIGURE 4.7 – *Cette figure présente la subdivision du Quad-Tree généré à partir de la même vidéo que les empreintes au sol présentées à la figure 4.5. Remarquons que la finesse de modélisation est moindre au centre de la zone observée, ce qui est dû au fait que la résolution au sol est plus importante en bas des images qu'en haut. D'autre part, la forme en C est due à la trajectoire circulaire suivie par la plate-forme d'acquisition.*

d'une branche, le corps de notre algorithme développe la structure arborescente vers le bas et effectue les appels récursifs vers les descendants. Le pseudo-code de cette méthode est fourni par l'algorithme 4.2.

La figure 4.7 présente un exemple de subdivision obtenue grâce à cet algorithme d'adaptation de la résolution, avec une résolution minimale de cellule de 25cm au sol et une aire de la cellule dans l'image correspondant à un groupe de 3×3 pixels. Cette visualisation montre que la structure résultante, constituée d'environ 142 000 cellules feuilles, est volumineuse, ce qui illustre le nécessaire compromis à trouver entre la place mémoire occupée par la base de données et la précision de modélisation possible. Afin de minimiser l'occupation mémoire, certaines approches de la littérature [32] proposent d'effectuer une compression a posteriori du modèle en fusionnant les cellules adjacentes ayant un modèle d'apparence et une élévation proches. Cette idée semble effectivement intéressante dans un contexte de modélisation de la scène observée, en revanche elle n'est pas adaptée au cas de la détection de changements. En effet, selon les modèles d'apparence utilisés, cette diminution de la résolution peut également s'accompagner d'une diminution de la performance de détection, en particulier pour les changements de petite taille.

Calcul des élévations Enfin, il est nécessaire de définir l'élévation associée aux cellules nouvellement créées. Ces élévations peuvent être initialisées à l'aide d'information a priori, par exemple grâce à un MNS disponible ou avec une hypothèse de sol à altitude constante. Par la suite, ces élévations peuvent éventuellement être affinées à l'aide d'un algorithme de reconstruction 3D dense [34, 94]. Cependant, nous montrons à la section 6.2.2 qu'en milieu faiblement urbain, les performances obtenues en utilisant un MNT ou un MNE sont sensiblement similaires, ce qui laisse penser qu'un affinage des élévations par reconstruction 3D est superflu.

L'intégration d'élévations issues d'un MNS dans les cellules du Quad-Tree nécessite malgré tout un traitement adapté. En effet, une cellule donnée du Quad-Tree peut être plus grande ou plus petite que le pas d'échantillonnage du MNS et il sera donc nécessaire de procéder respectivement à une moyenne ou une interpolation des élévations sur la zone géographique correspondante. D'autre part, afin de permettre un accès efficace aux cellules lors de requêtes spatiales, il est important de conserver dans chaque cellule les bornes inférieure et supérieure

Entrées : Matrice de projection P_I et empreinte au sol E_I de l'image I considérée, cellule du Quad-Tree $\mathcal{C}$ considérée par l'appel récursif courant, profondeur minimale $\text{prof}_{\min}(\mathcal{C})$ parmi les branches du sous-arbre de $\mathcal{C}$, MNS de la région observée, taille de cellule minimale l_{min}, nombre de pixels maximum par cellule N_{pixels}
Sorties : Structure arborescente adaptée à la résolution de l'image considérée

1: Calculer la projection de $\mathcal{C}$ dans I à l'aide de P_I et du MNS de la zone
2: **Si** Intersection$(\mathcal{C}, E_I) = \varnothing$ **Alors**
3: **Retourner** $\text{prof}_{\min}(\mathcal{C})$
4: **Sinon Si** Resolution$(\mathcal{C})/2^{\text{prof}_{\min}(\mathcal{C})} < l_{min}$ **Alors**
5: **Retourner** $\text{prof}_{\min}(\mathcal{C})$
6: **Sinon Si** Aire(projection de $\mathcal{C}$ dans I) $< N_{\text{pixels}}$ **Alors**
7: **Retourner** $\text{prof}_{\min}(\mathcal{C})$
8: **Fin Si**
9: $\text{prof}_{\min}(\mathcal{C}) \leftarrow \infty$
10: **Pour** Chaque descendant $\mathcal{D}$ de $\mathcal{C}$ **Faire**
11: **Si** $\mathcal{D}$ n'est pas défini **Alors**
12: Créer $\mathcal{D}$
13: $\text{prof}_{\min}(\mathcal{D}) \leftarrow 0$
14: **Fin Si**
15: $\text{prof}_{\min}(\mathcal{C}) \leftarrow \text{MIN}\left[\text{prof}_{\min}(\mathcal{C}), 1 + \text{Résultat de l'appel récursif sur } \mathcal{D}\right]$
16: **Fin Pour**
17: **Retourner** $\text{prof}_{\min}(\mathcal{C})$

ALGORITHME 4.2 – *Algorithme récursif d'adaptation de la résolution du Quad-Tree selon les observations disponibles.*

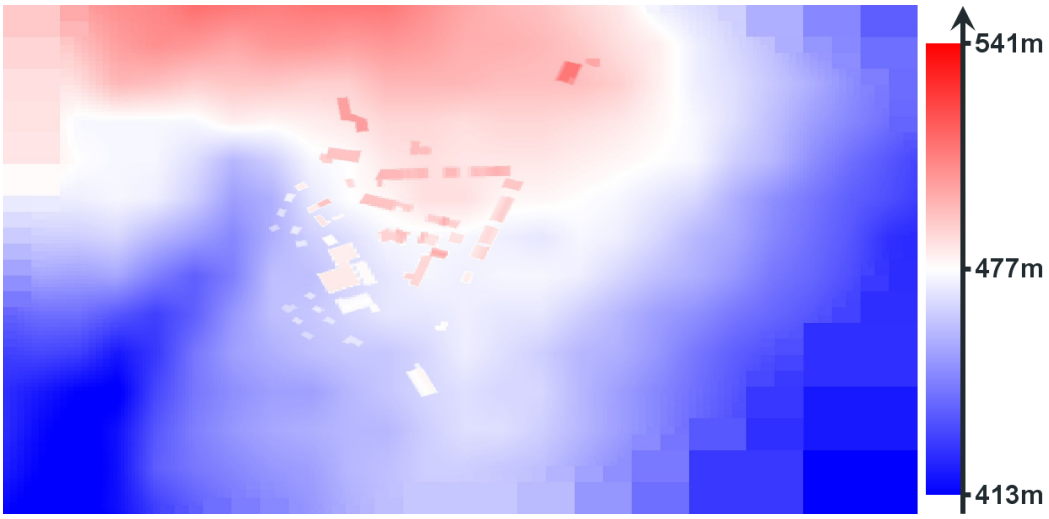


FIGURE 4.8 – *Cette figure présente les élévations associées aux cellules de la subdivision présentée à la figure 4.7.*

Entrées : Empreinte au sol E_I de l'image de test I considérée, nœud du R-Tree $\mathcal{N}$ considéré par l'appel récursif courant
Sorties : Liste L d'images de référence montrant la même zone que l'image de test I

Si $\mathcal{N}$ ne possède pas de descendants **Alors**
  $\mathcal{N}$ contient une image de référence $I_\mathcal{N}$ dont l'empreinte au sol est désignée par $E(I_\mathcal{N})$
  Si Intersection $\left(E_I, E(I_\mathcal{N})\right) \neq \varnothing$ **Alors**
    $L \leftarrow L \cup \{I_\mathcal{N}\}$
  Fin Si
Sinon
  Pour Chaque descendant $\mathcal{D}$ de $\mathcal{N}$ **Faire**
    Si Intersection entre les boites englobantes de $\mathcal{D}$ et de E_I **Alors**
      Appel récursif sur $\mathcal{D}$
    Fin Si
  Fin Pour
Fin Si

ALGORITHME 4.3 – *Algorithme récursif de recherche dans un R-Tree, permettant d'obtenir les images de référence montrant la même zone que l'image de test spécifiée.*

de l'ensemble des élévations des cellules descendantes. La figure 4.8 présente l'exemple des élévations associées à la subdivision présentée à la figure 4.7.

4.2.2 Requêtes spatiales dans les données indexées

Cette section détaille les traitements, mis en œuvre lors de l'exploitation d'une image de test grâce à la base de données, relatifs d'une part à la recherche par le contenu parmi les images de référence indexées dans le R-Tree, et d'autre part aux requêtes sur les modèles d'apparence contenus par le Quad-Tree augmenté. Ces traitements supposent que les paramètres d'acquisition de l'image de test considérée sont connus (voir la section 3.1.2 pour une méthode d'estimation en ligne).

4.2.2.1 Recherche d'images de référence

Comme expliqué précédemment, le travail de l'analyste image porte essentiellement sur les acquisitions brutes, et il est donc important de faciliter la comparaison manuelle entre une image de test et une image de référence montrant la même zone. Pour cela, les images des vidéos de référence sont indexées dans un R-Tree comme expliqué à la section 4.2.1.1. Cette organisation arborescente permet ensuite, grâce à un algorithme de requête, d'accéder efficacement aux images montrant une zone spécifique. Connaissant les paramètres d'acquisition d'une image de test donnée, il est alors possible de calculer l'empreinte au sol de cette image, et d'obtenir les images de référence montrant la même zone géographique que l'image de test.

L'algorithme permettant la recherche des images de référence correspondant à une image de test donnée est très simple. Le pseudo-code correspondant est fourni par l'algorithme 4.3. Le principe de cet algorithme de recherche consiste à explorer une branche de l'arbre uniquement lorsque la boite englobante du nœud courant du R-Tree intersecte la boite englobante de l'empreinte au sol de l'image de test. Lorsque c'est le cas, la recherche est propagée récursivement aux descendants du nœud courant. Si le nœud courant n'a pas de descendant, alors il contient l'empreinte au sol d'une image de référence, qui est ajoutée à la liste des résultats lorsqu'il y a bien intersection entre les deux empreintes au sol.

Notons qu'en présence d'un fort taux de recouvrement entre les images de référence, la liste de résultats renvoyée lors d'une requête peut contenir la quasi-totalité des images de référence de la base. Il est alors nécessaire d'en choisir une à montrer à l'analyste image, par exemple celle

Entrées : Matrice de projection P_I de l'image de sortie
Sorties : Image I contenant le résultat du traitement souhaité

1: **Pour** Chaque pixel q de l'image de sortie I **Faire**
2: Calculer à l'aide de P_I la direction du rayon r_q issu de q
3: Scanner la demi droite définie par le rayon r_q à la recherche de la première cellule occultante $\mathcal{C}_q$
4: **Si** $\mathcal{C}_q$ existe **Alors**
5: Appliquer le traitement souhaité entre q et $\mathcal{C}_q$
6: **Fin Si**
7: **Fin Pour**

ALGORITHME 4.4 – *Algorithme itératif de lancer de rayon classique, permettant d'associer une cellule du Quad-Tree augmenté à chaque pixel de l'image spécifiée.*

Entrées : Matrice de projection P_I de l'image I considérée, cellule du Quad-Tree $\mathcal{C}$ considéré par l'appel récursif courant

1: Déterminer à l'aide de P_I l'enveloppe convexe $E_{\mathcal{C}}$ de la projection de $\mathcal{C}$ dans l'image I
2: **Si** $E_{\mathcal{C}}$ est en dehors de l'image I **Alors**
3: **Retourner**
4: **Fin Si**
5: **Pour** Chaque descendant $\mathcal{D}$ de $\mathcal{C}$ **Faire**
6: **Si** $\mathcal{D}$ est défini **Alors**
7: Appel récursif sur $\mathcal{D}$
8: **Fin Si**
9: **Fin Pour**
10: **Si** $\mathcal{C}$ contient un modèle d'apparence **Alors**
11: Scanner le segment entre $\mathcal{C}$ et la caméra associée à l'image I à la recherche d'une cellule occultante
12: **Si** $\mathcal{C}$ n'est pas occultée **Alors**
13: Calculer l'observation interpolée **o** correspondant à la zone $E_{\mathcal{C}}$ dans l'image I
14: Appliquer le traitement souhaité entre $\mathcal{C}$ et l'observation interpolée **o**
15: **Fin Si**
16: **Fin Si**

ALGORITHME 4.5 – *Algorithme récursif de lancer de rayon inversé, permettant d'associer une observation interpolée à chaque cellule du Quad-Tree augmenté visible dans l'image spécifiée.*

dont l'aire du polygone d'intersection est la plus importante ou celle dont le point de vue est le plus proche. À ce stade, il est également envisageable d'intégrer d'autres critères de filtrage (e.g. date et heure d'acquisition, modalité de l'image, nature de la plate-forme d'acquisition, critère de qualité image, etc) afin de permettre à l'analyste image de spécifier plus précisément l'image de référence avec laquelle il souhaite comparer l'image de test.

4.2.2.2 Accès efficace aux modèles d'apparence

L'accès aux modèles d'apparence contenus dans le Quad-Tree augmenté est une étape cruciale à la fois lors de la modélisation des apparences et lors de la détection de changements. En effet, lors de la modélisation des apparences, il est nécessaire de mettre en correspondance les pixels des images de référence avec les modèles d'apparence correspondants, afin de les mettre à jour. D'autre part, lors de la détection de changements, cette mise en correspondance est également nécessaire afin de permettre la comparaison entre les pixels de l'image de test et les modèles d'apparence correspondants. La méthode utilisée pour permettre un accès efficace aux cellules du Quad-Tree, qui contiennent les modèles d'apparence, dépend du traitement souhaité.

Dans le cas où le résultat du traitement spécifié doit être obtenu dans les coordonnées image, par exemple pour un rendu du modèle ou une détection de changements, l'accès aux cellules est

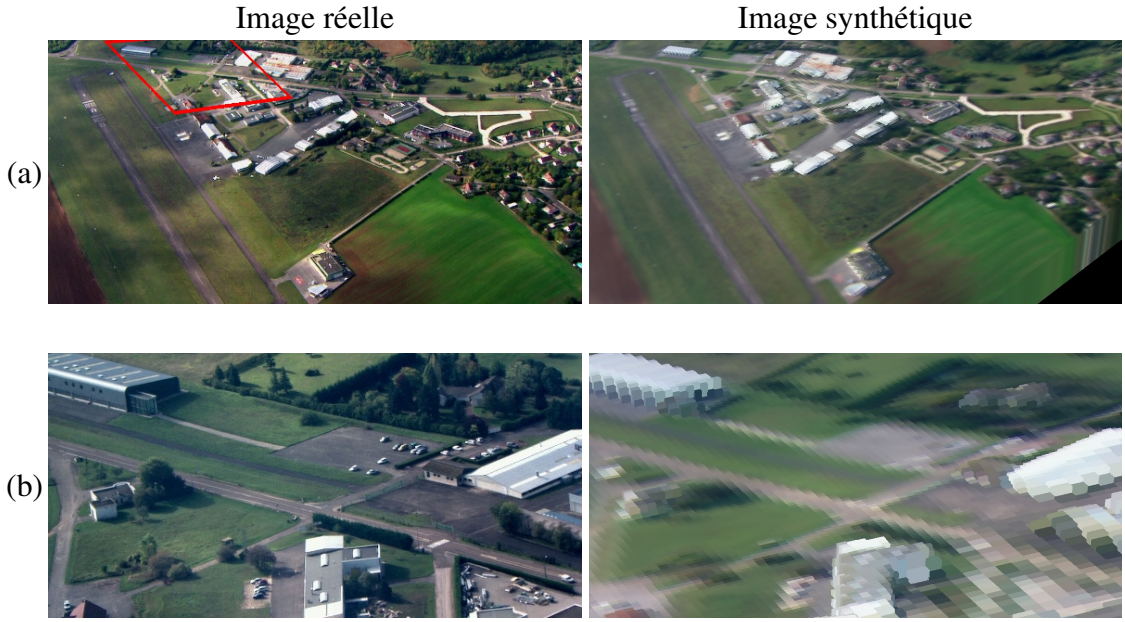


FIGURE 4.9 – *Cette figure compare les images réelles aux images synthétiques, issues des mêmes points de vue, générées grâce au Quad-Tree augmenté, selon que la résolution au sol est comparable (a) ou supérieure (b) à celle des images utilisées pour construire le modèle 3D d'apparence. Le cadre rouge situé dans l'image réelle présentée en (a) permet de visualiser l'emprise de l'image réelle présentée en (b). Copyright © 2010 - 2012 Cassidian - All rights reserved.*

effectué à l'aide d'un algorithme de lancer de rayon classique. Le pseudo-code de cette méthode est fourni par l'algorithme 4.4.

En revanche, dans le cas où le traitement porte sur les modèles d'apparence, typiquement pour leur mise à jour, il est préférable d'effectuer l'accès aux cellules du Quad-Tree à l'aide d'un algorithme de lancer de rayon inversé. En effet, l'algorithme de lancer de rayon classique associe une cellule du Quad-Tree à chaque pixel ou observation de l'image considérée, mais il ne garantit pas qu'inversement, une observation sera associée à chaque cellule du Quad-Tree visible dans l'image. C'est un problème rencontré fréquemment dans le cadre du ré-échantillonnage d'images après transformation géométrique, ce qui peut mener à des "trous" dans l'image finale si le ré-échantillonnage n'est pas effectué correctement. Pour éviter cela, il est préférable d'effectuer un lancer de rayon inversé, c'est-à-dire associant une observation à chaque cellule visible du Quad-Tree. Le pseudo-code de cette méthode est fourni par l'algorithme 4.5.

Ces deux méthodes de mise en correspondance entre les cellules du Quad-Tree augmenté et les observations d'une image donnée font une exploitation intensive de l'algorithme d'intersection [114] entre un rayon et une cellule alignée par rapport aux axes (souvent abrégée *AABB* dans la littérature, pour *Axis-Aligned Bounding Box*). Cet algorithme peut être aisément étendu pour effectuer une recherche efficace des cellules du Quad-Tree augmenté intersectant un rayon donné. Le pseudo-code de la méthode associée est fourni par l'algorithme 4.6.

Pour finir, afin d'illustrer la capacité de représentation d'une scène à l'aide du modèle par Quad-Tree augmenté, la figure 4.9 compare deux images aériennes réelles avec les images synthétiques générées, selon les mêmes points de vue, par rendu du modèle 3D d'apparence. Les intensités utilisées pour générer ces images synthétiques correspondent à la moyenne, en chaque cellule du Quad-Tree augmenté, des observations issues de la vidéo *Aérodrome 2*.

Entrées : Cellule du Quad-Tree $\mathcal{C}$ considéré par l'appel récursif courant, origine $\mathbf{M}_r$ et direction $\mathbf{V}_r$ du rayon spécifié r
Sorties : Liste L des cellules intersectant le rayon spécifié

1: Déterminer la longueur d'intersection $l_{\mathcal{C}\cap r}$ et la distance entre $\mathbf{M}_r$ et $\mathcal{C}$ le long du rayon r ▷ voir [114]
2: **Si** $l_{\mathcal{C}\cap r} > 0$ **Alors**
3: $L \leftarrow L \cup \{\mathcal{C}\}$
4: Trier L par ordre croissant de distance à $\mathbf{M}_r$
5: **Pour** Chaque descendant $\mathcal{D}$ de $\mathcal{C}$ **Faire**
6: **Si** $\mathcal{D}$ est défini **Alors**
7: Appel récursif sur $\mathcal{D}$
8: **Fin Si**
9: **Fin Pour**
10: **Fin Si**

ALGORITHME 4.6 – *Algorithme récursif de recherche des cellules d'un Quad-Tree augmenté intersectant un rayon défini par son origine et sa direction.*

4.3 Modélisation des apparences

La modélisation à proprement parler des apparences observées dans les vidéos de référence peut être effectuée de manière indépendante de leur organisation tri-dimensionnelle. Cette modélisation met en œuvre deux mécanismes essentiels : d'une part la mise à jour incrémentale d'un modèle d'apparence en fonction d'une nouvelle observation de référence, et d'autre part la comparaison entre un modèle d'apparence et une observation de test pour la détection de changements.

Plusieurs techniques de modélisation ont été utilisées dans le cadre de cette thèse et les mécanismes de mise à jour et de comparaison associés à chacune sont décrits dans les sections suivantes. La plupart des techniques utilisées sont issues du domaine de la soustraction de fond, qui, comme nous l'avons vu au chapitre 2, est proche du domaine de la détection de changements mais implique certains a priori différents (relatifs notamment à l'oubli progressif et aux données manquantes aléatoires). Les techniques de modélisation d'apparence utilisées ont donc été adaptées pour répondre aux a priori du domaine qui nous intéresse.

Par ailleurs, de manière à pouvoir employer différentes techniques de modélisation de manière transparente, l'étape de comparaison entre un modèle d'apparence et une observation fournit un score continu de détection de changement compris entre 0 (l'observation ne correspond pas à un changement) et 1 (l'observation correspond à un changement). Par la suite, ce score de détection de changement peut par exemple être comparé à un seuil pour déterminer les pixels correspondant à des changements. Ce seuil peut être variable afin d'évaluer les performances de la technique de modélisation (voir chapitre 6). En exploitation, ce seuil peut également être déterminé à l'aide d'un test statistique d'hypothèse, comme celui mentionné à la section 4.1. Nous verrons par ailleurs au chapitre 5 que le score de détection de changements peut être exploité plus intelligemment que via une simple comparaison avec un seuil.

La suite de cette section présente les détails concernant les différentes techniques de modélisation d'apparence utilisées. Plus précisément, la section 4.3.1 présente la technique de modélisation pixélique par gaussienne unique. La section 4.3.2 présente la technique de modélisation pixélique par mélange de gaussiennes. Enfin, la section 4.3.3 présente la technique de modélisation par analyse incrémentale en composantes principale.

4.3.1 Modèle par gaussienne unique

La technique de modélisation des apparences par gaussienne unique a été employée par les premières techniques visant la soustraction de fond. Cette technique consiste à modéliser la distribution de probabilité des observations successives à l'aide d'une unique gaussienne, qui est estimée de manière incrémentale et indépendamment en chaque pixel. La suite de cette section présente les équations, adaptées pour le domaine de la détection de changements, relatives à la mise à jour du modèle et à son exploitation.

Mise à jour du modèle La mise à jour incrémentale du modèle par gaussienne unique consiste simplement à mettre à jour la moyenne empirique et la matrice de covariance associées à la distribution gaussienne. Soient $\mathbf{o}_t^{(i)} \in \mathbb{R}^D$ et $w_t^{(i)} \in \mathbb{R}$ désignant respectivement l'observation et le coefficient de pondération associés à une cellule d'indice i donné à l'étape t. Ici, D représente la dimension d'une observation (e.g. $D = 3$ pour des observations en couleurs RGB). Nous désignerons respectivement par $W_t^{(i)}$, $\boldsymbol{\mu}_t^{(i)}$ et $\Sigma_t^{(i)}$ la somme des coefficients de pondération, la moyenne empirique et la matrice de covariance des observations jusqu'à l'étape t incluse. Pour plus de simplicité, cette méthode étant appliquée de manière indépendante en chaque cellule, l'indice i sera sous-entendu dans la suite de cette section. Les équations de mise à jour incrémentale sont alors les suivantes :

$$\begin{array}{l} \text{Pour } t = 0 : \\ \left\{ \begin{array}{l} W_0 = w_0 \\ \boldsymbol{\mu}_0 = \mathbf{o}_0 \\ \Sigma_0 = 0_{D\times D} \end{array} \right. \end{array} \quad \text{et} \quad \begin{array}{l} \text{Pour } t \geqslant 1 : \text{ soit } \alpha_t = \frac{w_t}{W_{t-1}+w_t} \\ \left\{ \begin{array}{l} W_t = W_{t-1} + w_t \\ \boldsymbol{\mu}_t = \boldsymbol{\mu}_{t-1} + \alpha_t \cdot (\mathbf{o}_t - \boldsymbol{\mu}_{t-1}) \\ \Sigma_t = (1-\alpha_t) \cdot [\Sigma_{t-1} + \alpha_t \cdot (\mathbf{o}_t - \boldsymbol{\mu}_{t-1})(\mathbf{o}_t - \boldsymbol{\mu}_{t-1})^T] \end{array} \right. \end{array} \tag{4.6}$$

Détection de changements Cette technique de modélisation permettant d'estimer une distribution de probabilité sur les observations, il est naturel de l'exploiter pour estimer la probabilité qu'une nouvelle observation corresponde à un changement. Soient W, $\boldsymbol{\mu}$ et Σ désignant respectivement la somme des coefficients de pondération, la moyenne empirique et la matrice de covariance de l'ensemble des observations de référence [3]. Soit enfin $\mathbf{o} \in \mathbb{R}^D$ une observation de test, dont nous souhaitons déterminer si elle correspond à un changement. Le score de détection de changements, désigné par ε, est alors obtenu grâce à l'équation suivante, issue de la normalisation entre 0 et 1 de la densité de probabilité associée à la gaussienne d'espérance $\boldsymbol{\mu}$ et de covariance Σ :

$$\varepsilon_{\text{lg}}(\mathbf{o}) = 1 - \exp\left[-\frac{1}{2} \cdot (\mathbf{o}-\boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{o}-\boldsymbol{\mu})\right] \tag{4.7}$$

4.3.2 Modèle par mélange de gaussiennes

La technique de modélisation des apparences par mélange de gaussiennes a été introduite pour le domaine de la soustraction de fond par Stauffer et Grimson [102], afin de permettre une plus grande précision de modélisation pour les environnements dynamiques. Cette technique consiste à modéliser la distribution de probabilité des observations successives à l'aide d'une somme pondérée de distributions gaussiennes, qui est estimée de manière incrémentale et indépendamment en chaque pixel. Le fait d'utiliser plusieurs gaussiennes permet de modéliser précisément des comportements dynamiques mais répétitifs (en d'autres termes, multi-modaux) dans les observations successives, dus par exemple à une branche d'arbre qui bouge, à un écran qui scintille, aux réflexions spéculaires des vagues sur un plan d'eau, etc. La figure 4.10 illustre,

3. Notons que ces grandeurs sont toujours implicitement indexées par rapport à la cellule d'indice i.

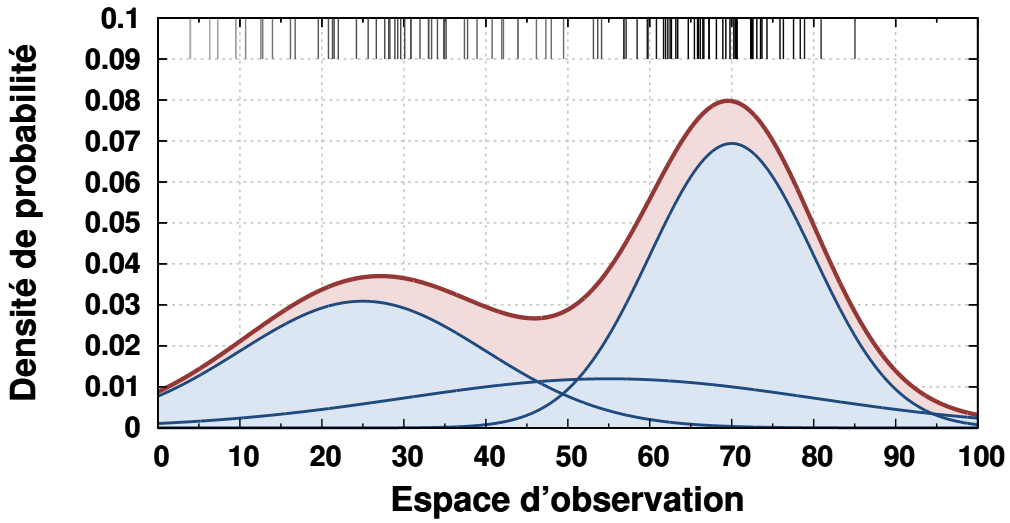


FIGURE 4.10 – *Cette figure illustre, dans le cas uni-dimensionnel, l'allure d'une distribution par mélange de gaussiennes (en rouge) et un ensemble d'observations issues de cette distribution (traits noirs du haut). Cette distribution consiste en une somme pondérée de trois distributions gaussiennes (en bleu), dont les coefficients de pondération peuvent être non-uniformes.*

dans le cas uni-dimensionnel, l'allure d'une distribution par mélange de gaussiennes et un ensemble d'observations issues de celle-ci. La suite de cette section présente les équations, adaptées pour le domaine de la détection de changements, relatives à la mise à jour du modèle et à son exploitation.

Mise à jour du modèle La mise à jour incrémentale du modèle par mélange de gaussiennes consiste à affecter chaque observation à l'une des gaussiennes de la somme pondérée (ou à en créer une nouvelle si aucune ne correspond), puis à mettre à jour cette gaussienne grâce aux mêmes équations que celles utilisées pour le modèle par gaussienne unique. Soit N_t le nombre de gaussiennes à l'étape t. Pour une cellule donnée d'indice i, désignons par $\left\{\mathcal{G}_{t,k}^{(i)}\right\}_{k\in[\![1,N_t]\!]}$ les gaussiennes de la somme pondérée à l'étape t, et par $\left\{\omega_{t,k}^{(i)}\right\}_{k\in[\![1,N_t]\!]}$ les coefficients leur correspondant dans la somme pondérée. Comme à la section précédente, cette méthode est appliquée de manière indépendante pour chaque cellule, et par conséquent nous considérerons toutes les grandeurs utilisées dans la suite de cette section comme implicitement indexées par rapport à i. L'affectation d'une observation à l'une des gaussiennes se fait en parcourant les gaussiennes par ordre décroissant de vraisemblance d'occurrence, qui est définie pour une gaussienne $\mathcal{G}_{t,k}$ comme étant proportionnelle au rapport entre son coefficient de pondération $\omega_{t,k}$ et le déterminant de sa matrice de covariance $\Sigma_{t,k}$. L'observation $\mathbf{o}_t \in \mathbb{R}^D$ à l'étape t, associée au coefficient de pondération w_t, est alors affectée à la première gaussienne $\mathcal{G}_{t,k}$ constituant un appariement valide avec l'observation, c'est à dire pour laquelle la distance Mahalanobis entre $\mathbf{o}_t$ et $\boldsymbol{\mu}_{t,k}$ est inférieure à un seuil. Une valeur de 2.5 est généralement utilisée pour ce seuil, et c'est ce que nous utilisons en pratique. Si aucune gaussienne ne constitue un appariement valide avec l'observation courante, alors une nouvelle gaussienne est créée et initialisée avec l'observation courante. Pour éviter que le nombre de gaussiennes n'augmente trop, il est limité à trois gaussiennes. Pour cela, lorsque le nombre maximal de gaussiennes est atteint mais qu'il faut en créer une nouvelle, celle associée au coefficient de pondération le plus faible est supprimée avant de créer la nouvelle. Le pseudo-code de cette technique de mise à jour est fourni par l'algorithme 4.7. Notons que cette méthode de modélisation est moins précise qu'une méthode de

Entrées : Observation courante $\mathbf{o}_t$ associée au coefficient de pondération w_t, gaussiennes $\left\{\mathcal{G}_{t-1}^k\right\}_{k\in[\![1,N_{t-1}]\!]}$ et leurs coefficients de pondération $\left\{\omega_{t-1}^k\right\}_{k\in[\![1,N_{t-1}]\!]}$ à l'étape précédente
Sorties : Gaussiennes $\left\{\mathcal{G}_t^k\right\}_{k\in[\![1,N_t]\!]}$ et leurs coefficients de pondération $\left\{\omega_t^k\right\}_{k\in[\![1,N_t]\!]}$ à l'étape courante

$\alpha \leftarrow \frac{w_t}{w_t+\sum_k w_{t-1}^k}$
Pour $k \in [\![1, N_{t-1}]\!]$ **Faire**
    $\omega_t^k \leftarrow (1-\alpha)\cdot\omega_{t-1}^k$
    Si $\mathcal{G}_{t-1}^k$ est la première gaussienne constituant un appariement valide avec $\mathbf{o}_t$ **Alors**
        Mettre à jour $\mathcal{G}_{t-1}^k$ selon l'équation 4.6
        $\omega_t^k \leftarrow \omega_t^k + \alpha$
    Fin Si
Fin Pour
$N_t \leftarrow N_{t-1}$
Si Aucune gaussienne ne constitue un appariement valide avec $\mathbf{o}_t$ **Alors**
    Si le nombre maximum de gaussiennes est atteint **Alors**
        Supprimer la gaussienne associée au coefficient de pondération le plus faible
        $N_t \leftarrow N_t - 1$
    Fin Si
    Ajouter une nouvelle gaussienne $\mathcal{G}_t^{1+N_t}$ initialisée selon l'équation 4.6
    $N_t \leftarrow 1 + N_t$
Fin Si
Réordonner les gaussiennes $\left\{\mathcal{G}_t^k\right\}_{k\in[\![1,N_t]\!]}$ par ordre décroissant de vraisemblance d'occurrence

ALGORITHME 4.7 – *Technique de mise à jour incrémentale de la modélisation par mélange de gaussiennes à l'aide de l'observation courante.*

type Espérance-Maximisation, mais qu'elle peut être effectuée de manière incrémentale, ce qui constitue une nécessité vu le volume des données à traiter.

Détection de changements Comme pour la technique précédente, cette technique de modélisation permettant d'estimer une distribution de probabilité sur les observations, il est naturel de l'exploiter pour estimer la probabilité qu'une observation de test corresponde à un changement. Soit N le nombre de gaussiennes utilisées une fois terminée la modélisation de l'ensemble des observations de référence, et soient $\left\{\mathcal{G}^k\right\}_{k\in[\![1,N]\!]}$ les gaussiennes de la somme pondérée. Soit enfin $\mathbf{o} \in \mathbb{R}^D$ une observation de test, dont nous souhaitons déterminer si elle correspond à un changement. De manière intuitive, une observation ne doit être déclarée comme changement potentiel que si elle ne correspond à aucune des observations de référence. Par conséquent, nous proposons de réutiliser le score de détection de changements défini à la section précédente pour les gaussiennes uniques, et d'associer à $\mathbf{o}$ le score de détection de changements minimum parmi les scores calculés par rapport à chaque gaussienne. Ceci débouche donc sur l'expression suivante :

$$\varepsilon_{\text{gmm}}(\mathbf{o}) = \min_k \left\{\varepsilon_{\text{lg},\mathcal{G}^k}(\mathbf{o})\right\} \tag{4.8}$$

4.3.3 Analyse incrémentale en composantes principales

La technique de modélisation des apparences par analyse incrémentale en composantes principales a été introduite pour le domaine de la soustraction de fond par Li [67]. Le principe de cette technique, inspirée de la technique des *eigenfaces* pour la reconnaissance de visages [109], consiste à considérer une image donnée comme un gigantesque vecteur colonne. L'espace de variation des vecteurs correspondant à l'ensemble des images des vidéos de référence est alors analysé à l'aide d'une analyse en composantes principales (ACP) afin d'en déterminer

les modes de variation principaux. En pratique, la grande taille des vecteurs considérés combinée au nombre important d'images de référence rend impossible le calcul direct de la matrice de covariance associée, dont la connaissance est nécessaire pour effectuer une ACP classique. L'idée est donc d'effectuer une ACP approchée, mise à jour incrémentalement pour chaque image de référence.

L'utilisation de cette technique pour le domaine de la détection de changements requiert plusieurs adaptations. Pour commencer, contrairement au domaine de la soustraction de fond, la totalité des images de référence sont connues et il est donc possible de les parcourir en ordre aléatoire et non pas séquentiellement. Du fait que l'ACP incrémentale implique une nécessaire approximation, le parcours aléatoire des images permet d'éviter que l'effort de modélisation ne porte que sur une portion non représentative des images (voir l'algorithme de mise à jour). D'autre part, les équations de mise à jour proposées par Li [67] impliquent un oubli progressif des données passées dans le calcul approché de la matrice de covariance. Cet oubli progressif étant indésirable dans le cadre de la détection de changements, il a été supprimé des équations proposées plus bas. Enfin, la technique proposée par Li [67] inclut une version robuste permettant de tolérer les observations aberrantes grâce à un raisonnement sur les résidus. Cette version robuste peut être exploitée [4] pour traiter le cas des données manquantes, dues aux occultations aléatoires causées par les effets géométriques.

Mise à jour du modèle L'algorithme de mise à jour de la technique de modélisation des apparences par ACP incrémentale commence par rassembler dans un grand vecteur les observations $\mathbf{o}_t^i \in \mathbb{R}^D$ issues de chaque cellule d'indice i dans le Quad-Tree augmenté. Nous désignerons ce grand vecteur d'observations par $\mathbf{x}_t \in \mathbb{R}^{DN_{\text{cell}}}$ où N_{cell} est le nombre de cellules du Quad-Tree (par exemple, $N_{\text{cell}} = 142\,000$ pour le Quad-Tree illustré à la figure 4.7). Pour représenter le fait que certaines observations puissent être manquantes du fait des occultations et du champ de vue limité, nous utilisons un vecteur de coefficients d'occultation désigné par $\boldsymbol{\delta}_t \in \{0,1\}^{DN_{\text{cell}}}$, où une valeur de coefficient de 0 signifie que l'observation est manquante [5]. En pratique, ce vecteur de coefficients d'occultation résulte de l'exécution de l'algorithme de lancer de rayon inversé (voir algorithme 4.5). Par ailleurs, afin d'associer un poids identique à chaque observation dans le modèle final (i.e. pas d'oubli progressif) d'une cellule d'indice i donné, nous utilisons pour chaque observation $\mathbf{o}_t^i$ un poids dégressif pour la mise à jour incrémentale, égal à $\alpha_t^i = \frac{\delta_t^i}{\sum_{n=0}^{t} \delta_n^i}$, où δ_t^i est le coefficient d'occultation associé à la cellule d'indice i. Notons qu'il est nécessaire de définir un poids de mise à jour distinct pour chaque cellule à cause du caractère aléatoire des occultations possibles. Comme pour les observations, ces poids de mise à jour sont rassemblés dans un grand vecteur de dimension DN_{cell}, désigné par $\boldsymbol{\alpha}_t$. Les équations suivantes (inspirées des équations 4.6) permettent alors de calculer de manière incrémentale les vecteurs de moyenne empirique $\boldsymbol{\mu}_t$ et de variance empirique $\mathbf{v}_t$ des observations passées à l'étape t :

$$\begin{array}{ll} \text{Pour } t = 0 : & \text{Pour } t \geqslant 1 : \\ \left\{ \begin{array}{l} \boldsymbol{\mu}_0 = \mathbf{x}_0 \\ \mathbf{v}_0 = \mathbf{0}_{DN \times 1} \end{array} \right. \quad \text{et} & \left\{ \begin{array}{l} \boldsymbol{\mu}_t = \boldsymbol{\mu}_{t-1} + \boldsymbol{\alpha}_t \circ (\mathbf{x}_t - \boldsymbol{\mu}_{t-1}) \\ \mathbf{v}_t = (1 - \boldsymbol{\alpha}_t) \circ [\mathbf{v}_{t-1} + \boldsymbol{\alpha}_t \circ (\mathbf{x}_t - \boldsymbol{\mu}_{t-1})^{②}] \end{array} \right. \end{array} \tag{4.9}$$

où $\circ$ et ^{②} désignent respectivement la multiplication et la mise au carré élément par élément. La matrice de covariance Σ_t des observations passées à l'étape t, nécessaire pour effectuer l'ACP des observations, est de très grande dimension, égale à $(DN_{\text{cell}}) \times (DN_{\text{cell}})$, et est donc incalculable en pratique. Elle peut cependant être approchée à l'aide des vecteurs propres $\{\mathbf{u}_{t-1}^k\}_{k \in [\![1, N_{\text{CP}}]\!]}$ associés aux N_{CP} plus grandes valeurs propres à l'étape $t-1$, désignées par

4. Il est également possible d'envisager d'exploiter cette version robuste pour traiter le cas des observations aberrantes dans les acquisitions (e.g. erreurs de transmissions, halos et reflets sur les lentilles de la caméra, etc), mais cela n'a pas été testé dans le cadre de cette thèse par manque de données correspondantes.

5. Afin d'aborder le cas des observations aberrantes, il conviendrait de ne pas limiter les valeurs de ces coefficients à des valeurs binaires, mais d'utiliser des valeurs continues déduites des résidus, comme proposé par Li [67].

$\{\lambda_{t-1}^k\}_{k\in[\![1,N_{\text{CP}}]\!]}$. En pratique, nous utilisons $N_{\text{CP}} = 20$. En désignant par $\check{\mathbf{x}}_t = \boldsymbol{\delta}_t \circ (\mathbf{x}_t - \boldsymbol{\mu}_t)$ le vecteur d'observations centré tolérant aux données manquantes, il est alors possible d'obtenir la relation suivante :

$$\begin{aligned}\Sigma_t &= (1 - \frac{1}{t+1}) \cdot \Sigma_{t-1} + \frac{\frac{1}{t+1}}{1 - \frac{1}{t+1}} \cdot \check{\mathbf{x}}_t \check{\mathbf{x}}_t^T \\ &\approx \frac{t}{t+1} \cdot \sum_{k=1}^{N_{\text{CP}}} (\lambda_{t-1}^k \cdot \mathbf{u}_{t-1}^k {\mathbf{u}_{t-1}^k}^T) + \frac{1}{t} \cdot \check{\mathbf{x}}_t \check{\mathbf{x}}_t^T \\ &= \mathrm{A}_t {\mathrm{A}_t}^T\end{aligned} \tag{4.10}$$

$$\text{avec} \qquad \mathrm{A}_t = \left[\begin{array}{cccc} \mathbf{y}_1 & \dots & \mathbf{y}_{N_{\text{CP}}} & \sqrt{\frac{1}{t}} \cdot \check{\mathbf{x}}_t \end{array} \right] \tag{4.11}$$

$$\mathbf{y}_k = \sqrt{\frac{t}{t+1} \lambda_{t-1}^k} \cdot \mathbf{u}_{t-1}^k$$

Pour terminer la mise à jour incrémentale, il est nécessaire de calculer les nouveaux vecteurs propres $\{\mathbf{u}_t^k\}_{k\in[\![1,N_{\text{CP}}]\!]}$ associés aux N_{CP} valeurs propres $\{\lambda_t^k\}_{k\in[\![1,N_{\text{CP}}]\!]}$ les plus grandes à l'étape t, ce qui requiert de diagonaliser la matrice $\mathrm{A}_t {\mathrm{A}_t}^T$. Or, comme la matrice de covariance Σ_t, cette matrice est de très grande dimension et est également incalculable en pratique. Par conséquent, la matrice ${\mathrm{A}_t}^T \mathrm{A}_t$, qui est de dimension $(N_{\text{CP}}+1) \times (N_{\text{CP}}+1)$ beaucoup plus abordable, est diagonalisée à la place. En effet, les valeurs propres de cette matrice sont les mêmes que celles de la matrice $\mathrm{A}_t {\mathrm{A}_t}^T$, et ses vecteurs propres associés aux N_{CP} plus grandes valeurs propres, notés $\{\mathbf{u'}_t^k\}_{k\in[\![1,N_{\text{CP}}]\!]}$, sont liés à ceux que nous cherchons par la relation suivante :

$$\text{Pour tout } k \in [\![1, N_{\text{CP}}]\!],\ \mathbf{u}_t^k = \mathrm{A}_t \mathbf{u'}_t^k \tag{4.12}$$

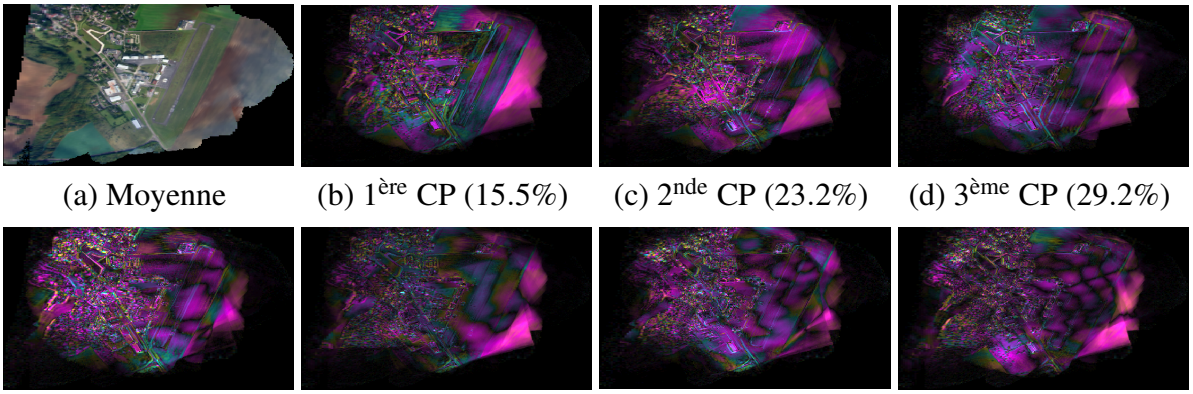
Par conséquent, il est possible de mettre à jour directement les vecteurs propres $\{\mathbf{u}_t^k\}_{k\in[\![1,N_{\text{CP}}]\!]}$ et les valeurs propres $\{\lambda_t^k\}_{k\in[\![1,N_{\text{CP}}]\!]}$ de manière incrémentale, à l'aide de la diagonalisation de la matrice ${\mathrm{A}_t}^T \mathrm{A}_t$. Ce mécanisme de mise à jour incrémentale des vecteurs et valeurs propres est mis en œuvre pour les étapes $t \geqslant N_{\text{CP}}$, et initialisé à l'étape $t_0 = N_{\text{CP}} - 1$ à l'aide de la diagonalisation de la matrice $\mathrm{A}^T \mathrm{A}$, où :

$$\mathrm{A} = \left[\begin{array}{ccc} \boldsymbol{\delta}_0 \circ (\mathbf{x}_0 - \boldsymbol{\mu}_{t_0}) & \dots & \boldsymbol{\delta}_{t_0} \circ (\mathbf{x}_{t_0} - \boldsymbol{\mu}_{t_0}) \end{array} \right] \tag{4.13}$$

Notons qu'à chaque itération, le vecteur propre correspondant à la valeur propre la plus faible est ignoré, afin de conserver un nombre fixe de vecteurs propre, égal à N_{CP}. Cela introduit une approximation par rapport au résultat de l'ACP classique, résultat ne pouvant pas être calculé du fait de la dimension trop importante de la matrice de covariance. Il est cependant possible de mesurer l'approximation commise par rapport au résultat idéal, grâce au calcul de la variance des observations. En effet, dans le cas d'une ACP classique effectuée sur une matrice de covariance Σ, la variance totale est égale à la somme de toutes les valeurs propres issues de la diagonalisation $\Sigma = \mathrm{U}^T \Lambda \mathrm{U}$. Ceci débouche donc sur la relation suivante :

$$\sum_{k=1}^{\dim(\Sigma)} \lambda_k = \text{trace}(\Lambda) = \text{trace}(\Sigma) \tag{4.14}$$

Par conséquent, la variance totale est égale à la somme des éléments diagonaux de la matrice de covariance. De retour à notre ACP incrémentale, il est donc possible de mesurer l'approximation commise en comparant la somme des N_{CP} valeurs propres finales à la somme des éléments du vecteur final de variance empirique $\mathbf{v}$. La figure 4.11 montre quelques-unes des premières composantes principales, obtenues à l'aide des mêmes données que pour les illustrations précédentes de ce chapitre, et mentionne les fractions cumulatives de variance expliquée par rapport à la variance totale (appelées énergies cumulatives).



(a) Moyenne (b) 1ère CP (15.5%) (c) 2nde CP (23.2%) (d) 3ème CP (29.2%)

(e) 4ème CP (33.9%) (f) 5ème CP (37.0%) (g) 6ème CP (39.3%) (h) 7ème CP (40.9%)

FIGURE 4.11 – *Cette figure présente l'apparence moyenne (a) et les sept premières composantes principales (b) à (h), obtenues par ACP incrémentale sur les mêmes données que pour les illustrations précédentes. Les énergies cumulatives sont fournies entre parenthèses pour chaque composante, et l'énergie cumulative finale, tenant compte des $N_{CP} = 20$ composantes principales calculées, est de 48.0%. Les composantes principales présentées ne possèdent pas d'interprétation claire, mais elles décrivent toutes des variations d'apparence diverses, dont notamment des imprécisions de recalage, et des effets de l'illumination. Cela mène à une modélisation des apparences contenues dans les vidéos de référence, pouvant ensuite être exploitée pour la détection de changements.*

D'autre part, comme mentionné plus haut, l'approximation commise par la version incrémentale de l'ACP peut poser problème dans le cas où les images de référence sont parcourues séquentiellement. En effet si c'était le cas, l'initialisation du mécanisme de mise à jour serait effectuée sur les N_{CP} premières images de référence, qui ont de fortes chances d'être très similaires vu la fréquence importante d'acquisition des images. Une variation importante d'apparence, intervenant plus tard dans la vidéo de référence, risquerait alors de ne pas être incluse dans le modèle final, car la variance associée, initialement faible, risquerait d'être ignorée par l'approximation effectuée à chaque étape. Pour éviter ce problème, les images de référence sont parcourues en ordre aléatoire, ce qui est possible dans le cadre de la détection de changement (contrairement au cadre de la soustraction de fond) car l'ensemble des images de référence est connu.

Détection de changements Une fois que la modélisation par ACP incrémentale a été calculée sur toutes les images de référence, elle peut être exploitée pour détecter les déviations par rapport à ce modèle. Désignons par $\text{U} = \left[\begin{array}{ccc} \mathbf{u}^1 & \dots & \mathbf{u}^{N_{\text{CP}}} \end{array} \right]$ la matrice $(DN_{\text{cell}}) \times N_{\text{CP}}$ des vecteurs propres finaux, et respectivement par $\boldsymbol{\mu}$ et $\mathbf{v}$ les vecteurs finaux de moyenne empirique et de variance empirique. Nous désignons par $\mathbf{x}$ un vecteur d'observations de test, associé à un vecteur $\boldsymbol{\delta}$ de pondération relatif aux possibles données manquantes. Désignons enfin par $\check{\mathbf{x}} = \boldsymbol{\delta} \circ (\mathbf{x} - \boldsymbol{\mu})$ le vecteur d'observations centré tolérant aux données manquantes. Le vecteur des résidus $\mathbf{r}$ est alors défini comme la différence entre $\check{\mathbf{x}}$ et sa projection sur le sous-espace engendré par les vecteurs propres $\{\mathbf{u}^k\}_{k \in [\![1, N_{\text{CP}}]\!]}$:

$$\mathbf{r}(\mathbf{x}) = \check{\mathbf{x}} - \text{U}\text{U}^T \check{\mathbf{x}} \tag{4.15}$$

Les éléments de ce vecteur des résidus peuvent alors être comparés aux éléments du vecteur de la variance empirique appris sur les observations de référence, définissant un vecteur de scores $\mathbf{s}$ comme suit :

$$\mathbf{s}(\mathbf{x}) = \frac{1}{\gamma^2} \cdot \left[\mathbf{r}(\mathbf{x}) \circ \mathbf{r}(\mathbf{x}) \right] \oslash \mathbf{v} \tag{4.16}$$

où $\oslash$ désigne la division élément par élément et où γ est une constante de mise à l'échelle liée à la probabilité a priori d'occurrence d'un changement [6]. Finalement, nous associons à chaque cellule du Quad-Tree augmenté un score de détection de changement calculé à partir des éléments correspondant à ces cellules dans le vecteur $\mathbf{s}$. Par exemple, dans le cas où les observations considérées sont en couleurs RGB, $D = 3$ et il est donc possible d'associer à chaque cellule du Quad-Tree augmenté trois éléments de $\mathbf{s}$, désignés pour une cellule donnée d'indice i par s_R^i, s_G^i et s_B^i. Ces éléments sont finalement fusionnés, pour donner un score de détection de changement scalaire, de la manière suivante :

$$\varepsilon_{\text{ipca}}(\mathbf{x}) = 1 - \frac{1}{1 + s_R^i(\mathbf{x})^2 + s_G^i(\mathbf{x})^2 + s_B^i(\mathbf{x})^2} \tag{4.17}$$

4.3.4 Détection effective des changements

Les algorithmes de modélisation d'apparence présentés dans les sections précédentes permettent d'analyser une image de test afin d'associer, à chaque observation, un score qui traduit son degré de conformité par rapport aux observations de référence. Ainsi, ces algorithmes mènent à l'estimation de ce que nous appellerons dans la suite une carte de scores, à valeurs continues comprises entre 0 et 1. Ces scores continus peuvent être vus comme des valeurs de confiance vis-à-vis de la détection de changements, indiquant le degré de certitude qu'une zone donnée correspond à un changement par rapport aux données de référence.

Toutefois, en pratique, il est souvent nécessaire d'estimer le masque de changements, c'est-à-dire de prendre une décision binaire entre changements et non-changements, afin par exemple de fournir une localisation des changements détectés. Pour cela, les approches de la littérature [34, 90, 102] utilisent généralement les cartes de scores directement, via une opération de seuillage, pour estimer le masque de changements binaire.

Cependant, dans le contexte d'un système d'assistance à l'analyse de vidéos aériennes, la conversion de ces cartes de scores en un masque de changements binaire peut bénéficier d'un certain nombre de traitements, qui permettent d'améliorer les performances finales. Ces traitements de consolidation, qui exploitent par exemple la redondance dans la vidéo de test ou encore l'interaction avec l'utilisateur, sont présentés au chapitre suivant.

6. En pratique, nous utilisons une valeur de $\gamma = 4.8$, qui donne de bons résultats. Toutefois, notons que la valeur de cette constante importe peu, car son ajustement a le même effet qu'ajuster le seuil de détection final.

Chapitre 5

Consolidation des détections

Le chapitre précédent a montré comment la redondance présente dans les observations de référence pouvait être exploitée pour créer des modèles d'apparence. Ces modèles d'apparence peuvent ensuite être utilisés pour détecter des déviations dans les observations de test, déviations qui sont alors interprétées comme des changements. Ce principe permet ainsi de détecter des régions dont l'apparence a subi un changement objectif de couleur ou de structure. Néanmoins, toutes ces régions ne correspondent pas forcément à des changements intéressants du point de vue de l'analyste image. Par exemple, les voitures se déplaçant le long d'une route très fréquentée ou les objets dont la direction de l'ombre portée a changé, correspondent à des changements objectifs d'apparence qui, en général, n'intéressent pas l'analyste image.

Par conséquent, dans le but d'affiner les résultats générés par les techniques de modélisation d'apparence, il peut être utile d'intégrer aux algorithmes de l'information a priori concernant la distinction entre changements pertinents et variations parasites. Dans le cadre de cette thèse, nous nous sommes intéressés plus particulièrement aux changements correspondant à des objets ou structures artificielles fixes (e.g. bâtiments, champs, engins en stationnement, personnes immobiles, etc), ce qui correspond à une large variété de scénarios opérationnels (voir le chapitre 1). Dans ce contexte, nous avons identifié deux axes permettant de consolider les résultats de détection : l'exploitation de la redondance présente dans la séquence des images de test pour consolider les détections associées à des objets fixes dans la scène (voir la section 5.1) et l'analyse fine de la carte de scores (voir la section 5.2).

Enfin, nous avons également exploré un troisième axe de consolidation, visant à affiner les résultats par une approche semi-automatique. Cette méthode, qui est décrite à la section 5.3, est basée sur un mécanisme d'apprentissage par retour interactif de pertinence et est plus flexible que le fait d'intégrer de l'information a priori associée à de nombreux cas différents. Elle peut également être vue comme une généralisation du principe d'exploitation d'information a priori, puisqu'elle fait intervenir l'analyste image lui-même, qui est le mieux placé pour définir la distinction idéale entre changements pertinents et variations parasites.

Sommaire

5.1 Consolidation temporelle

Nous avons vu au chapitre 2 que l'utilisation de modèles d'apparence permettait d'exploiter la redondance contenue dans les vidéos de référence, ce qui permet un gain de performance substantiel par rapport à l'utilisation d'une image de référence unique. Le même raisonnement peut être appliqué à la redondance contenue dans la vidéo de test, dont l'exploitation doit également permettre un gain de performance significatif. Toutefois, le fait que nous souhaitions pouvoir effectuer une détection en ligne des changements dans la vidéo de test constitue un obstacle non-négligeable à l'exploitation de la redondance dans la vidéo de test. En effet, non seulement est-il souhaitable d'effectuer un traitement rapide des images de tests, mais de plus l'ensemble de ces images n'est pas disponible lors de la détection de changements.

Pour remédier à cela, nous avons développé deux algorithmes [16] permettant d'exploiter cette redondance tout en restant compatibles avec la contrainte de traitement en ligne de la vidéo de test. Ces algorithmes prennent l'hypothèse que les changements que nous souhaitons détecter sont fixes (e.g. bâtiments, champs), ce qui est généralement le cas dans le domaine de la détection de changements en imagerie aérienne. En effet, dans le cas contraire, on parle plutôt de suivi d'objets mobiles, et une technique permettant d'exploiter la redondance temporelle dans ce contexte a été proposée par Yin et Collins [120].

En exploitant l'information a priori selon laquelle les changements pertinents sont fixes, nos deux algorithmes permettent d'améliorer la performance de détection en imposant une certaine cohérence entre les cartes de scores de détection de changements obtenues sur les images successives. Le premier, qui est décrit à la section 5.1.1, est basé sur un lissage temporel des scores de détection de changements. Le second, qui est décrit à la section 5.1.2, est lui basé sur une optimisation de la cohérence spatio-temporelle formulée dans le cadre de la propagation de croyance [10, 117].

5.1.1 Lissage temporel du score de détection

Pour commencer, nous avons développé [16] la méthode de consolidation par lissage temporel afin de permettre l'exploitation de manière simple et rapide de la redondance temporelle contenue dans la vidéo de test. Cette méthode consiste à calculer, de manière incrémentale et indépendante en chaque pixel, le score moyen de détection de changements sur les images de test successives, puis à décider pour chaque pixel s'il correspond à un changement ou non. Cette décision étant prise grâce au score moyen de détection de changements, la classification entre changements et non-changements est bien effectuée sur la base de l'ensemble des images de test observées, ce qui rend les résultats plus fiables.

La mise en œuvre de cette méthode pose cependant quelques problèmes techniques. En effet, comme la plate-forme d'observation peut se déplacer pendant l'acquisition de la vidéo, un même point physique peut être situé à des emplacements différents dans des images différentes. Il convient donc de mettre en correspondance les images successives afin de calculer correctement les scores moyens de détection de changements. Afin d'éviter de nombreux ré-échantillonnages successifs, il est préférable de ne pas calculer les scores moyens en coordonnées images, mais plutôt de les calculer en coordonnées spatiales dans un référentiel absolu. Pour cela, nous avons choisi de calculer ces scores dans le référentiel absolu lié au plan dominant du sol. Une alternative pourrait consister à calculer ces scores directement dans les cellules du modèle tri-dimensionnel afin d'améliorer la précision de la mise en correspondance. Cependant, les cellules correspondant généralement à plusieurs pixels dans l'image, cela entraînerait une réduction de la résolution de détection. Cette alternative n'a donc pas été retenue. Par ailleurs, le fait d'utiliser un référentiel absolu pour le calcul des scores moyens requiert d'imposer une limite a priori sur l'étendue géographique de la scène observée. Cette contrainte n'est toutefois pas limitante. En effet, l'objectif est d'effectuer la détection de changements par

rapport aux observations de référence, dont l'étendue géographique est connue lorsque le traitement de la vidéo de test commence. Or, il est inutile de prendre en compte les observations de test pour lesquelles aucune observation de référence n'est disponible. Par conséquent, il est suffisant que la limite a priori imposée sur l'étendue de la scène de test corresponde à l'étendue géographique de la scène observée dans les vidéos de référence. La suite de cette section décrit les détails techniques relatifs à la mise en œuvre de cette méthode de lissage temporel.

Soit $\{S_k\}_{k\in[\![0,t]\!]}$ la suite des cartes de scores de détection de changements obtenues jusqu'à l'étape courante t, telles qu'obtenues par l'une des techniques décrites à la section 4.3, et soit $\mathrm{H}_{k\leftarrow\mathrm{sol}}$ l'homographie transformant les coordonnées exprimées dans le plan dominant du sol vers les coordonnées de l'image à l'étape k. Soit $\mathbf{m}_{\mathrm{sol}}$ un point donné appartenant au plan dominant du sol dans la scène observée. Soit enfin $\mathrm{vis}_k(\mathbf{m}_{\mathrm{sol}})$ la fonction indiquant si le point $\mathbf{m}_{\mathrm{sol}}$ est visible dans l'image à l'étape k, égale à 1 s'il est visible et à 0 sinon. Le score lissé temporellement $\widehat{\varepsilon}_t(\mathbf{m}_{\mathrm{sol}})$ au point $\mathbf{m}_{\mathrm{sol}}$ est alors estimé comme suit :

$$\begin{aligned}\widehat{\varepsilon}_t(\mathbf{m}_{\mathrm{sol}}) &= \frac{1}{\sum_{k\in[\![0,t]\!]}\mathrm{vis}_k(\mathbf{m}_{\mathrm{sol}})}\cdot\sum_{k\in[\![0,t]\!]}\mathrm{vis}_k(\mathbf{m}_{\mathrm{sol}})\cdot S_k(\mathrm{H}_{k\leftarrow\mathrm{sol}}\mathbf{m}_{\mathrm{sol}})\\ &= \frac{\sum_{k\in[\![0,t-1]\!]}\mathrm{vis}_k(\mathbf{m}_{\mathrm{sol}})}{\sum_{k\in[\![0,t]\!]}\mathrm{vis}_k(\mathbf{m}_{\mathrm{sol}})}\cdot\widehat{\varepsilon}_{t-1}(\mathbf{m}_{\mathrm{sol}})+\frac{\mathrm{vis}_t(\mathbf{m}_{\mathrm{sol}})}{\sum_{k\in[\![0,t]\!]}\mathrm{vis}_k(\mathbf{m}_{\mathrm{sol}})}\cdot S_t(\mathrm{H}_{t\leftarrow\mathrm{sol}}\mathbf{m}_{\mathrm{sol}})\end{aligned}\tag{5.1}$$

L'équation précédente montre que l'estimation du score lissé temporellement peut être effectuée de manière incrémentale, ce qui permet une consolidation rapide et exploitant l'ensemble des observations de test passées. La complexité de cet algorithme, qui est exécuté pour chaque nouvelle image de test, est en $O(w\cdot h)$, où w et h représentent la dimension des images de test considérées.

Par la suite, pour obtenir le masque de changements binaire, ces scores lissés temporellement peuvent être comparés à un seuil, ou analysés plus finement comme expliqué à la section 5.2. Malgré le fait que la moyenne temporelle soit peu fiable pour les premières images de la vidéo de test, nous affichons les masques de changements calculés à l'aide de cette méthode dès la première image, afin que les résultats soient présentés à l'analyse image dès qu'ils sont disponibles.

D'autre part, cette méthode, bien qu'étant rapide et simple à mettre en œuvre, présente l'inconvénient de ne pas exploiter la vidéo de test au maximum de son potentiel. En effet, la redondance dans une vidéo n'est pas uniquement temporelle, mais également spatiale. Par ailleurs, l'intégration d'un modèle a priori décrivant le comportement des changements visés peut permettre d'améliorer les performances. Par conséquent, nous avons conçu une seconde technique, présentée à la section suivante, afin d'effectuer une consolidation plus précise des résultats.

5.1.2 Optimisation de la cohérence spatio-temporelle

Pour consolider les résultats de détection de changements de manière plus précise, nous avons proposé [16] une seconde méthode de consolidation temporelle basée sur une optimisation spatio-temporelle de la classification entre changements et non-changements, formulée dans le cadre de la propagation de croyance [10, 117]. Cette approche permet non seulement d'imposer une cohérence temporelle dans les résultats successifs de détection de changements, comme la méthode précédente, mais également d'imposer une cohérence spatiale dans les résultats.

Le cadre de la propagation de croyance permet de déterminer un ensemble structuré d'états cachés, c'est-à-dire non observables, à partir de leurs manifestations observables [118]. La structure dans les états cachés est modélisée grâce à un graphe non-orienté, dont chaque nœud est associé à un état donné et à sa manifestation. Ce cadre correspond très bien au problème

de la détection de changements, dans lequel nous cherchons à déterminer l'état non observable de changement ou de non-changement des pixels dans une vidéo, à partir de leurs manifestations représentées par les scores de détection de changements. Pour déterminer l'ensemble des états correspondant de manière optimale aux manifestations observées, l'algorithme de propagation de croyance effectue un échange itératif de messages entre nœuds voisins dans le graphe. D'autres cadres existent pour résoudre le même problème d'estimation, par exemple celui des *graph cuts* [20]. Cependant, nous avons préféré utiliser l'algorithme de propagation de croyance parce qu'il ne nécessite pas la construction explicite du graphe sous-jacent, ce qui permet de rendre sa mise en œuvre moins lourde.

Cette méthode reste néanmoins plus coûteuse que la méthode de lissage présentée à la section précédente. Afin de réduire les temps d'exécution, nous avons choisi de réduire les dimensions des images d'un facteur 4 et d'effectuer la consolidation directement dans les coordonnées images. D'autre part, nous utilisons une fenêtre temporelle glissante plutôt que d'exploiter l'ensemble des observation de test passées. La suite de cette section décrit les détails techniques relatifs à cette méthode d'optimisation spatio-temporelle.

Soit $\Omega_t = [\![t-T+1, t]\!]$ la fenêtre temporelle glissante de longueur T, utilisée pour la consolidation temporelle des résultats (en pratique, nous utilisons $T = 8$). Désignons par $\{I_k^{\text{test}}\}_{k\in\Omega_t}$ les T plus récentes images de test. Désignons respectivement par $\{S_k\}_{k\in\Omega_t}$ et $\{I_k^{\text{ref}}\}_{k\in\Omega_t}$ les T plus récentes cartes de scores de détection de changements et les T images obtenues par rendu 3D du modèle de référence selon les points de vue des images $\{I_k^{\text{test}}\}_{k\in\Omega_t}$. Dans la suite de cette section, toutes les images $\{I_k^{\text{test}}\}_{k\in\Omega_t}$, $\{I_k^{\text{ref}}\}_{k\in\Omega_t}$ ainsi que les cartes de scores $\{S_k\}_{k\in\Omega_t}$ sont supposées recalées par rapport à l'image de test I_t^{test} la plus récente. De plus, nous supposons que toutes les images et les cartes de scores correspondantes ont été sous-échantillonnées par un facteur 4. Nous désignons alors par $w \times h$ leurs dimensions communes, et nous définissons les intervalles suivants : $\mathcal{U} = [\![0, w-1]\!]$ et $\mathcal{V} = [\![0, h-1]\!]$. Pour simplifier les notations, un indice spatio-temporel donné sera désigné dans la suite par $l = (u, v, k) \in \mathcal{L}_t = \mathcal{U} \times \mathcal{V} \times \Omega_t$.

Le graphe utilisé pour modéliser la structure dans les états cachés est un champ de Markov aléatoire spatio-temporel, où chaque nœud correspond à un pixel dans les images sous-échantillonnées. Désignons ce graphe par $\mathcal{G}_t = \{\mathcal{N}_t, \mathcal{A}_t\}$, où $\mathcal{N}_t = \{n_l\}_{l\in\mathcal{L}_t}$ représente l'ensemble des nœuds et $\mathcal{A}_t = \{a_{l\leftrightarrow l'} = (n_l, n_{l'}) \in {\mathcal{N}_t}^2, \|l - l'\|_1 = 1\}$ l'ensemble des arêtes, $\|\cdot\|_1$ représentant la norme L_1. Cette définition caractérise le système de voisinage non-orienté entre nœuds, chacun ayant donc au plus six voisins, dont quatre spatialement et deux temporellement. La figure 5.1 illustre l'allure de ce graphe et illustre la correspondance avec la séquence d'images issues de la vidéo de test. Par ailleurs, à chaque nœud n_l du graphe est associé un état caché binaire $y_l \in \{0, 1\}$ et une manifestation $x_l \in \mathbb{R}$, définie pour l donné par $x_l = x_{u,v,k} = S_k(u, v)$.

Dans ce contexte, l'objectif de l'algorithme de propagation de croyance généralisé est d'estimer l'ensemble optimal d'états cachés binaires $\{y_l\}_{l\in\mathcal{L}_t}$ à partir des manifestations $\{x_l\}_{l\in\mathcal{L}_t}$ issues des carte de scores $\{S_k\}_{k\in\Omega_t}$. Pour cela, la loi de probabilité jointe, liant les états et les manifestations, est maximisée. Cette loi de probabilité jointe peut être mise sous la forme d'un modèle d'énergie potentielle de Potts (ou de Ising, dans le cas d'états binaires), de la façon suivante :

$$\begin{aligned} p\big(\{x_l, y_l\}_{l\in\mathcal{L}_t}\big) &= \frac{1}{Z} \prod_{a_{l\leftrightarrow l'}\in\mathcal{A}_t} \Psi_{a_{l\leftrightarrow l'}}(y_l, y_{l'}) \prod_{n_l\in\mathcal{N}_t} \Phi_{n_l}(y_l, x_l) \\ &= \frac{1}{Z} \exp\Big[-E\big(\{x_l, y_l\}_{l\in\mathcal{L}_t}\big) \Big] \end{aligned} \tag{5.2}$$

$$\text{avec} \quad E\big(\{x_l, y_l\}_{l\in\mathcal{L}_t}\big) = -\sum_{a_{l\leftrightarrow l'}\in\mathcal{A}_t} \ln \Psi_{a_{l\leftrightarrow l'}}(y_l, y_{l'}) - \sum_{n_l\in\mathcal{N}_t} \ln \Phi_{n_l}(y_l, x_l) \tag{5.3}$$

Le terme $\frac{1}{Z}$ de l'équation 5.2 représente un terme de normalisation garantissant que le résultat est bien une loi de probabilité. Cette loi de probabilité jointe est définie par la fonction

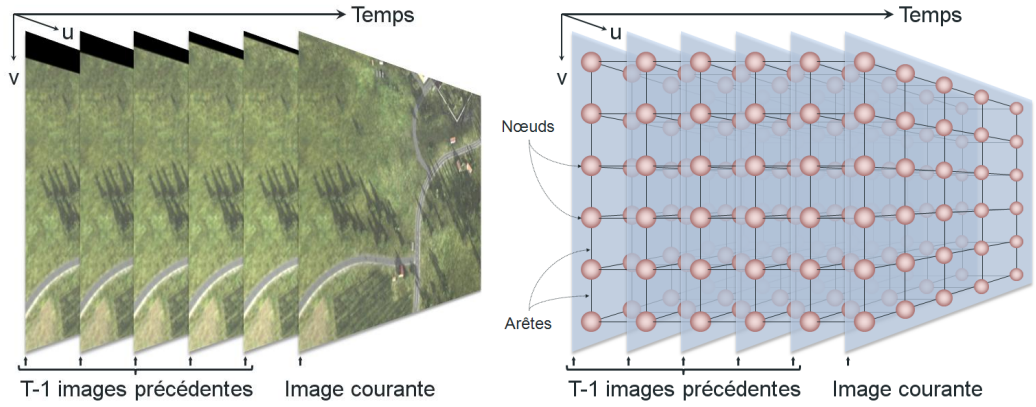


FIGURE 5.1 – *La figure de gauche (a) présente une séquence de six images issues de la vidéo de test toutes recalées par rapport à l'image courante. Les bandes noires en haut des images correspondent aux zones non observées mises en évidence par le recalage. La figure de droite (b) illustre l'allure du graphe associé à cette séquence, utilisé pour l'algorithme d'optimisation de la cohérence par propagation de croyance. Ce graphe est un champ de Markov spatio-temporel, où chaque nœud possède au plus six voisins.*

de compatibilité entre états voisins, représentée par $\Psi_{a_{l\leftrightarrow l'}}(y_l, y_{l'})$ pour $a_{l\leftrightarrow l'} \in \mathcal{A}_t$, et par la fonction de manifestation, représentée par $\Phi_{n_l}(y_l, x_l)$ pour $n_l \in \mathcal{N}_t$. La première caractérise les probabilités d'occurrence de chaque couple d'états possibles pour deux nœuds voisins, tandis que la seconde caractérise la probabilité d'occurrence conjointe pour une manifestation et un état donnés.

Dans le cas particulier de la consolidation temporelle pour la détection de changement dans des vidéos, nous avons proposé [16] d'utiliser l'expression suivante pour la fonction de manifestation $\Phi_{n_l}(y_l, x_l)$:

$$\Phi_{n_l}(y_l, x_l) = \begin{cases} f_{n_l}(x_l) & \text{si } y_l = 0 \text{ (absence de changement)} \\ 1 - f_{n_l}(x_l) & \text{si } y_l = 1 \text{ (présence de changement)} \end{cases} \tag{5.4}$$

$$\text{avec} \quad f_{n_l}(x_l) = \Big[c_0 \exp\big[-c_1 \Delta(n_l) \big] - c_0 + 1 \Big] \exp\left(-\frac{x_l}{\tau} \right) \tag{5.5}$$

$$\Delta(n_l) = \Delta(n_{u,v,k}) = \left| I_k^{\text{test}}(u,v) - I_k^{\text{ref}}(u,v) \right|$$

Dans l'équation ci-dessus, τ représente le seuil de détection, utilisé pour ajuster les taux de vrais et faux positifs, notamment pour la génération des courbes ROC. L'expression de $f_{n_l}(x_l)$ permet d'encourager le statut de changement ou non-changement d'un état y_l, en fonction des valeurs du score $x_l = x_{u,v,k} = S_k(u,v)$ de détection de changements associé et de la différence $\Delta(n_l)$ entre les images de test et de référence. Le facteur de droite, dans la définition de f_{n_l}, est grand lorsque x_l est petit devant le seuil de détection τ. Ce comportement permet d'encourager l'état de non-changement (respectivement, changement) pour y_l lorsque x_l est faible (respectivement élevé). Ce facteur est pondéré par un second facteur, qui est défini en fonction des constantes c_0 et c_1 et de la différence $\Delta(n_l)$ entre les images de test et de référence. Ce second facteur, qui est grand lorsque $\Delta(n_l)$ est faible et inversement, permet d'ajuster l'amplitude du premier en fonction de la différence $\Delta(n_l)$ entre les images. Plus précisément, lorsque $\Delta(n_l)$ est faible, ce second facteur est grand, ce qui permet d'augmenter les valeurs possibles de f_{n_l} et donc d'encourager l'état de non-changement. Une justification de cette heuristique est qu'une faible différence entre les images peut laisser penser qu'aucun changement n'est présent, par conséquent nous encourageons l'état de non-changement. Dans le cas contraire, si la différence d'image $\Delta(n_l)$ est importante, alors nous encourageons légèrement l'état de changement en diminuant les valeurs possibles de f_{n_l}. Toutefois, une importante différence entre les images peut également être due à des effets parasites divers, tels que l'illumination, et il est donc préférable

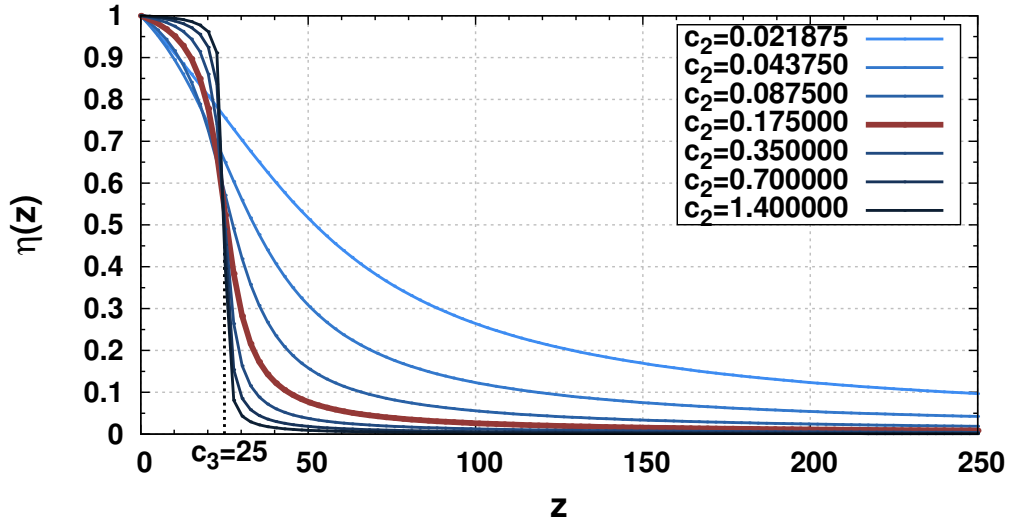


FIGURE 5.2 – *Cette figure illustre l'allure des fonctions de la famille $\eta(\cdot)$, définie à l'équation 5.7, pour $c_3 = 25$ et pour quelques valeurs de c_2. En pratique, nous utilisons une valeur de $c_2 = 0.175$, ce qui correspond à la courbe tracée en rouge.*

de ne pas trop diminuer les valeurs de f_{n_l}. En pratique, nous utilisons les valeurs $c_0 = \frac{1}{3}$ et $c_1 = \frac{\log(2)}{30}$, qui donnent de bons résultats.

Par ailleurs, nous avons également proposé [16] d'utiliser l'expression suivante pour la fonction de compatibilité entre états voisins :

$$\Psi_{a_{l\leftrightarrow l'}}(y_l, y_{l'}) = \begin{cases} 0.95\eta\left[|\Delta(n_l) - \Delta(n_{l'})|\right] + 0.5\left[1 - \eta\left[|\Delta(n_l) - \Delta(n_{l'})|\right]\right] & \text{si } y_l = y_{l'} \\ 0.05\eta\left[|\Delta(n_l) - \Delta(n_{l'})|\right] + 0.5\left[1 - \eta\left[|\Delta(n_l) - \Delta(n_{l'})|\right]\right] & \text{si } y_l \neq y_{l'} \end{cases} \tag{5.6}$$

$$\text{avec} \quad \eta(z) = \frac{\frac{\pi}{2} - \text{atan}\left[c_2(z - c_3)\right]}{\frac{\pi}{2} - \text{atan}(-c_2 c_3)} \tag{5.7}$$

Dans cette définition, la famille de fonctions $\eta(\cdot)$ est un ensemble de fonctions scalaires à base radiale dont la bande passante est contrôlée par le paramètre c_3 et dont la vitesse de décroissance est contrôlée par le paramètre c_2. La figure 5.2 présente l'allure de quelques fonctions de cette famille pour quelques valeurs du paramètre c_2. La présence de la fonction $\eta(\cdot)$ dans la définition de la fonction de compatibilité entre états voisins permet d'influencer la compatibilité d'états voisins en fonction du gradient dans l'image des différences $|\Delta(n_l) - \Delta(n_{l'})|$. Plus précisément, la valeur de $\eta\left[|\Delta(n_l) - \Delta(n_{l'})|\right]$ est grande lorsque $|\Delta(n_l) - \Delta(n_{l'})|$ est faible. Par conséquent la définition proposée ci-dessus permet d'encourager des états voisins identiques lorsque le gradient dans l'image des différences est faible (pondération de $\eta(\cdot)$ par 0.95 si $y_l = y_{l'}$, et par 0.05 sinon). Une justification de cette heuristique est que lorsque le gradient dans l'image des différences est faible, alors les nœuds n_l et $n_{l'}$ correspondent vraisemblablement à un même objet dans la scène et par conséquent les états devraient être cohérents. Au contraire, lorsque le gradient dans l'image des différences est important, nous ne favorisons aucun des deux cas par rapport à l'autre (pondération de $1 - \eta(\cdot)$ par 0.5 dans les deux cas), car alors les nœuds n_l et $n_{l'}$ correspondent vraisemblablement à deux objets différents dans la scène. Les états correspondants sont alors décorrélés, pouvant aussi bien être identiques que différents, la décision étant prise sur la base de la fonction de manifestation. En pratique, nous obtenons des résultats satisfaisants avec la fonction $\eta(\cdot)$ correspondant à $c_2 = 0.175$ et $c_3 = 25$.

À l'aide des définitions fournies par les équations 5.4 et 5.6, l'algorithme itératif de propagation de croyance permet d'estimer, en chaque nœud du graphe, la probabilité qu'il corresponde à un changement. Pour cela, à chaque itération, l'algorithme propage, de chaque nœud vers

ses voisins, des messages portant une information de vraisemblance relative aux états possibles du voisin visé. La convergence de cet algorithme vers une solution optimale n'est garantie que lorsque le graphe utilisé ne contient pas de cycles, ce qui n'est pas le cas avec un champ de Markov. Malgré cela, de nombreux auteurs [117] ont constaté qu'en présence de cycles, l'algorithme de propagation de croyance, qui est alors qualifié de généralisé, converge vers une solution intéressante en pratique.

L'utilisation de cet algorithme de consolidation temporelle permet une amélioration des performances de détection de changements par rapport à la méthode présentée à la section précédente (voir chapitre 6). Cependant, cette meilleure précision s'accompagne également de temps de traitements plus long. La complexité de cet algorithme, qui est exécuté pour chaque nouvelle image de test, est en $O(N_{\text{iter}} \cdot N_{\text{voisins}} \cdot w \cdot h \cdot T)$, où N_{iter} est le nombre d'itérations de l'algorithme et N_{voisins} est le nombre de voisins par nœud. En pratique, nous avons $N_{\text{iter}} = 4$ et $N_{\text{voisins}} = 6$. Par conséquent, les dimensions des images considérées, w et h, étant réduites d'un facteur 4 par rapport à celles considérées par la méthode présentée à la section précédente, nous obtenons un algorithme environ 12 fois plus lent que le précédent (voir les mesures précises de temps de calcul au chapitre 6). Bien que notre effort n'ait pas porté sur ce point dans le cadre de cette thèse, notons toutefois qu'en pratique, l'algorithme de propagation de croyance peut être accéléré de manière significative grâce à une implémentation efficace [42] ou grâce à une parallélisation sur GPU [23].

5.1.3 Lissage temporel hybride

Les méthodes présentées plus haut, basées sur un lissage temporel (section 5.1.1) et une optimisation spatio-temporelle de la cohérence (section 5.1.2), sont appliquées dans des conditions différentes. En effet, la première utilise la totalité des images observées jusqu'à l'étape courante, tandis que la seconde n'utilise que les $T = 8$ plus récentes. Les calculs de la première sont également effectués dans le référentiel du sol, tandis que ceux de la seconde sont effectués en coordonnées image et impliquent un recalage et un sous-échantillonnage des images considérées.

Par conséquent, afin de permettre une juste comparaison entre les deux méthodes, dont le détail sera présenté au chapitre 6, nous avons également développé une méthode de lissage temporel que nous qualifierons d'hybride, la méthode présentée à la section 5.1.1 étant alors qualifiée de complète. Cette méthode de lissage temporel hybride calcule les scores moyens de détection de changements en coordonnées image et sur les $T = 8$ images de test les plus récentes. Cela permettra notamment de montrer que les différences de performances, mises en évidence au chapitre 6, sont bien dues aux algorithmes eux-mêmes et pas simplement aux différences dans les conditions d'application.

5.2 Binarisation des scores de détection

Les techniques de détection de changements présentées au chapitre 4, basées sur la modélisation des apparences, permettent l'estimation d'une carte de scores continus de détection de changements. Les cartes de scores successives peuvent éventuellement subir une consolidation temporelle pour améliorer leur précision. Cependant, les scores continus finaux doivent dans tous les cas être convertis en classes binaires, afin d'estimer le masque de changements.

Cette étape, que nous désignerons par binarisation, est généralement effectuée grâce à un seuillage des scores de détection de changements. L'intérêt de l'opération de seuillage est qu'elle est très simple et donc très rapide à effectuer. Cependant, sa simplicité fait qu'une partie de l'information contenue dans les cartes de scores est perdue lors de la binarisation. La figure 5.3 illustre cette perte d'information à l'aide de deux exemples.

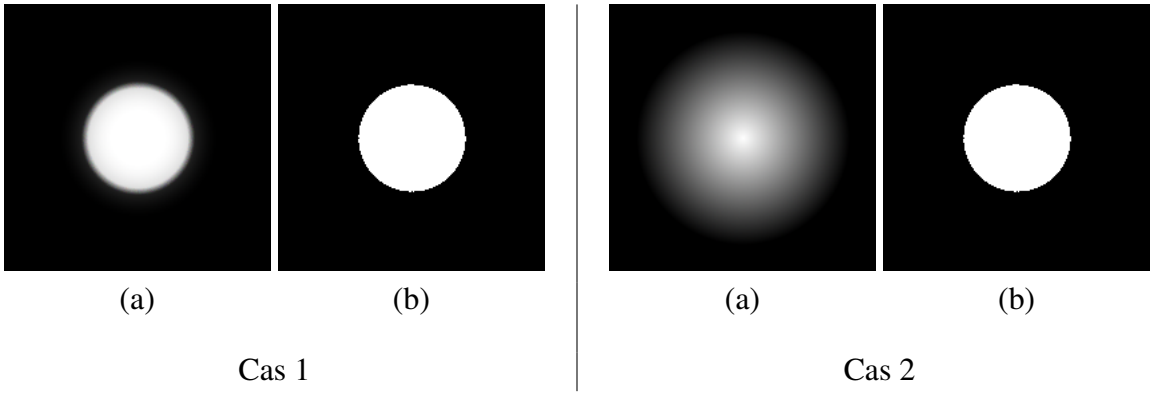


FIGURE 5.3 – *Cette figure illustre la perte d'information intervenant lors de la binarisation d'une carte de score de détection de changements par seuillage. Deux cartes de scores synthétiques clairement différentiables sont présentées en (1a) et (2a). Cependant, après binarisation par un seuillage à l'aide d'un seuil identique, les deux masques de changements résultants (1b) et (2b) sont identiques.*

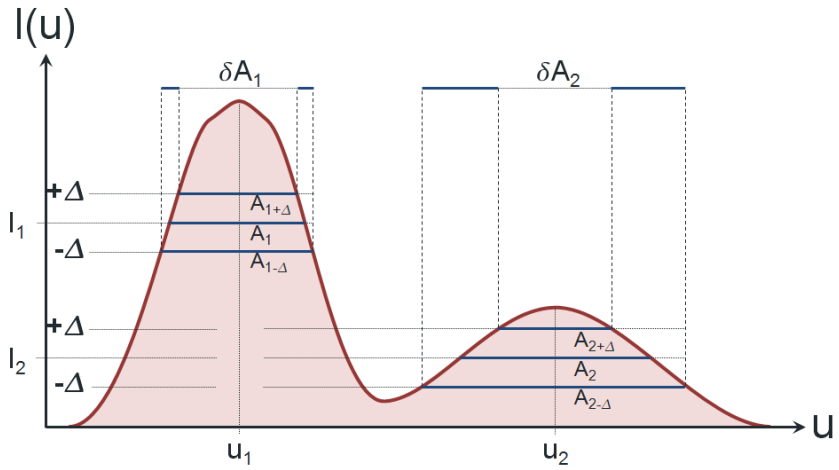


FIGURE 5.4 – *Cette figure illustre dans le cas uni-dimensionnel la notion de région extrême maximale à stabilité maximale, plus communément désignée par MSER dans la littérature [75].*

En revanche, une analyse plus fine de la carte de scores lors de la binarisation peut donner de meilleurs résultats. Ainsi, nous avons développé une technique permettant d'effectuer la binarisation à l'aide de l'algorithme d'extraction [37] de régions extrêmes maximales à stabilité maximale (MSER, pour *Maximally Stable Extremal Region* dans la littérature), qui permet un meilleur filtrage des faux positifs dus à l'illumination.

En effet, dans le contexte de la détection de changements dans des vidéos aériennes, les changements d'intérêt correspondent le plus souvent à des structures artificielles dont les frontières sont nettes et bien définies, tels que des bâtiments, des engins en stationnement, des champs, des personnes, etc. Au contraire, les effets dus à l'illumination se manifestent le plus souvent dans la carte de score par des zones aux frontières plus floues et étendues [1], en particulier en utilisant une technique d'atténuation de l'illumination, comme proposé à la section 3.2.

En pratique, cette différence de comportement peut être capturée par la notion de MSER. Cette notion, introduite par Matas et al. [75], décrit les régions correspondant à des maxima

1. Cela peut dépendre des conditions d'acquisition et en particulier des conditions météorologiques. Par exemple en cas de temps clair et très ensoleillé, les objets dans la scène peuvent générer des ombres portées dures dont les frontières sont très nettes. Le filtrage des faux positifs associés, par atténuation ou par analyse de la carte de scores, peut alors être difficile. Cependant, ces faux positifs peuvent être facilement éliminés à l'aide d'information a priori spécifique. Voir également la section 5.3 pour une solution possible.

locaux et dont l'aire varie peu lorsque le seuil de binarisation varie (voir l'illustration de la figure 5.4). Plus précisément, définissons d'abord la bordure externe ∂Q d'une région connexe Q comme étant l'ensemble des pixels $q_b \notin Q$ adjacents à Q. Une région donnée Q est alors qualifiée d'extrême maximale dans une image I lorsque :

$$\forall q \in Q \text{ et } \forall q_b \in \partial Q,\ I(q) > I(q_b) \tag{5.8}$$

Cette définition permet de définir des suites imbriquées de régions extrêmes maximales regroupées autour des maxima locaux dans l'image I, comme par exemple les régions imbriquées autour de u_1 et celles imbriquées autour de u_2 dans l'illustration de la figure 5.4. Soit $\{Q_\tau\}_{\tau \in \mathbb{R}^+}$ une telle suite imbriquée de régions extrêmes maximales, où $Q_\tau = \{q,\ I(q) > \tau\}$. La stabilité d'une région Q_{τ_0} donnée de cette suite est alors définie comme étant inversement proportionnelle au taux de variation local de l'aire de cette région :

$$\text{stabilité}_\Delta(Q_{\tau_0}) \propto \left[\frac{\text{aire}(Q_{\tau_0-\Delta}) - \text{aire}(Q_{\tau_0+\Delta})}{\text{aire}(Q_{\tau_0})} \right]^{-1} \tag{5.9}$$

où Δ est un paramètre caractérisant la localité du taux de variation. Dans le cas de la figure 5.4, la stabilité de la région Q_1 définie autour de u_1 est proportionnelle à $[\delta A_1/A_1]^{-1}$ et celle de la région Q_2 définie autour de u_2 est proportionnelle à $[\delta A_2/A_2]^{-1}$. La pente de la fonction I étant plus forte autour de u_1 que celle autour de u_2, pour des largeurs de pics équivalentes, la stabilité de Q_1 est donc supérieure à celle de Q_2.

Cette notion de région extrême maximale à stabilité maximale peut être appliquée dans le cas de la consolidation des résultats en détection de changements. En effet, les régions possédant des frontières nettes dans la carte de scores correspondent à des régions binarisées dont l'aire varie peu voire pas du tout lorsque le seuil de binarisation change (voir la figure 5.3, cas 1). Ces régions ont donc une forte stabilité et par conséquent elles peuvent être détectées par l'algorithme d'extraction de MSER [37]. Au contraire, les régions possédant des frontières floues dans la carte de scores correspondent à des régions binarisées dont l'aire varie beaucoup lorsque le seuil de binarisation change (voir la figure 5.3, cas 2), et sont donc ignorées par l'algorithme d'extraction de MSER. Les performances obtenues à l'aide de la méthode classique de seuillage et avec la méthode d'extraction de MSER sont présentées et discutées au chapitre 6.

5.3 Retour interactif de pertinence

À ce stade, l'approche de détection de changements composée des méthodes et algorithmes présentés jusqu'ici donne des résultats satisfaisants mais encore parsemés de faux positifs. Il est notamment possible de citer le cas des ombres dures, totalement noires et aux frontières très nettes, qui sont mal atténuées par les techniques de pré-traitements (section 3.2) et qui échappent également à l'analyse fine de la carte de scores présentée à la section précédente. Nous pourrions envisager de filtrer les derniers faux positifs, qui dépendent du contexte d'application souhaité, en intégrant plus d'information a priori permettant de désambiguïser ces erreurs. Cependant, cela pourrait non seulement alourdir les traitements, car beaucoup de cas différents devraient être gérés, mais cela rendrait également les traitements de plus en plus spécialisés et donc de moins en moins flexibles. Or, le problème des faux positifs résiduels peut être vu d'une autre manière en remarquant qu'en réalité, l'analyste image est le mieux placé pour définir ce qui distingue un changement pertinent d'une variation parasite.

Par conséquent, l'approche idéale pour éliminer les faux-positifs résiduels consiste à apprendre la distinction entre changement pertinent et variation parasite directement par interaction avec l'analyste image. Cela peut être effectué de manière intuitive et transparente pour l'utilisateur grâce au mécanisme d'apprentissage par retour de pertinence (*relevance feedback* dans

la littérature). L'apprentissage par retour de pertinence est un mécanisme semi-automatique mettant ponctuellement l'utilisateur à contribution pour améliorer progressivement les performances de détection. Un tel mécanisme semi-automatique est tout à fait adapté à notre cas d'application, puisque dans le cadre d'un système d'assistance à l'analyse, l'utilisateur participe déjà à l'analyse des données. Il est en revanche important de faire en sorte que la mise à contribution de l'utilisateur ne rende pas le système lourd et inutilisable en pratique.

La suite de cette section est organisée de la façon suivante. La section 5.3.1 présente le principe de la méthode d'apprentissage par retour de pertinence que nous avons développé dans le cadre de la détection de changements, et ses spécificités par rapport aux méthodes de retour de pertinence utilisées généralement dans le domaine de la fouille de données. Par la suite, la section 5.3.2 introduit le descripteur proposé pour caractériser une région détectée comme changement possible, et la section 5.3.3 détaille la méthode de classification utilisée.

5.3.1 Principe de fonctionnement

Le mécanisme d'apprentissage par retour interactif de pertinence a été introduit initialement pour le domaine de la fouille de données [97] et, par la suite, s'est principalement développé dans ce cadre [122]. Le principe général de ce mécanisme interactif consiste dans un premier temps à présenter à l'utilisateur les résultats issus d'un processus de génération initial. L'utilisateur est ensuite mis à contribution afin de donner un jugement de pertinence relatif aux résultats générés, information qui est exploitée pour affiner le processus de génération des résultats. Enfin, le nouveau processus est appliqué pour générer de nouveaux résultats, et le mécanisme est recommencé jusqu'à ce que l'utilisateur soit satisfait des résultats.

Comme mentionné plus haut, dans notre cas, la motivation principale justifiant l'introduction d'un mécanisme de retour de pertinence dans le contexte de la détection de changements est la réduction des faux positifs résiduels. Nous pouvons donc adapter le mécanisme d'apprentissage interactif en considérant l'ensemble des algorithmes présentés jusqu'ici comme processus de génération initial. Les régions détectées par ce processus de génération sont ensuite examinées par l'utilisateur, qui fournit un retour de pertinence sous la forme d'annotations binaires, indiquant si une région détectée donnée correspond à un changement pertinent ou à une variation parasite. Ces annotations sont ensuite utilisées pour entraîner un classificateur automatique permettant d'apprendre la distinction entre vrais positifs et faux positifs, telle que définie par l'utilisateur. Ainsi, l'objectif de cette approche est de réduire le nombre de faux positifs tout en évitant le plus possible d'augmenter le nombre de faux négatifs. En d'autres termes, cette approche doit permettre d'éliminer des résultats les régions détectées ne correspondant pas à des changements pertinents, tout en conservant un maximum des changements pertinents correctement détectés. Notons que cela signifie que cette approche ne permettra pas de réduire le nombre de faux négatifs, ce qui nécessiterait d'ajuster les algorithmes de détection et d'effectuer une nouvelle détection à chaque étape du mécanisme interactif. Il a été choisi de ne pas mettre en œuvre cette option, afin de ne pas alourdir l'algorithme et de permettre un retour de pertinence transparent pour l'analyste image.

Par ailleurs, l'utilisation d'un mécanisme de retour de pertinence dans le cadre de la détection de changements en ligne dans une vidéo de test présente des caractéristiques spécifiques que nous ne retrouvons pas dans le cadre de la fouille de données. En particulier, ce mécanisme de retour de pertinence doit être compatible avec le caractère en ligne de la détection de changements effectuée. Il est donc nécessaire de traiter le flux des images de test dans leur ordre de réception et d'afficher les régions détectées (i.e. les changements potentiels) sur chaque image. Cela a notamment pour conséquence de limiter le gain de performance apporté par les techniques d'apprentissage actif. En effet, ces techniques visent à sélectionner et trier de manière optimale les régions présentées à l'utilisateur pour annotation, afin de maximiser le gain d'information retiré. Or, dans notre cas, l'ordre de présentation est imposé et les régions détectées,

Entrées : Régions $\{Q_j^t\}_{j\in[\![0,N_t-1]\!]}$ détectées comme changements potentiels sur l'image courante I_t
Sorties : Classification des régions détectées en deux classes, classificateur $\mathcal{C}_t$ entraîné selon les annotations fournies par l'utilisateur sur l'image I_t

1: **Pour** $j \in [\![0, N_t - 1]\!]$ **Faire**
2: Extraire le descripteur $\mathbf{d}_j^t$ associé à la région Q_j^t
3: **Si** $t > 0$ **Alors**
4: Prédire la classe $\hat{y}_j^t \in \{0,1\}$ de la région Q_j^t grâce au descripteur $\mathbf{d}_j^t$ et au classificateur $\mathcal{C}_{t-1}$
5: **Sinon**
6: Affecter Q_j^t à la classe des changements ($\hat{y}_j^t \leftarrow 1$)
7: **Fin Si**
8: **Fin Pour**
9: Afficher les détections et leurs classes dans l'image I_t
10: Pour $J \subset [\![0, N_t - 1]\!]$ choisi par l'utilisateur, recueillir les classes réelles $\{y_j^t\}_{j\in J}$ associées aux régions $\{Q_j^t\}_{j\in J}$
11: **Si** $J \neq \varnothing$ **Alors**
12: Entraîner le classificateur $\mathcal{C}_t$ à l'aide des annotations courantes et précédentes $\left\{ \left(y_j^k, \mathbf{d}_j^k\right) \right\}_{j\in J, k\in[\![0,t]\!]}$
13: **Fin Si**

ALGORITHME 5.1 – *Mécanisme d'apprentissage par retour interactif de pertinence dans le cadre de la détection de changements en ligne.*

après filtrage éventuel par le mécanisme de retour de pertinence, doivent toutes être affichées pour chaque image. Bien sur, ces contraintes n'empêchent par exemple pas de désigner à l'utilisateur les régions dont l'annotation permettrait un gain substantiel d'information vis-à-vis du classificateur. Cependant, elles limitent le gain de performances de ces méthodes d'apprentissage actif par rapport au cadre plus ouvert de la fouille de données.

Par conséquent, nous avons choisi d'implémenter ce mécanisme d'apprentissage par retour de pertinence de la manière suivante. À la réception de chaque nouvelle image de test, les régions candidates sont d'abord extraites grâce aux algorithmes présentés jusqu'ici. Si le classificateur n'a pas encore été défini, nous considèrons que toutes les régions détectées correspondent à la catégorie des changements pertinents. Sinon, le classificateur est utilisé pour décider si chaque région détectée correspond à un changement pertinent ou non. L'analyste image peut alors corriger les catégories de chaque région détectée, et s'il le fait, les annotations fournies sont utilisées pour mettre à jour le classificateur, qui passe ensuite à l'image suivante. Le pseudo-code de cette approche est fourni par l'algorithme 5.1.

Les sections suivantes décrivent les détails techniques relatifs à la mise en œuvre du mécanisme d'apprentissage par retour de pertinence, et plus précisément le descripteur et la méthode de classification utilisés.

5.3.2 Descripteur de régions

Le descripteur employé pour caractériser les régions dans une image, dans le cadre du mécanisme d'apprentissage par retour interactif de pertinence, doit permettre de distinguer un changement pertinent d'une variation parasite. Cela nécessite donc de caractériser une région donnée en analysant son évolution entre les observations de référence et de test, par exemple en exploitant des mesures issues de la carte de scores de détection de changements ou calculées conjointement dans les images de référence et de test.

D'autre part, comme dans tout problème d'apprentissage statistique, il est important de s'interroger sur la dimension des descripteurs comparée au nombre d'exemples d'apprentissage. En effet, pour obtenir un résultat de bonne qualité, il est crucial de disposer de suffisamment d'exemples d'apprentissage pour explorer correctement l'espace des descripteurs. Or, dans le cadre d'un apprentissage par retour de pertinence, chaque annotation nécessite un effort de la part de l'utilisateur. Il est donc peu réaliste d'espérer disposer d'un très grand nombre

d'exemples d'apprentissages, et il est par conséquent nécessaire d'employer un descripteur dont la dimension n'est pas trop importante. Le problème du manque de données d'apprentissage influe également sur le choix du classificateur, ce qui sera abordé à la section 5.3.3.

Par conséquent, nous avons proposé un descripteur constitué de 22 mesures, chacune exprimant un a priori sur la distinction entre changement pertinent et variation parasite. Afin de pouvoir être calculées rapidement, les mesures utilisées sont très simples et se basent notamment sur des caractéristiques de forme, de couleur, d'intensité et de gradient. Naturellement, elles seraient individuellement très peu performantes pour prédire correctement la classe d'une détection donnée, mais une combinaison pertinente de ces mesures peut être déterminée par apprentissage.

La suite de cette section décrit les différentes composantes du descripteur utilisé, en justifiant pour chacune l'intérêt dans le cadre de la détection de changements. Dans la suite, nous désignons l'image de test courante et la carte de scores associée respectivement par I^{test} et S. Nous désignons également le rendu du modèle de référence selon le point de vue de l'image de test courante par I^{ref}. Pour une image donnée I, nous désignons respectivement par I_R, I_G et I_B les canaux rouge, vert et bleu associés à l'image. Enfin, nous considérons une région donnée désignée par Q, dont la bordure extérieure est désignée par ∂Q. Le descripteur associé à cette région est désigné par $\mathbf{d}_Q \in \mathbb{R}^{22}$, et ses diverses composantes par $\left\{ {d_Q}^{(i)} \right\}_{i \in [\![1,22]\!]}$.

Exploitation de l'intensité L'intensité des observations a été beaucoup utilisée dans la littérature pour distinguer les changements pertinents des variations parasites. En particulier, même si cette mesure est peu pertinente en cas de variation des conditions d'illumination, nous pouvons penser que plus la différence d'intensité entre une observation de référence et de test est importante, plus il y a de chances qu'un changement significatif soit survenu. Par conséquent, les trois premières composantes du descripteur mesurent respectivement la différence moyenne d'intensité, sur la région considérée, entre les canaux rouge, vert et bleu des images de référence et de test :

$$ {d_Q}^{(1)} = \frac{1}{\text{card}(Q)} \sum_{q \in Q} \left| I_R^{\text{test}}(q) - I_R^{\text{ref}}(q) \right| \tag{5.10} $$

$$ {d_Q}^{(2)} = \frac{1}{\text{card}(Q)} \sum_{q \in Q} \left| I_G^{\text{test}}(q) - I_G^{\text{ref}}(q) \right| \tag{5.11} $$

$$ {d_Q}^{(3)} = \frac{1}{\text{card}(Q)} \sum_{q \in Q} \left| I_B^{\text{test}}(q) - I_B^{\text{ref}}(q) \right| \tag{5.12} $$

$$ \tag{5.13} $$

D'autre part, Watanabe et al. [112] ont montré que le rapport $\frac{I_L^{\text{test}}(q)}{I_L^{\text{ref}}(q)}$ des intensités en niveaux de gris prend des gammes de valeurs différentes selon qu'aucun changement ne soit survenu, qu'une variation d'ombre soit survenue ou qu'un changement significatif soit survenu. En particulier, ils montrent qu'en cas de variation d'ombre, la valeur du rapport d'intensité est soit très grande soit très faible. Cela correspond à l'intuition car si une ombre est présente dans l'une des observations mais pas dans l'autre, le rapport des intensité aura soit une valeur faible au numérateur soit une valeur faible au dénominateur. Pour exploiter ce constat, nous avons donc défini la grandeur suivante, où ε représente une valeur très faible (en pratique égale à 3 pour des observations dans $[\![0,255]\!]$) :

$$ r\left(I_L^{\text{test}}(q), I_L^{\text{ref}}(q) \right) = \begin{cases} 0 & \text{si } \left| I_L^{\text{test}}(q) - I_L^{\text{ref}}(q) \right| < \varepsilon \\ \frac{2}{\frac{I_L^{\text{test}}(q)}{I_L^{\text{ref}}(q)} + \frac{I_L^{\text{ref}}(q)}{I_L^{\text{test}}(q)}} & \text{sinon} \end{cases} \tag{5.14} $$

En cas de variation d'ombre au point q, nous avons $I_L^{\text{ref}}(q) \ll I_L^{\text{test}}(q)$ ou $I_L^{\text{test}}(q) \ll I_L^{\text{ref}}(q)$, donc $I_L^{\text{test}}(q)/I_L^{\text{ref}}(q) + I_L^{\text{ref}}(q)/I_L^{\text{test}}(q)$ tend vers $+\infty$. D'autre part, nous souhaitons que le cas

où il n'y a aucun changement au point q, c'est-à-dire où $\left|I_L^{\text{test}}(q) - I_L^{\text{ref}}(q)\right| < \varepsilon$, corresponde à la même classe que le cas où il y a variation d'ombre en q. Ainsi, la grandeur définie à l'équation 5.14 ne prend des valeurs proches de 1 que dans les cas correspondant à la présence d'un changement non causé par l'illumination, donc supposé d'intérêt potentiel.

En pratique, pour plus de robustesse, nous calculons dans un premier temps l'histogramme $\mathcal{H}$ des valeurs de $r\left(I_L^{\text{test}}(q), I_L^{\text{ref}}(q)\right)$ pour tous les pixels $q \in Q$. Cet histogramme $\mathcal{H}$, constitué de huit intervalles uniformément répartis sur $[0,1]$, permet de déterminer l'intervalle de valeurs le plus représenté. Nous définissons alors la fonction $\delta_{\mathcal{H}}(q)$ égale à 1 si la valeur de $r\left(I_L^{\text{test}}(q), I_L^{\text{ref}}(q)\right)$ est comprise dans l'intervalle le plus représenté, et à 0 sinon. La quatrième composante du descripteur est alors définie comme suit :

$$d_Q{}^{(4)} = \frac{1}{\sum_{q\in Q}\delta_{\mathcal{H}}(q)} \sum_{q\in Q} \delta_{\mathcal{H}}(q) \cdot r\left(I_L^{\text{test}}(q), I_L^{\text{ref}}(q)\right) \tag{5.15}$$

Exploitation de la couleur Comme évoqué à la section 3.2, la couleur peut également permettre de distinguer les changements pertinents des variations parasites, notamment celles dues à l'illumination.

En particulier, sous l'hypothèse que la couleur de l'illumination ne change pas, la teinte est une mesure intéressante pour la comparaison entre des observations acquises sous différentes conditions d'illumination [47]. Ainsi, si la différence de teinte entre une observation de référence et une observation de test est importante, cela peut indiquer qu'un changement significatif a eu lieu. Au contraire, si cette différence est faible, cela peut indiquer qu'aucun changement ou qu'un changement non significatif a eu lieu. Par conséquent, nous avons défini la cinquième composante de notre descripteur à l'aide de ce principe, en considérant la différence de teinte moyenne sur la région pour plus de robustesse :

$$d_Q{}^{(5)} = \frac{1}{\text{card}(Q)} \sum_{q\in Q} \min\left[\left|\text{hue}\left(I^{\text{test}}(q)\right) - \text{hue}\left(I^{\text{ref}}(q)\right)\right|,\ 360 - \left|\text{hue}\left(I^{\text{test}}(q)\right) - \text{hue}\left(I^{\text{ref}}(q)\right)\right| \right] \tag{5.16}$$

où $\text{hue}\left(I(q)\right)$ représente la teinte de l'observation $I(q)$ sous la forme d'un angle exprimé en degrés.

D'autre part, l'espace de représentation des couleurs décrit à la section 3.2.3, basé sur les coordonnées chromatiques logarithmiques, fournit un cadre permettant d'estimer la proportion d'information indépendante de l'illumination dans une observation donnée. Or, le fait que la proportion d'information indépendante de l'illumination soit faible pour une observation donnée peut affecter sa fiabilité vis-à-vis de la détection de changement, et peut donc influencer la décision concernant sa catégorie. Il peut donc être intéressant d'inclure cette mesure dans le descripteur.

Pour cela, considérons un pixel $q \in Q$ donné, dont l'observation associée dans l'image de test est représentée par $\boldsymbol{\chi}(q) = (\chi_R, \chi_G, \chi_B)$ en coordonnées chromatiques logarithmiques. La part d'information indépendante de l'illumination peut être obtenue en projetant $\boldsymbol{\chi}(q)$ de manière orthogonale au vecteur $\mathbf{V}_{illum}(\xi)$ (voir section 3.2). Il est donc possible d'estimer la proportion d'information dépendant de l'illumination dans $\mathbf{c}$ grâce à la fonction $\iota_{illum}\left(\xi, \boldsymbol{\chi}(q)\right)$ définie de la façon suivante :

$$\iota_{illum}\left(\xi, \boldsymbol{\chi}(q)\right) = \frac{\left|\left\langle \boldsymbol{\chi}(q) \middle| \mathbf{V}_{illum}(\xi) \right\rangle\right|}{\|\boldsymbol{\chi}(q)\|} \tag{5.17}$$

où la notation $\langle\cdot|\cdot\rangle$ représente le produit scalaire entre deux vecteurs. Cette mesure étant sensible au bruit dans l'image, nous calculons dans un premier temps l'histogramme $\mathcal{H}$ des valeurs de ι_{illum} pour tous les pixels $q \in Q$. Cet histogramme $\mathcal{H}$, constitué de huit intervalles uniformément répartis sur $[0,1]$, permet de déterminer l'intervalle de valeurs le plus représenté. Nous

définissons alors la fonction $\delta_{\mathcal{H}}(q)$ égale à 1 si la valeur de $\iota_{illum}\big(\xi, \boldsymbol{\chi}(q)\big)$ est comprise dans l'intervalle le plus représenté, et à 0 sinon. Nous avons alors défini la sixième composante du descripteur comme suit :

$$d_Q{}^{(6)} = \frac{1}{\sum_{q\in Q} \delta_{\mathcal{H}}(q)} \sum_{q\in Q} \delta_{\mathcal{H}}(q) \cdot \iota_{illum}\big(\xi, \boldsymbol{\chi}(q)\big) \tag{5.18}$$

Exploitation du gradient D'autre part, pour distinguer un changement pertinent d'une variation parasite, il peut être intéressant d'analyser le gradient dans la carte des scores, au niveau de la bordure d'une région donnée, ainsi que l'évolution du gradient image entre les images de test et de référence. Pour cela, dans le contexte d'un calcul de gradient, la notion de bordure $\partial Q_\perp$ associée à une région Q donnée peut être formalisée grâce à l'ensemble de couples de points défini comme suit :

$$\partial Q_\perp = \Big\{ (q, q_b) \in Q \times \partial Q, \text{ tel que } q \text{ est adjacent à } q_b \text{ et } q - q_b \text{ est orthogonal à la tangente à } Q \text{ en } q_b \Big\} \tag{5.19}$$

Étant intéressés par des changements relatifs aux structures artificielles dans la scène, les frontières des changements pertinents auront tendance à être nettes et par conséquent le gradient au niveau de la bordure sera élevé. Au contraire, les frontières des variations parasites, en particulier celles dues à l'illumination, auront tendance à être floues et par conséquent le gradient dans les images ou la carte des scores, au niveau de la bordure, sera faible. Nous avons donc utilisé ce principe pour définir les quatre composantes suivantes de notre descripteur de régions.

La septième composante du descripteur mesure le gradient moyen dans la carte de score au niveau de la bordure de la région considérée et est définie comme suit :

$$d_Q{}^{(7)} = \frac{1}{\mathrm{card}(\partial Q_\perp)} \sum_{(q,q_b)\in\partial Q_\perp} |S(q) - S(q_b)| \tag{5.20}$$

De manière similaire, les trois composantes suivantes du descripteur mesurent respectivement l'évolution moyenne du gradient au niveau de la bordure de la région considérée, entre l'image de référence et celle de test, respectivement pour les canaux rouge, vert et bleu :

$$\begin{aligned}
d_Q{}^{(8)} &= \frac{1}{\mathrm{card}(\partial Q_\perp)} \sum_{(q,q_b)\in\partial Q_\perp} \Big| \Big| I_R^{\text{test}}(q) - I_R^{\text{test}}(q_b) \Big| - \Big| I_R^{\text{ref}}(q) - I_R^{\text{ref}}(q_b) \Big| \Big| \\
d_Q{}^{(9)} &= \frac{1}{\mathrm{card}(\partial Q_\perp)} \sum_{(q,q_b)\in\partial Q_\perp} \Big| \Big| I_G^{\text{test}}(q) - I_G^{\text{test}}(q_b) \Big| - \Big| I_G^{\text{ref}}(q) - I_G^{\text{ref}}(q_b) \Big| \Big| \\
d_Q{}^{(10)} &= \frac{1}{\mathrm{card}(\partial Q_\perp)} \sum_{(q,q_b)\in\partial Q_\perp} \Big| \Big| I_B^{\text{test}}(q) - I_B^{\text{test}}(q_b) \Big| - \Big| I_B^{\text{ref}}(q) - I_B^{\text{ref}}(q_b) \Big| \Big|
\end{aligned} \tag{5.21}$$

Exploitation du taux de variation de l'aire Comme expliqué à la section 5.2, une analyse fine de la carte de scores peut permettre de distinguer les changements pertinents des variations parasites. Le descripteur que nous utilisons intègre donc la notion de stabilité de la région considérée, qui est inversement proportionnelle au taux de variation de l'aire de la région. Cette stabilité est calculée pour huit valeurs différentes du paramètre Δ afin de donner une description multi-échelle de la région dans la carte de scores :

$$\text{Pour } i \in [\![0,7]\!],\ d_Q{}^{(11+i)} = \text{stabilité}_{\Delta_i}(Q) \quad \text{avec } \Delta_i = \frac{1+2i}{255} \tag{5.22}$$

Exploitation de la forme Enfin, la forme de la région considérée peut également indiquer si elle correspond plutôt à un changement significatif ou à une variation parasite. Par exemple, les faux positifs dûs à d'éventuels problèmes de recalage (près de routes, de bâtiments, etc) ont tendance à avoir une longueur importante mais une faible épaisseur, typiquement de quelques pixels seulement. Par conséquent, la forme d'une région peut également être utile pour distinguer un changement d'intérêt potentiel d'une détection parasite, et nous avons donc défini les quatre dernières composantes du descripteur à l'aide de diverses mesures caractéristiques de la forme des régions.

Pour commencer, nous avons utilisé une mesure caractérisant l'élongation de la région considérée, c'est-à-dire le rapport entre sa longueur et son épaisseur. En effet, nous pouvons nous attendre à ce que les faux positifs dûs aux problèmes de recalage donnent lieu à des régions dont l'élongation est très importante (e.g. le long de routes, de bâtiments etc) tandis qu'une région correspondant à une structure artificielle aura une forme plus équilibrée et donc une élongation plus ou moins proche de 1. La composante suivante du descripteur est donc une mesure de l'élongation de la région, calculée grâce à la matrice de covariance des positions des pixels dans la région.

Plus précisément, soit $\Sigma = \left[\begin{smallmatrix} v_0 & c \\ c & v_1 \end{smallmatrix}\right]$ la matrice de covariance des positions des pixels dans la région considérée, et soit λ_0 et λ_1 les valeurs propres associées. Nous nous intéressons à la grandeur suivante, permettant de distinguer les régions aux élongations importantes des régions aux élongations proche de 1, indépendamment des valeurs relatives de λ_0 et λ_1 :

$$\begin{aligned} r(\lambda_0, \lambda_1) &= \frac{2}{\sqrt{\frac{\lambda_0}{\lambda_1}} + \sqrt{\frac{\lambda_1}{\lambda_0}}} \\ &= \frac{2\sqrt{\lambda_0 \lambda_1}}{\lambda_0 + \lambda_1} \\ &= \frac{2\sqrt{\det(\Sigma)}}{\mathrm{trace}(\Sigma)} \end{aligned}$$

Nous définissons donc la dix-neuvième composante du descripteur de la manière suivante :

$$d_Q{}^{(19)} = 2\frac{\sqrt{v_0 v_1 - c^2}}{v_0 + v_1} \tag{5.23}$$

La non-compacité d'une région peut également servir à éliminer les détections parasites aux formes trop complexes et étirées pour correspondre à un changement pertinent. La vingtième composante du descripteur mesure donc la non-compacité de la région définie comme le rapport du carré du périmètre par l'aire de la région :

$$d_Q{}^{(20)} = \frac{\mathrm{card}(\partial Q)^2}{\mathrm{card}(Q)} - 4\pi \tag{5.24}$$

De plus, pour les mêmes raisons, le descripteur intègre deux mesures supplémentaires mesurant la régularité de la bordure de la forme. Cette régularité peut être mesurée grâce à l'entropie de l'histogramme des directions des tangentes à la bordure de la région considérée. La régularité des directions des tangentes à la bordure peut être de deux types, menant à deux mesures distinctes. La première mesure analyse la régularité de ces directions dans le référentiel absolu de l'image, et permet par exemple de distinguer une forme rectangulaire d'une forme plus chaotique. La seconde mesure analyse la régularité de ces directions dans le référentiel local du point de contact entre la tangente et la bordure, et permet par exemple de distinguer un cercle d'une forme plus chaotique.

Plus formellement, soit $\{\mathbf{V}_{\tan}(q_b)\}_{q_b \in \partial Q}$ la séquence des directions des tangentes à la bordure de la région Q considérée. Désignons par $\theta^{\mathrm{abs}}_{\tan}(q_b)$ l'angle, dit absolu, formé par le vecteur $\mathbf{V}_{\tan}(q_b)$ par rapport à l'axe Ox dans l'image. Désignons de plus par $\alpha_Q(q_b)$ l'angle formé par le vecteur $q_b - \boldsymbol{\mu}_Q$ par rapport à l'axe Ox dans l'image, où $\boldsymbol{\mu}_Q$ représente le barycentre de la région

Q. Désignons enfin par $\theta^{\text{rel}}_{\text{tan}}(q_b) = \theta^{\text{abs}}_{\text{tan}}(q_b) - \alpha_Q(q_b)$ l'angle, dit relatif, formé par le vecteur $\mathbf{V}_{\text{tan}}(q_b)$ par rapport au vecteur $q_b - \boldsymbol{\mu}_Q$. Il est alors possible de former deux histogrammes $\mathcal{H}^{\text{abs}}$ et $\mathcal{H}^{\text{rel}}$ concernant respectivement les angles absolus et relatifs vis-à-vis des points de la bordure de la région considérée. Nous définissons alors les deux dernières composantes du descripteur de la manière suivante :

$$d_Q{}^{(21)} = \text{entropie}(\mathcal{H}^{\text{abs}}) \tag{5.25}$$

$$d_Q{}^{(22)} = \text{entropie}(\mathcal{H}^{\text{rel}}) \tag{5.26}$$

Notons que les régions dont la taille est inférieure à une dizaine de pixels sont éliminées. En effet, dans la grande majorité des cas, elles ne représentent que l'influence du bruit mais pourraient être classées de manière erronée par les différentes mesures de formes présentées ci-dessus.

Comme mentionné plus haut, il est clair que, considérées individuellement, ces mesures sont insuffisantes pour faire correctement la distinction entre changements pertinents et variations parasites. Cependant, considérées conjointement et avec l'aide des annotations fournies par l'analyste image, elles permettent d'obtenir une bonne précision de prédiction, comme le démontrent les résultats présentés au chapitre 6.

5.3.3 Classification des régions

Pour la classification des régions, dans le cadre de l'apprentissage par retour de pertinence, nous avons choisi d'utiliser l'algorithme des machines à vecteurs de support (SVM, pour *Support Vector Machines* dans la littérature [31]). Ce choix est justifié par le principe de maximisation de la marge mis en œuvre dans les SVM, qui permet de garantir une bonne capacité de généralisation, y compris dans les cas délicats où peu d'exemples d'apprentissage sont disponibles. Cette propriété est appréciable dans notre cas d'application. En effet, elle permet d'obtenir un classificateur exploitant le peu d'information disponible de manière judicieuse, y compris durant la phase initiale du mécanisme de retour de pertinence, au cours de laquelle l'utilisateur n'a encore fourni que très peu d'exemples d'apprentissage.

D'autre part, toujours dans le but de limiter la quantité d'exemples d'apprentissage requis, nous avons choisi d'utiliser des SVM linéaires. En adéquation avec ce choix, la définition du descripteur, présentée à la section précédente, a été conçue pour séparer au maximum les deux classes considérées. Pour vérifier ce point, nous avons effectué une analyse discriminante des descripteurs associées aux régions détectées dans la vidéo *Aérodrome 1* (voir le chapitre 6). La figure 5.5 présente la projection des points de l'espace des descripteurs dans un sous-espace à deux dimensions permettant de maximiser la séparation des classes de changements et de non-changements. Cette visualisation montre qu'une séparation linéaire stricte de ces deux classes n'est pas possible, du fait d'un léger recouvrement des deux modes et de points extrêmes très dispersés. Cependant, il est clair qu'une séparation linéaire peut permettre d'obtenir une classification de bonne qualité, le taux d'erreur étant d'autant plus faible que nous travaillons dans l'espace à 22 dimensions plutôt que dans le sous-espace à deux dimensions. Par conséquent, les performances de classification obtenues à l'aide de l'approche linéaire s'étant avérées satisfaisantes, il n'a pas été jugé utile d'introduire l'utilisation de noyaux, qui aurait de plus présenté l'inconvénient d'augmenter les temps de traitement.

Par ailleurs, notre choix d'utiliser des SVM pour l'apprentissage peut sembler incompatible avec le cadre incrémental d'un mécanisme de retour de pertinence. En effet, les SVM n'ont pas été conçues pour permettre l'apprentissage incrémental sur des données obtenues progressivement, bien que certaines techniques aient été proposées dans ce but. Pour contourner ce problème, nous effectuons un nouvel apprentissage sur l'ensemble des données disponibles à chaque fois que de nouvelles annotations sont reçues. Ce procédé reste extrêmement rapide tant

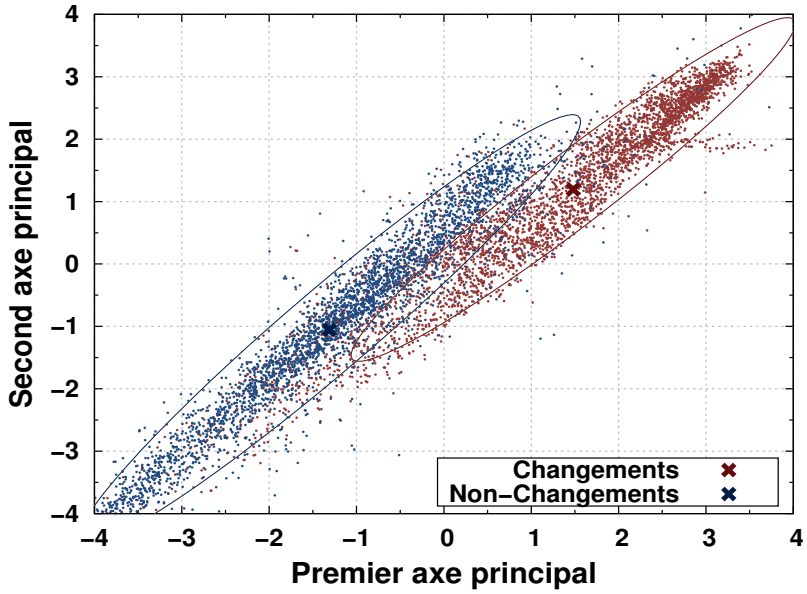


FIGURE 5.5 – *Cette figure présente les résultats de l'analyse discriminante effectuée sur les descripteurs associés aux régions détectées sur la vidéo* Aérodrome 1. *Le graphique montre la projection des points dans un sous-espace à deux dimensions permettant de maximiser la séparation des classes de changements (rouge) et de non-changements (bleu). Les ellipses correspondent à l'approximation gaussienne des distributions pour une confiance de* 99.995%, *ce qui permet de constater que les distributions ne sont pas gaussiennes (en particulier celle correspondant aux changements).*

que les données d'apprentissage sont en quantité raisonnable, ce qui coïncide avec l'hypothèse selon laquelle l'utilisateur ne fournira vraisemblablement qu'une quantité limitée d'annotations.

À chaque fois que de nouvelles annotations sont disponibles, elles sont ajoutées à l'ensemble des données d'apprentissage disponibles. Le classificateur est alors recalculé pour correspondre à l'ensemble de ces données d'apprentissage, qui sont préalablement normalisées (moyenne nulle et variance unité). De plus, lors de l'exécution de l'algorithme d'apprentissage, nous utilisons une pondération différente pour les deux classes. En effet, les exemples d'apprentissage correspondant à des changements pertinents pouvant être plus rares que ceux correspondant à des variations parasites, il convient d'imposer une forte pénalité en cas de classification erronée de ces exemples, afin d'éviter qu'ils ne soient tout simplement ignorés. Pour cela, nous attribuons en pratique une pondération à chaque classe correspondant à la proportion d'exemples d'apprentissage disponibles pour la classe alternative. Ainsi, si une classe est largement plus représentée que la classe alternative parmi les exemples d'apprentissage, elle sera associée à un faible coefficient de pondération, et inversement.

5.4 Bilan

Les techniques de consolidation présentées dans ce chapitre permettent d'affiner la précision de la détection de changements, d'une part en exploitant la redondance présente dans la vidéo de test, et d'autre part en intégrant des modèles a priori caractérisant les changements visés. Pour aller plus loin, nous avons montré comment une interaction avec l'analyste image, via un mécanisme de retour de pertinence, pouvait permettre d'adapter la détection de changements à ses besoins sans alourdir les traitements. Pour démontrer l'intérêt de notre approche, le prochain chapitre présente les résultats d'évaluation quantitative relatifs aux algorithmes présentés dans ce chapitre et les deux chapitres précédents.

Chapitre 6

Évaluation quantitative

LES chapitres précédents ont décrit l'approche modulaire proposée dans le cadre de cette thèse pour effectuer la détection de changements dans des vidéos aériennes. Cette modularité permet de rendre indépendants les différents traitements mis en œuvre, et donc de pouvoir comparer les performances associées à différentes approches pour chaque traitement.

Le présent chapitre présente donc les expérimentations effectuées dans le but d'évaluer notre approche de détection de changements. Ces expérimentations ont permis de quantifier les performances globales vis-à-vis de la tâche de détection de changements. Elles ont également permis d'analyser les performances intermédiaires de certains modules ainsi que l'impact de chacun sur les performances finales. Les différents cas de figure testés au cours de ces expérimentations ont ainsi permis de mesurer la robustesse de l'approche, c'est-à-dire sa tolérance, par rapport à divers facteurs perturbateurs tels que l'incertitude sur la pose de la caméra, l'incertitude sur les élévations dans la scène, et ainsi de suite.

Ce chapitre commence avec la section 6.1, qui présente les données réelles et synthétiques utilisées pour l'évaluation de notre approche. Les sections suivantes présentent les expérimentations réalisées et analysent les résultats obtenus. La section 6.2 concerne les pré-traitements et regroupe les expérimentations relatives à la géo-localisation, aux élévations dans la scène et au traitement de l'illumination. La section 6.3 compare différents algorithmes de modélisation d'apparence. La section 6.4 regroupe ensuite les expérimentations relatives aux traitements de consolidation des résultats. Enfin, la section 6.5 fournit une analyse sur quelques aspects généraux de notre approche, ainsi qu'une discussion sur les résultats obtenus.

Sommaire

FIGURE 6.1 – *Cette figure illustre les possibilités de génération de vidéos aériennes synthétiques du logiciel VBS2, avec l'exemple d'une vidéo acquise depuis une trajectoire parfaitement circulaire au dessus d'un village virtuel.*

6.1 Données d'évaluation

L'évaluation des résultats obtenus par un algorithme quelconque a deux objectifs principaux : déterminer la précision de ces résultats et quantifier la robustesse de l'approche par rapport à divers facteurs perturbateurs. Dans le premier cas, il est nécessaire de disposer de la vérité-terrain associée aux données, et il peut alors être utile d'avoir recours à des données synthétiques pour lesquelles la vérité-terrain est facile à obtenir. Dans le second cas, il est préférable de travailler avec des données réelles, afin de confronter les algorithmes à des perturbations réalistes.

Par conséquent, l'évaluation des méthodes et algorithmes développés dans le cadre de cette thèse a été effectuée à l'aide de deux types de vidéos aériennes : des vidéos synthétiques, présentées à la section 6.1.1, et des vidéos réelles, présentées à la section 6.1.2. Nous avons vu au chapitre 2 que l'acquisition de données réelles dans le contexte de la détection de changements entre vidéos était une tâche lourde et délicate, exigeant de plus un fastidieux travail d'annotation pour obtenir la vérité-terrain. Pour contourner ce problème, nous avons donc développé une approche [18], présentée à la section 6.1.3, consistant à insérer des changements virtuels dans des vidéos réelles par réalité augmentée.

6.1.1 Données synthétiques

Afin d'évaluer certains traitements en conditions contrôlées, nous avons parfois eu recours à des vidéos aériennes synthétiques. Ces données synthétiques sont particulièrement appréciables pour l'évaluation de certains algorithmes, pour lesquels l'acquisition de données pertinentes ou l'extraction de la vérité-terrain est difficile en pratique. Cela est notamment intéressant pour évaluer les algorithmes de géo-localisation ou pour l'extraction de modèles 3D correspondant à la scène observée.

Les vidéos synthétiques utilisées ont été générées à l'aide du logiciel de simulation photo-réaliste Virtual Battle Station 2 (VBS2), développé par Bohemia Interactive Simulation. Ce logiciel possède une large bibliothèque de modèles permettant une modélisation réaliste et dynamique de vastes régions géographiques, dont la surface dépasse la centaine de kilomètres carré. D'autre part, ce logiciel dispose de nombreuses fonctionnalités intéressantes, telles que la simulation de conditions météorologiques précises ou le contrôle des conditions d'illumination. Enfin, VBS2 présente l'avantage considérable de permettre l'extraction de la vérité-terrain, et plus précisément d'extraire les trajectoires d'acquisition réelles, les modèles 3D des régions géographiques observées, ainsi que les masques de changements réels.

(a) Trajectoires

FIGURE 6.2 – *Cette figure présente à gauche, les trajectoires d'acquisition estimées superposées avec une carte Google Maps de la région (a), et à droite, des échantillons d'images (b)-(f), pour les vidéos aériennes acquises au dessus de l'aérodrome de Darois, près de Dijon. Les trajectoires des vidéos* Aérodrome 1 *à* Aérodrome 5 *sont respectivement tracées en rouge, blanc, bleu, vert et noir. Copyright © 2010 - 2012 Cassidian - All rights reserved.*

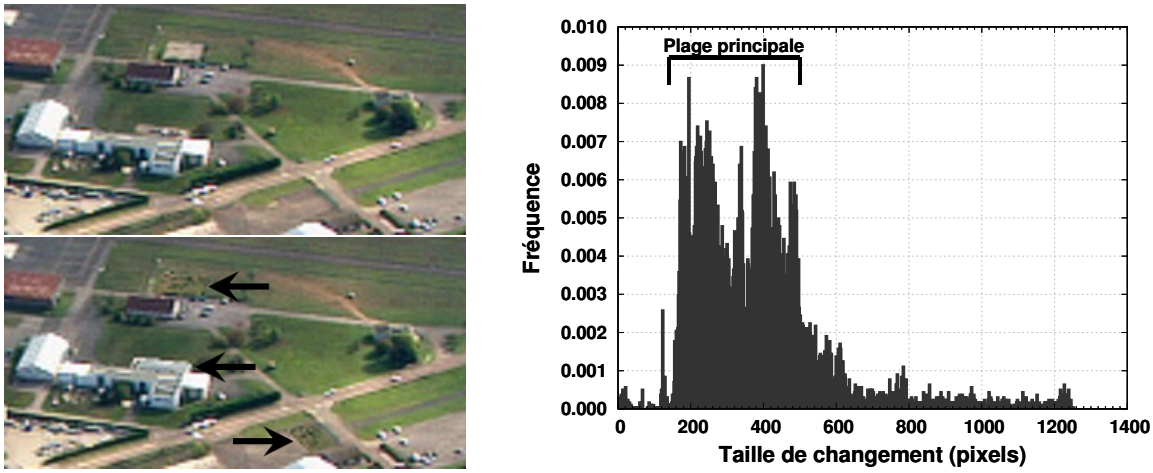
La figure 6.1 présente l'exemple d'une vidéo de synthèse, générée à l'aide de VBS2 pour l'évaluation des algorithmes de géo-localisation. Cependant, les vidéos de synthèse pouvant être générées très facilement en fonction des besoins d'évaluation, elles seront décrites plus en détails au cas par cas dans les sections concernées.

Toutefois, malgré leur apparence visuellement très réaliste, ces données de synthèse ne sont pas suffisantes pour évaluer correctement les performances en traitement d'image. En effet, les données synthétiques sont généralement générées selon diverses hypothèses idéales concernant le capteur ou la scène observée, éliminant de nombreux effets perturbateurs tels que le bruit dans les images, la gamme dynamique limitée du capteur, les effets de flou, etc. Il est par conséquent nécessaire d'évaluer également la méthode à l'aide de données réelles.

6.1.2 Données réelles

Les vidéos aériennes réelles utilisées pour nos évaluations ont été acquises dans le cadre d'une campagne d'acquisition de données d'observation aérienne menée par Cassidian, en collaboration avec Astrium Satellites et EADS Innovation Works. Ces données d'observation ont été acquises en France par un avion équipé d'une tourelle d'observation EO / IR. Cette tourelle a permis d'enregistrer le flux vidéo visible grâce à une caméra HD stabilisée (en 1280×720 pixels).

Les données utilisées pour l'évaluation correspondent à cinq vidéos acquises au dessus de l'aérodrome de Darois, près de Dijon, que nous désignerons dans la suite par *Aérodrome 1* à *Aérodrome 5*. Ces cinq vidéos correspondent à différentes conditions d'acquisitions, en termes de points de vue, de conditions d'illumination, de résolution au sol ou de délai temporel. La figure 6.2 présente les trajectoires d'acquisition de ces vidéos, estimées par la méthode d'interpolation présentée à la section 3.1.1, ainsi que quelques échantillons des images associées. La vidéo *Aérodrome 1* a été acquise le 14 octobre 2011 et est constituée de 215 images. Les vidéos *Aérodrome 2* et *Aérodrome 3* sont respectivement constituées de 1 137 et 462 images, et sont issues d'une unique vidéo scindée en deux (d'où la continuité des trajectoires), acquise quelques minutes après la vidéo *Aérodrome 1*. La vidéo *Aérodrome 4* est constituée de 154 images et a



(a) Exemples de changements virtuels (b) Histogramme des tailles

FIGURE 6.3 – *Cette figure présente une image (a) issue de la vidéo* Aérodrome 3 *avant et après insertion de changements virtuels (mis en évidence par les flèches noires), ainsi que l'histogramme des tailles de changements (b). Ces tailles, qui sont mesurées en nombre de pixels, dépendent de la profondeur du changement dans l'image. Un changement typique a ainsi une taille d'environ* 20×20 *pixels. Copyright © 2010 - 2012 Cassidian - All rights reserved.*

été acquise une dizaine de minutes après la première vidéo, avec une résolution au sol supérieure aux trois premières vidéos. Enfin, la vidéo *Aérodrome 5* est constituée de 71 images et a été acquise le 30 novembre 2011, soit 47 jours après les quatre premières. Elle contient par conséquent de nombreux changements par rapport aux quatre vidéos précédentes.

Les vidéos *Aérodrome 1* à *Aérodrome 3* seront utilisées très fréquemment dans ce chapitre pour l'évaluation de notre approche de détection de changements. Les deux autres vidéos, *Aérodrome 4* et *Aérodrome 5*, seront utilisées à la section 6.5 pour illustrer les limites de l'approche.

6.1.3 Données forgées par réalité augmentée

La campagne d'acquisition des données d'observation a été menée exclusivement de manière aérienne, et n'a pas été accompagnée d'une campagne au sol visant à mettre en œuvre des changements à détecter. Telles quelles, les vidéos aériennes présentées à la section précédente sont donc d'un intérêt limité pour la détection de changements. Par conséquent, afin d'évaluer les algorithmes développés dans le cadre de cette thèse, nous avons développé une méthode de réalité augmentée consistant à insérer des changements virtuels dans les vidéos brutes.

Cette méthode permet non seulement de générer un nombre illimité de vidéos pertinentes pour la détection de changements, mais également d'obtenir la vérité-terrain associée de manière rapide et automatique. D'autre part, les données d'évaluation qui en résultent sont adaptées au scénario opérationnel visé dans le cadre de cette thèse (voir section 1.2.3). En effet, le fait d'insérer des changements virtuels dans des vidéos acquises de manière rapprochée dans le temps permet de se rapprocher du cas applicatif selon lequel la plate-forme d'observation effectue des passages réguliers et des acquisitions fréquentes. Ceci permet donc de concentrer l'effort sur les difficultés principales rencontrées dans ce contexte, en particulier les problèmes liés aux points de vue arbitraires et aux variations modérées de contenu (e.g. illumination, objets mobiles, variations d'apparences, etc).

Pour cela, nous avons extrait, de diverses données d'observation, un certain nombre de textures rectangulaires, qui ont été insérées à des emplacements prédéfinis dans la scène observée de manière à former un changement. Plus précisément, étant donné un emplacement prédéfini, nous déterminons la zone correspondante dans chaque image de la vidéo considérée à l'aide de la géo-localisation (voir section 3.1.1). Il est alors possible de modifier les images issues des

vidéos brutes en y insérant les textures, qui se comportent de manière réaliste par rapport au déplacement de la caméra. Le masque de changement idéal peut également être obtenu, pour chaque image de la vidéo, par le même procédé. Notons que pour plus de réalisme, il pourrait être envisageable de faire correspondre l'apparence de la texture insérée avec les conditions locales de l'image (illumination, bruit, flou, etc). Cependant, ce niveau de réalisme n'est pas nécessaire pour évaluer les performances en détection de changements et par conséquent les traitements correspondants n'ont pas été mis en œuvre.

Pour l'évaluation des performances, et plus précisément l'estimation des courbes ROC, nous avons utilisé les vidéos *Aérodrome 1* à *Aérodrome 3*. La vidéo *Aérodrome 2* a servi de vidéo de référence, et les deux autres vidéos ont été utilisées comme vidéos de test, après insertion de changements virtuels. Une dizaine de changements, tous visibles dans chacune des images de ces deux vidéos, ont été insérés et répartis uniformément dans la scène observée.

La figure 6.3 illustre les principales caractéristiques des changements virtuels insérés dans les vidéos. Elle présente notamment un exemple d'image aérienne avant et après insertion de changements virtuels, qui montre que les changements insérés ne sont pas trivialement identifiables. Cette figure présente également l'histogramme des tailles de changements insérés, après projection dans les vidéos aériennes.

La suite de ce chapitre présente les résultats d'évaluation des méthodes développées dans le cadre de cette thèse.

6.2 Évaluation des pré-traitements

Cette section présente les résultats d'évaluation des méthodes de pré-traitement décrites au chapitre 3. Les évaluations portent sur les performances finales en détection de changements, et sur les résultats et temps d'exécution intermédiaires lorsqu'applicable.

6.2.1 Techniques de géo-localisation

Notre approche de détection de changements fait une utilisation intensive des matrices de projection associées à chaque image considérée, afin de mettre en correspondance leurs pixels avec les cellules du modèle 3D d'apparence. Par conséquent, les techniques de géo-localisation présentées à la section 3.1 jouent un rôle fondamental et il est donc important d'évaluer leur précision.

Cette précision peut être mesurée à l'aide de deux critères principaux : l'erreur d'estimation, relative à la valeur des paramètres d'acquisition (e.g. position, orientation, et calibration), et l'erreur de reprojection, caractérisant la précision de la mise en correspondance entre les points 3D de la scène et les points 2D correspondants dans l'image. Ces deux critères sont intéressants et ont tout deux été analysés, toutefois dans le cadre de la détection de changements, l'effort essentiel doit porter sur la minimisation de l'erreur de reprojection. En effet, nous pouvons tolérer que l'estimation des trajectoires soit approximative. En revanche, l'objectif est d'atteindre une bonne performance en détection de changements, ce qui sera impossible si la mise en correspondance des pixels et des modèles d'apparence n'est pas suffisamment précise.

Plus spécifiquement, pour l'évaluation de l'algorithme d'asservissement visuel, nous avons supposé connus les paramètres d'acquisition de la première image de la vidéo considérée et avons exécuté l'algorithme d'estimation en ligne sur le reste des images. Nous avons alors analysé les résultats associés aux algorithmes d'asservissement visuel dans le cas général et restreint, et nous avons également analysé l'impact de la technique d'accélération (voir section 3.1) sur la précision de l'estimation dans le cas restreint. Par ailleurs, pour l'évaluation de l'algorithme d'interpolation de pose, nous avons commencé par effectuer une calibration, à base de correspondances entre points de contrôle 3D et points 2D, pour un certain nombre d'images-clés réparties uniformément dans la vidéo considérée. Nous avons ensuite exécuté l'algorithme

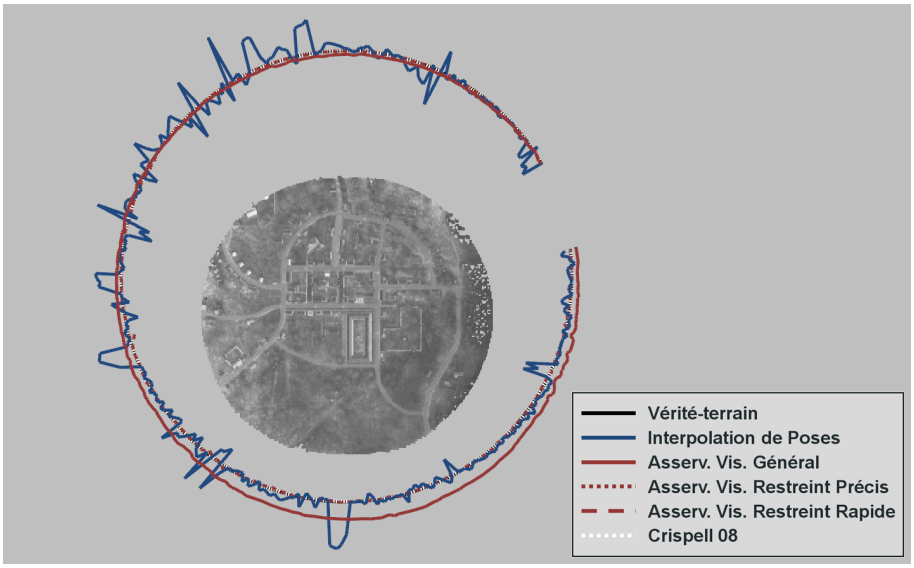


FIGURE 6.4 – *Cette figure compare la précision d'estimation des différents algorithmes de géo-localisation présentés à la section 3.1 d'un point de vue global. Les courbes associées à la vérité-terrain, à l'asservissement visuel restreint, dans les cas précis et rapide, et à la méthode de Crispell et al. [33] peuvent être difficiles à distinguer, car elles sont quasiment superposées.*

Algorithme	Temps total (min)	Temps par image (s)
Interpolation de pose	193	32.9
Asservissement visuel		
General	147	25.0
Restreint précis	135	23.0
Restreint rapide	30	5.0
Crispell 08	156	26.6

TABLE 6.1 – *Temps de calcul associés aux algorithmes de géo-localisation, selon les différentes versions, appliqués sur une vidéo de* 352 *images de taille* 800×600 *pixels.*

d'interpolation sur les images restantes de la vidéo. Enfin, pour comparer nos algorithmes avec une méthode de la littérature, nous avons implémenté et évalué la méthode de Crispell et al. [33]. Cette méthode est basée sur la combinaison d'un filtre de Kalman, permettant de prédire les paramètres d'acquisition courants à l'aide des précédentes estimations, avec une étape de correction par asservissement visuel. Cette dernière étape diffère cependant de la méthode proposée dans le cadre de cette thèse. En effet, la matrice Jacobienne utilisée pour l'asservissement est basée sur une approximation de la relation liant, d'une part le décalage de paramètres d'acquisition de deux caméras, et d'autre part la matrice de recalage entre les images correspondantes. Nous verrons plus bas que cette approximation a un impact significatif sur l'erreur de reprojection, ce qui rend la méthode de Crispell et al. [33] peu adaptée dans le contexte de la détection de changements.

Erreur d'estimation des trajectoires La figure 6.4 compare les trajectoires estimées à l'aide des différents algorithmes, à partir d'une vidéo aérienne synthétique de 352 images générée selon une trajectoire parfaitement circulaire. D'autre part, la table 6.1 compare les temps d'exécution obtenus avec chaque algorithme.

Dans le cas de notre algorithme d'interpolation de pose, nous avons utilisé 8 images-clés calibrées à l'aide des correspondances 2D-3D exactes associées à des points de contrôle générés automatiquement. Les résultats montrent que la trajectoire résultant de l'interpolation des

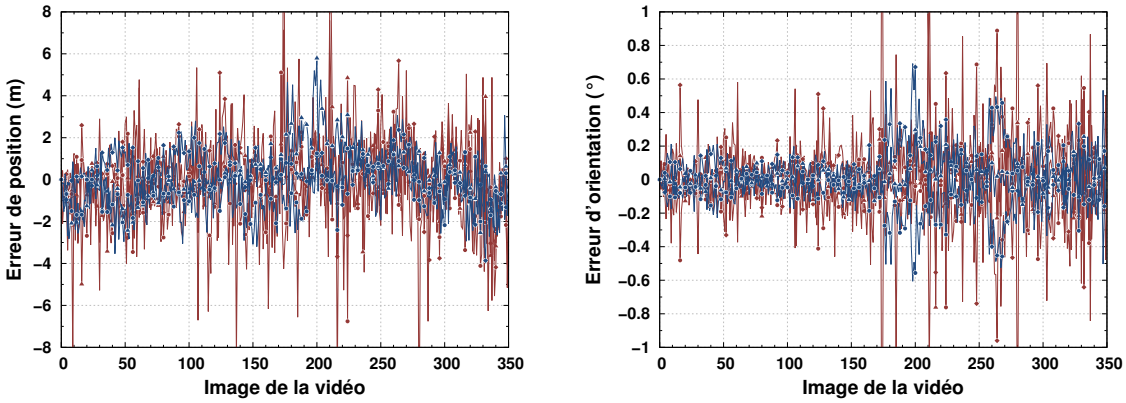
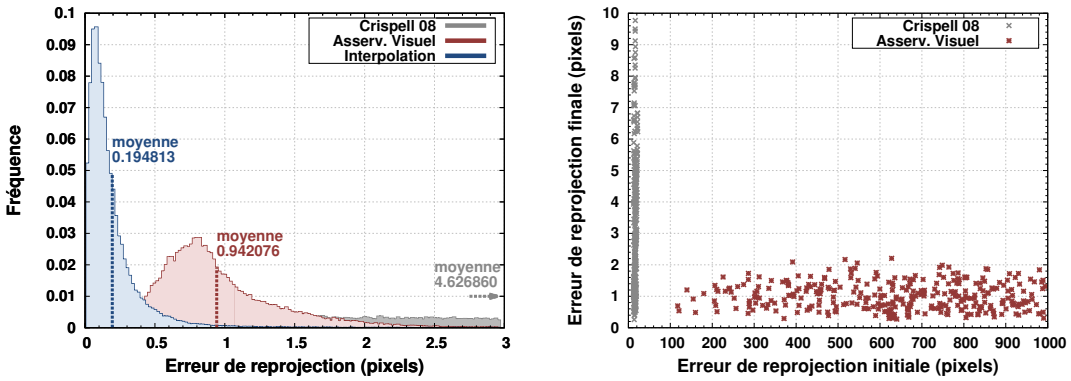


FIGURE 6.5 – *Cette figure compare la précision d'estimation des algorithmes précis (en bleu) et rapide (en rouge) d'asservissement visuel dans le cas restreint. Le graphique de gauche compare les erreurs d'estimation relatives aux positions, et celui de droite compare celles relatives aux orientations. Ces courbes sont très bruitées et leurs détails sont peu pertinents, mais elles permettent cependant de déterminer une tendance générale concernant l'amplitude des erreurs dans chaque cas.*

poses, bien que suivant correctement la tendance générale de la trajectoire réelle, est relativement bruitée. Ceci s'explique par le fait que, d'un point de vue des paramètres d'acquisition, il y a une forte ambiguïté entre d'une part une diminution de la distance entre la caméra et la scène, et d'autre part une augmentation du facteur de zoom. Par conséquent, la minimisation de l'erreur de reprojection des points de contrôle peut localement mener à un écart par rapport à la trajectoire réelle, qui est compensé par une légère augmentation ou diminution du zoom.

La trajectoire estimée par l'algorithme d'asservissement visuel présente une bien meilleure continuité. Dans le cas général où les paramètres de calibration sont estimés à chaque image, la trajectoire estimée présente des écarts par rapport à la trajectoire réelle. Comme pour l'algorithme d'interpolation de poses, ceci est également dû à l'ambiguïté entre le positionnement et les distances focales. Toutefois, dans le cas de l'asservissement visuel général, cette ambiguïté peut mener à une divergence de l'erreur d'estimation car aucune contrainte supplémentaire n'est disponible, à la différence de l'interpolation qui impose une contrainte plus forte. En pratique, cette divergence peut être limitée en effectuant un amortissement sur les ajustements apportés aux paramètres de calibration, afin de maîtriser leur évolution. En revanche, dans le cas restreint où les paramètres de calibration sont supposés constants (et donc connus via la donnée de ceux de la première image), les résultats montrent que la trajectoire estimée est quasiment superposée à la trajectoire réelle. De même, la trajectoire estimée à l'aide de la méthode de Crispell et al. [33] est aussi très proche de la vérité-terrain.

Les trajectoires estimées à l'aide des algorithmes précis et rapide d'asservissement visuel dans le cas restreint sont quasiment identiques. Par conséquent, la figure 6.5 compare en détail la précision obtenue à l'aide de chacun de ces algorithmes. Ces graphiques présentent la superposition des courbes d'erreurs de position en x, y et z et d'orientation en ψ, θ et ϕ, en fonction des indices des images de la vidéo considérée. Les détails de ces courbes sont difficilement lisibles, du fait de leurs fortes variations, mais leur intérêt est moins dans le détail que dans la tendance générale qui en ressort. Ainsi, ces courbes montrent que l'amplitude des variations est légèrement plus faible pour l'algorithme précis ($\pm$6m en position et $\pm 0.6°$ en orientation) que pour l'algorithme rapide (plus de $\pm$8m en position et plus de $\pm 1°$ en orientation). Cependant, la table 6.1 montre que l'algorithme rapide permet de gagner un facteur 5 sur le temps d'exécution, ce qui peut justifier son utilisation de préférence à l'algorithme précis.



(a) (b)

FIGURE 6.6 – *Cette figure analyse les erreurs de reprojection associées à la géo-localisation. Le graphique de gauche (a) compare les histogrammes des erreurs de reprojection pour les algorithmes d'interpolation, d'asservissement visuel (restreint précis) et pour la méthode de Crispell et al. [33]. Le graphique de droite (b) compare la robustesse de correction, en fonction de l'erreur initiale associée à la position fournie en entrée, pour l'algorithme d'asservissement visuel restreint et la méthode de Crispell et al. [33].*

Erreur de reprojection D'autre part, nous avons également comparé les algorithmes d'interpolation de poses, d'asservissement visuel restreint (approche précise) et la méthode de Crispell et al. [33], en analysant l'erreur de reprojection commise par chacun d'eux. La figure 6.6 présente deux graphiques relatifs à ce critère d'évaluation.

En premier lieu, le graphique 6.6a compare les histogrammes des erreurs de reprojection, sur les images de la vidéo considérée et pour un ensemble de points de contrôle répartis dans la scène. Ces histogrammes montrent que l'algorithme d'interpolation est associé à une excellente précision, puisqu'il commet une erreur de reprojection moyenne de 0.2 pixels. Ceci peut être expliqué par le fait que le principe de cet algorithme est justement de minimiser cette erreur de reprojection. La précision associée à l'algorithme d'asservissement visuel dans le cas restreint est un peu plus faible, puisque l'erreur de reprojection moyenne est d'environ 0.9 pixels. En revanche, l'erreur de reprojection moyenne atteint 4.6 pixels dans le cas de la méthode de Crispell et al. [33], ce qui correspond à une précision nettement plus faible que celles obtenues à l'aide de nos algorithmes. Cette mauvaise précision peut être expliquée par le fait que la matrice Jacobienne utilisée dans l'étape de correction correspond à une approximation du lien entre le mouvement 3D de la caméra et la modification de la matrice de recalage entre images. En effet, cette approximation a pour conséquence de rendre plus difficile l'estimation précise du décalage à appliquer aux paramètres d'acquisition, étant donné le recalage observé entre les images. Ainsi, bien que cette méthode permette une estimation satisfaisante de la trajectoire globale, elle peine à affiner cette estimation pour atteindre une faible erreur de reprojection.

En second lieu, le graphique 6.6b analyse la robustesse de correction obtenue avec notre algorithme d'asservissement visuel. En effet, cette méthode requiert une valeur initiale pour effectuer l'estimation des paramètres d'acquisition de l'image considérée. En pratique, nous utilisons simplement les paramètres d'acquisition estimés pour l'image précédente, qui sont ensuite affinés par l'étape de prédiction de notre algorithme puis corrigés par l'étape de correction. Notons que, dans le cas de la méthode de Crispell et al. [33], un filtre de Kalman est utilisé pour prédire directement les paramètres d'acquisition attendus pour l'image courante, en fonction de ceux estimés pour l'image précédente. Cette prédiction, qui est en pratique plus ou moins précise, est ensuite directement utilisée par l'étape de correction pour affiner l'estimation, ce qui peut donc mener à des imprécisions.

Pour analyser la robustesse de notre méthode par rapport à la valeur initiale utilisée, nous avons généré des perturbations aléatoires composées d'une translation d'amplitude pouvant al-

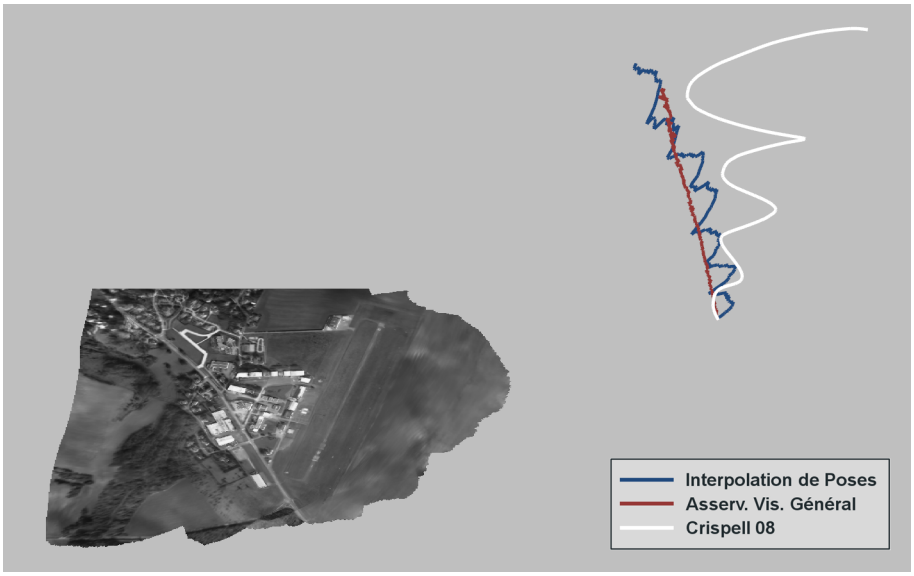


FIGURE 6.7 – *Cette figure présente les trajectoires estimées sur la vidéo* Aérodrome 1*, pour laquelle la vérité-terrain est inconnue, à l'aide des algorithmes d'interpolation de poses, d'asservissement visuel général et de la méthode de Crispell et al. [33].*

ler jusqu'à 50m et d'une rotation d'angle pouvant aller jusqu'à 45° autour d'un axe aléatoire. Nous avons ensuite généré des valeurs initiales bruitées en appliquant ces perturbations aléatoires aux paramètres d'acquisition issus de la vérité-terrain. Nous avons enfin utilisé ces valeurs initiales bruitées, à la place de celle issues de l'estimation à l'image précédente, pour effectuer l'estimation des paramètres d'acquisition de l'image courante. Le graphique 6.6b, dont le nuage de points permet de visualiser l'erreur de reprojection associée à la pose finale estimée en fonction de l'erreur de reprojection associée à la pose initiale bruitée, montre une très bonne robustesse de l'algorithme. En effet, l'erreur de reprojection finale est systématiquement ramenée en dessous de 2.5 pixels, pour des erreurs initiales allant jusqu'à plus de 1000 pixels. Le graphique 6.6b compare ces valeurs avec celles obtenues par la méthode de Crispell et al. [33], qui, même en l'absence de perturbations (erreurs de reprojection initiales ici comprises entre 12 et 22 pixels), génère des erreurs de reprojection finales allant jusqu'à plus de 10 pixels.

Données réelles Enfin, nous avons également testé nos algorithmes de géo-localisation dans le cas d'une vidéo aérienne réelle, dont la trajectoire réelle est inconnue. Pour cela, la figure 6.7 compare les trajectoires estimées sur la vidéo *Aérodrome 1* à l'aide des algorithmes d'interpolation de poses, d'asservissement visuel général et à l'aide de la méthode de Crispell et al. [33]. Ces résultats montrent une bonne cohérence des trajectoires estimées par nos algorithmes d'interpolation et d'asservissement visuel. De plus, contrairement à la trajectoire estimée par l'algorithme d'interpolation de pose, qui est irrégulière, celle obtenue par asservissement visuel présente une meilleure régularité. En revanche, la trajectoire estimée par la méthode de Crispell et al. [33] diverge au bout d'une vingtaine d'images. Cette divergence peut être expliquée par le fait que la vidéo *Aérodrome 1* a été obtenue à l'aide d'une acquisition stabilisée. Cette stabilisation a pour effet de faire varier de manière imprévisible les paramètres de calibration des caméras, rendant ainsi nécessaire l'estimation de leur évolution au fil de la vidéo. Par conséquent, il est difficile pour les méthodes qui ne permettent pas l'estimation de ces paramètres de calibration, de parvenir à estimer correctement la trajectoire associée à une vidéo stabilisée, telle que la vidéo *Aérodrome 1*.

D'autre part, nous avons analysé l'impact du choix de la méthode de géo-localisation sur les performances en détection de changements. La figure 6.8 compare les courbes ROC, obtenues sur la vidéo *Aérodrome 1*, associées aux méthodes d'interpolation de pose, d'asservissement

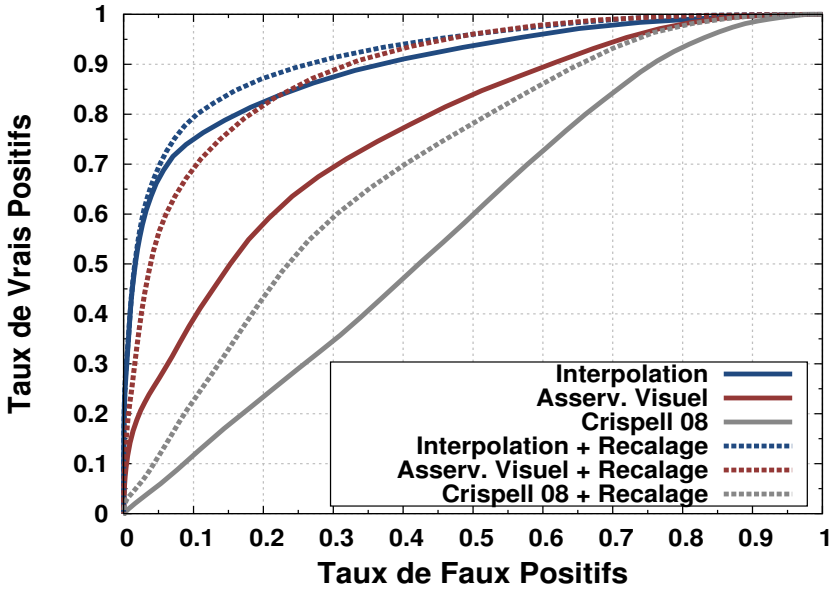


FIGURE 6.8 – *Cette figure compare les courbes ROC associées aux performances de détection de changements obtenues sur la vidéo* Aérodrome 1*, en fonction de la méthode de géo-localisation employée : méthode d'interpolation de poses, d'asservissement visuel général ou méthode de Crispell et al. [33]. La détection est effectuée soit en utilisant directement les paramètres d'acquisition estimés (courbes avec trait plein), soit en effectuant un recalage préalable entre l'image courante et le modèle 3D (courbes avec pointillés) afin de réduire l'erreur de reprojection.*

visuel général et à la méthode de Crispell et al. [33]. Cette figure permet également de comparer les performances obtenues en utilisant directement les paramètres d'acquisition estimés par ces méthodes, avec les performances obtenues en effectuant, préalablement à la détection de changements, un simple recalage affine entre l'image courante et le modèle 3D. Ce recalage préalable a pour but de réduire au maximum l'erreur de reprojection avant de procéder à la détection de changements.

Dans le cas des courbes de la figure 6.8 obtenues sans recalage préalable, les résultats montrent un écart considérable de performances en détection de changements, selon la technique utilisée pour la géo-localisation. D'une part, lorsque la géo-localisation est obtenue par la méthode de Crispell et al. [33], les performances en détection de changements sont mauvaises, ce qui était prévisible au vu des importantes erreurs de reprojections obtenues avec cette technique. D'autre part, malgré la similarité des trajectoires, les performances obtenues à l'aide de la géo-localisation par asservissement visuel général sont nettement inférieures à celles obtenues à l'aide de la géo-localisation par interpolation de poses. Ceci peut être expliqué par le fait que, comme évoqué plus haut, les performances en détection de changements sont fortement liées à l'erreur de reprojection commise par la géo-localisation. Or, comme le montre l'histogramme de la figure 6.6a, la méthode d'asservissement visuel est associée à des erreurs de reprojection plus élevées que celles obtenues par la méthode d'interpolation de poses. Ceci se traduit par de moins bonnes performances en détection de changements, car la mise en correspondance entre les modèles d'apparence et les pixels des images de test est alors effectuée de manière imprécise.

Par ailleurs, les courbes de la figure 6.8 obtenues avec recalage préalable montrent toutes une amélioration des performances par rapport à l'utilisation directe des paramètres d'acquisition estimés. De plus, ces résultats montrent que l'écart de performances, selon que l'on utilise une géo-localisation par interpolation de poses ou par asservissement visuel, est considérablement réduit. Ceci peut s'expliquer par le fait que la méthode d'asservissement visuel effectue une mise en correspondance, qui est certes approchée et associée à des erreurs de reprojection relativement élevées, mais qui est suffisamment précise pour être corrigée par un simple re-

calage affine préalablement à la détection de changement. En revanche, ce n'est pas le cas de la mise en correspondance effectuée par la méthode de Crispell et al. [33], pour laquelle les performances obtenues avec recalage préalable restent inutilisables en pratique.

Pour finir, notons que la qualité du modèle 3D d'apparence, qui est utilisé de manière intensive par l'algorithme d'asservissement visuel, joue un rôle important dans la précision de la trajectoire estimée par cet algorithme. En particulier, l'estimation de la transformation de recalage entre le rendu du modèle 3D et l'image réelle, qui intervient dans l'étape de correction, peut être imprécise si ce modèle n'est pas suffisamment précis ou si les apparences utilisées pour le rendu (e.g. moyenne des observations de référence) ne correspondent pas à celles observées dans l'image réelle (e.g. variations d'illumination, variations d'apparence extrêmes, etc).

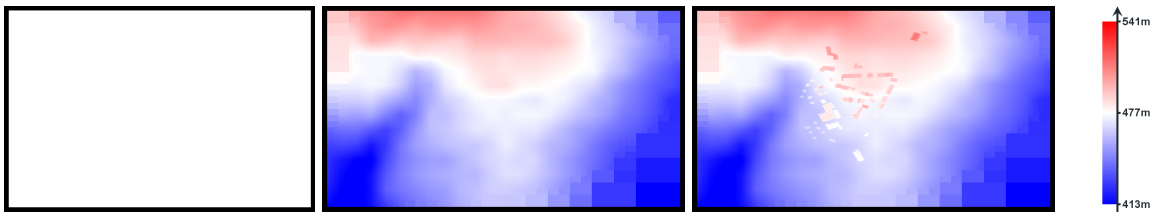
Il resterait de nombreux aspects à analyser concernant ces méthodes de géo-localisation. En particulier, concernant l'algorithme d'asservissement visuel, il pourrait être intéressant d'analyser la robustesse par rapport à l'incertitude sur le plan dominant dans la scène observée, ou de tester différentes techniques de recalage pour améliorer la qualité et la stabilité de l'alignement avec le modèle 3D d'apparence. Cependant, la géo-localisation étant une problématique secondaire par rapport au problème de la détection de changements abordé dans le cadre de cette thèse, nous avons stoppé ici nos évaluations concernant les méthodes associées.

Dans la suite de ce chapitre, les évaluations sont effectuées en utilisant les trajectoires estimées par l'algorithme d'interpolation de poses, à la fois pour la génération du modèle 3D d'apparence et pour la détection de changements.

6.2.2 Influence de la précision des élévations

Un des principaux inconvénients liés à l'approche tri-dimensionnelle pour la détection de changements est que la performance finale peut dépendre de la qualité du modèle 3D utilisé, qui est généralement difficile à estimer en pratique. Par conséquent, pour démontrer l'intérêt d'une telle approche, il est essentiel d'évaluer l'impact des imprécisions sur la connaissance du modèle 3D. Pour cela, et afin de ne pas dériver sur la vaste problématique de la reconstruction 3D, nous avons utilisé différentes hypothèses a priori sur les élévations de la scène observée. La figure 6.9 illustre ces différentes hypothèses, qui correspondent en premier lieu à un sol plan à l'altitude constante de 477m (figure 6.9a), en second lieu à l'information disponible dans le MNT issu des données SRTM3 (figure 6.9b) et enfin à un MNE partiel (figure 6.9c), dérivé du MNT précédent et contenant en plus des annotations manuelles des bâtiments présents dans la scène. Les élévations correspondant à ces hypothèses a priori ont ensuite été utilisées pour générer un modèle 3D d'apparence à partir de la vidéo *Aérodrome 2*, tel que décrit à la section 4.2. Notons que les différentes hypothèses d'élévation débouchent sur les même temps d'exécution, car la génération du modèle 3D d'apparence effectue les mêmes traitements dans les trois cas. Enfin, les modèles 3D d'apparence obtenus ont été utilisés pour détecter les changements sur les vidéos *Aérodrome 1* et *Aérodrome 3*, dans lesquelles des changements ont été insérés à des emplacements stratégiques pour comparer les différentes hypothèses d'élévation (e.g. près de bâtiments, aux endroits où l'altitude est différente de l'altitude moyenne, etc).

La figure 6.10 présente les courbes ROC associées aux performances de détection de changements dans les trois cas mentionnés ci-dessus. Ces courbes montrent que le fait d'utiliser l'hypothèse de sol plan débouche sur des performances sensiblement inférieures aux deux autres cas. D'autre part, nous pouvons voir que les courbes associées à l'utilisation d'un MNT ou d'un MNE sont quasiment identiques. Ce résultat peut laisser penser que des erreurs d'élévation modérées, issues des estimations erronées d'un algorithme de reconstruction 3D, influenceraient peu les performances de détection de changements. Ce point peut être expliqué par l'utilisation d'une structure de carte d'élévation pour la représentation de l'information d'élévation,



(a) *Plan* **(b)** *MNT* **(c)** *MNE* **(d)**

FIGURE 6.9 – *Cette figure illustre les différents a priori sur les élévations dans la scène utilisés pour évaluer l'impact de la précision du modèle 3D sur les performances en détection de changements. Les hypothèses utilisées sont des élévations planes (a), issues d'un modèle numérique de terrain (b) ou d'un modèle numérique d'élévation (c). L'échelle de couleurs utilisée pour représenter les élévations est rappelée en (d).*

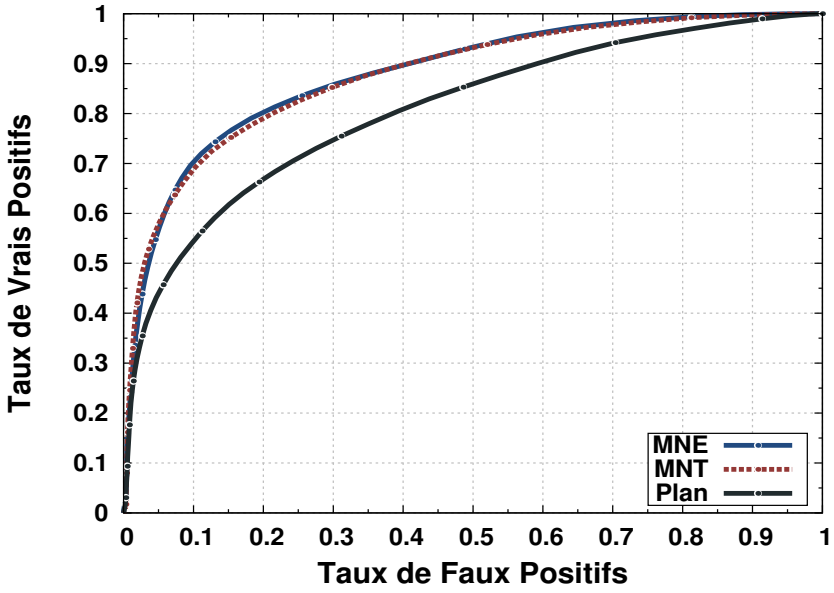


FIGURE 6.10 – *Cette figure compare les courbes ROC associées aux performances de détection de changements obtenues pour différentes hypothèses sur les élévations.*

qui est un type de modèle 3D fortement contraint. Ceci permet une bonne stabilité en présence d'erreurs de lancer de rayon, qui débouche sur une bonne robustesse de représentation.

Plus généralement, cette forte similarité des résultats lors de l'utilisation d'un MNT ou d'un MNE montre que, dans les régions faiblement urbaines où la hauteur des bâtiments est faible devant la distance d'observation, l'exploitation d'un simple MNT suffit à obtenir de bonnes performances de détection de changements. Ce constat est intéressant, car les données MNT sont aujourd'hui disponibles gratuitement pour l'ensemble des régions du monde, et peuvent donc être obtenues beaucoup plus facilement que les données MNE.

Pour nuancer ce constat, notons cependant que l'utilisation de données d'élévation plus précises, telles que celles issues d'un MNE ou d'une reconstruction 3D, peut être nécessaire hors du cadre de la détection de changements. C'est en particulier le cas pour l'algorithme d'asservissement visuel qui, pour fonctionner correctement, nécessite de disposer d'un modèle 3D d'apparence correspondant le mieux possible à la scène observée.

De plus, les données MNE étant disponibles, ce sont celles-ci que nous avons utilisées pour effectuer les évaluations présentées dans la suite de ce chapitre.

6.2.3 Représentations invariantes à l'illumination

Nous avons évoqué au chapitre 2 les difficultés liées aux faux positifs générés par les effets de l'illumination. L'approche de détection de changements proposée dans le cadre de cette thèse utilise plusieurs mécanismes afin de filtrer un maximum de ces faux positifs, le premier étant la conversion des observations dans une représentation invariante à l'illumination. Nous avons donc évalué quantitativement l'impact de ces différentes représentations sur les performances de détection de changements.

Pour cela, nous avons effectué en pré-traitement la conversion des trois vidéos considérées, en utilisant tour à tour les trois types de coordonnées chromatiques et les quatre représentations décrites à la section 3.2. Dans chacun des douze cas résultants, nous avons généré le modèle 3D d'apparence à l'aide de la vidéo *Aérodrome 2* et nous avons évalué les performances de détection de changements sur la vidéo *Aérodrome 1*, qui a été acquise sous des conditions d'illumination différentes de celles de la vidéo *Aérodrome 2*. Les performances de détection de changements ont été analysées en effectuant une consolidation par lissage temporel.

La figure 6.11 présente les courbes ROC obtenues en combinant les différents types de coordonnées chromatiques, c'est-à-dire les coordonnées chromatiques classiques, logarithmiques et $L1L2L3$, avec les différentes représentations, c'est-à-dire les représentations brute, normalisée, désaturée et compensée (voir les définitions à la section 3.2). Les courbes sont tracées avec une couleur commune lorsqu'elles correspondent à des coordonnées chromatiques de même type, et avec un même type de point lorsqu'elles correspondent à une même représentation.

De manière générale, l'objectif des algorithmes d'atténuation de l'illumination est d'obtenir une représentation des observations qui soit invariante par rapport aux effets de l'illumination mais qui permette malgré tout de détecter les changements significatifs. Cependant, ces deux contraintes sont souvent contradictoires en pratique, ce qui débouche sur un compromis à trouver entre la minimisation des faux positifs et la minimisation des faux négatifs. Ce compromis est bien illustré par la figure 6.11, qui montre que pour des valeurs du taux de faux positifs supérieures à 15%, les meilleurs taux de vrais positifs sont obtenus sans atténuation de l'illumination. Ainsi, lorsque la présence de fausses alarmes est tolérée, le meilleur moyen de parvenir à détecter la plupart des changements consiste à travailler avec les observations brutes, c'est-à-dire sans atténuation de l'illumination.

Néanmoins, la réduction du nombre de faux positifs est généralement souhaitable, en particulier pour un système semi-automatique d'analyse vidéo, qui est censé alléger la charge de travail de l'opérateur. Pour cela, les algorithmes présentés à la section 3.2 permettent de limi-

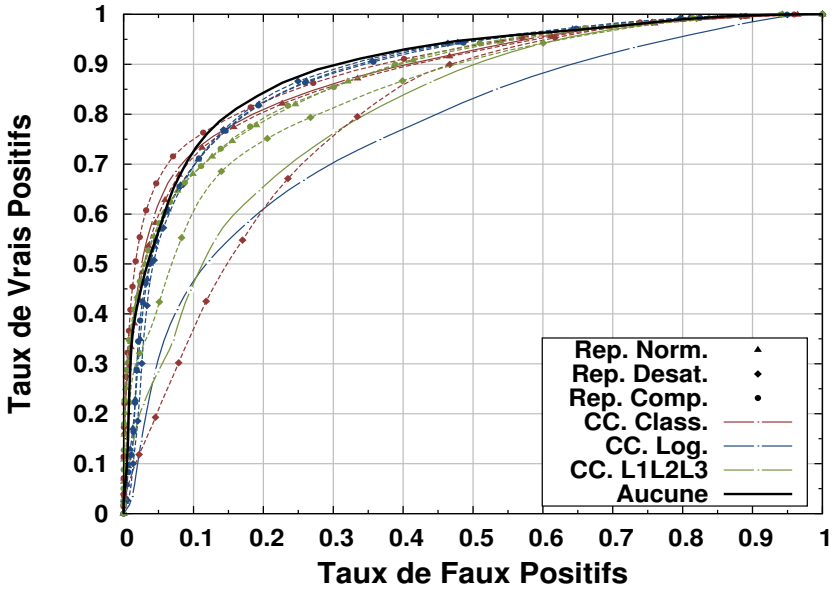


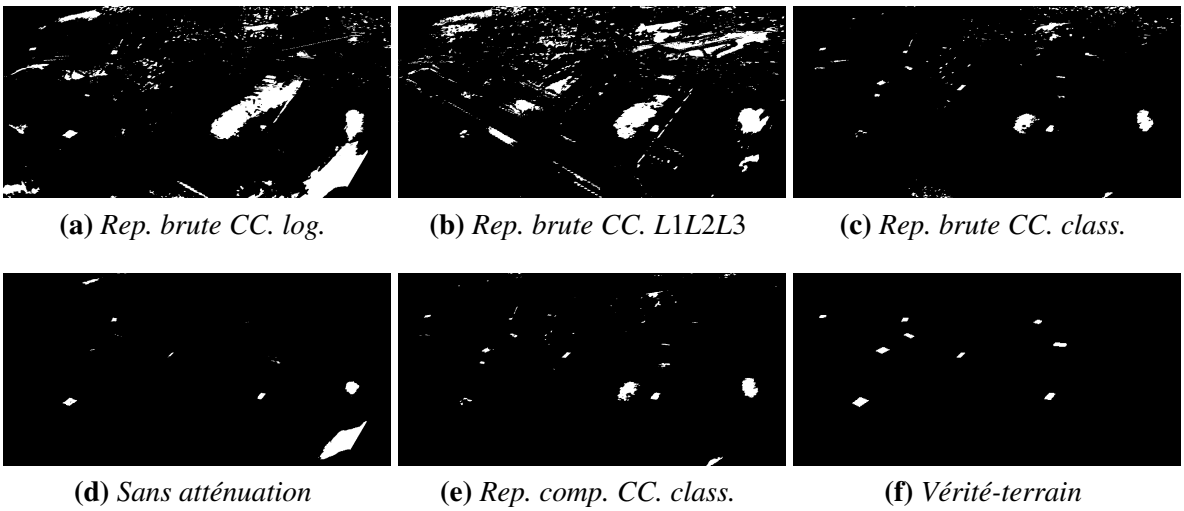
FIGURE 6.11 – *Cette figure compare les performances obtenues sur la vidéo* Aérodrome 1*, à l'aide des différents algorithmes d'atténuation de l'illumination : coordonnées chromatiques (CC.) classiques, logarithmiques et L1L2L3, selon les représentations brute, normalisée, désaturée et compensée.*

ter les faux positifs dûs aux effets de l'illumination, au prix d'une réduction du taux de vrais positifs.

La figure 6.11 permet de comparer les performances de ces algorithmes. Ainsi, la technique permettant d'obtenir les meilleurs résultats pour des valeurs du taux de faux positifs inférieures à 15% est celle utilisant la représentation compensée basée sur les coordonnées chromatiques classiques. Cette technique donne des résultats légèrement meilleurs qu'avec la représentation brute, ce qui peut s'expliquer par le pouvoir discriminatif plus élevé de l'espace de couleur *RGB* classique. Comme mentionné à la section 3.2.1, les performances obtenues avec la représentation normalisée associée aux coordonnées chromatiques classiques sont moins bonnes qu'avec la représentation brute, du fait de nombreux faux positifs générés dans les zones sombres. D'autre part, la représentation désaturée, qu'elle soit associée aux coordonnées chromatiques classiques, logarithmiques ou *L*1*L*2*L*3, donne des performances inférieures aux autres représentations car elle réintroduit dans les observations la variabilité due aux effets de l'illumination, ce qui a pour conséquence de générer de nombreux faux positifs supplémentaires.

Dans le cas des coordonnées chromatiques logarithmiques et *L*1*L*2*L*3, les représentations brutes donnent de mauvais résultats. Ceci peut s'expliquer par le fait que les représentations brutes associées à ces deux types de coordonnées chromatiques sont très instables et pas assez discriminantes, ce qui est confirmé par les illustrations de la section 3.2. Pour ces deux types de coordonnées chromatiques, la reconversion des observations dans l'espace de couleurs *RGB*, à l'aide des représentations normalisée, désaturée ou compensée, permet une nette amélioration des performances, sans toutefois permettre d'atteindre celles associées à une détection de changements sans atténuation de l'illumination.

La figure 6.12 présente une comparaison visuelle des résultats obtenus sans atténuation de l'illumination et avec différentes techniques d'atténuation, dont celles utilisant la représentation brute associée aux coordonnées chromatiques classiques, logarithmiques et *L*1*L*2*L*3, et celle utilisant la représentation compensée associée aux coordonnées chromatiques classiques. Les résultats obtenus à l'aide des deux techniques basées sur la représentation brute associée aux coordonnées chromatiques logarithmiques et *L*1*L*2*L*3 sont considérablement inférieurs à ceux obtenus sans atténuation de l'illumination, du fait de la présence de nombreux faux positifs. En revanche, l'utilisation des coordonnées chromatiques classiques permet de réduire le nombre



(a) *Rep. brute CC. log.* **(b)** *Rep. brute CC. L1L2L3* **(c)** *Rep. brute CC. class.*

(d) *Sans atténuation* **(e)** *Rep. comp. CC. class.* **(f)** *Vérité-terrain*

FIGURE 6.12 – *Cette figure permet une comparaison visuelle des résultats typiques obtenus en fonction de différents algorithmes d'atténuation de l'illumination, ici obtenus sur l'image* 100 *de la vidéo* Aérodrome 1.

Temps (ms)	Rep. brute	Rep. norm.	Rep. désat.	Rep. comp.
Aucune	30.5	-	-	-
CC. class.	35.2	37.0	38.1	37.7
CC. log.	128.9	134.6	135.7	137.1
CC. *L1L2L3*	36.9	45.7	46.3	49.3

TABLE 6.2 – *Temps de calcul moyens par image, exprimés en millisecondes, associés aux techniques de représentation invariante à l'illumination, et obtenus sur la vidéo* Aérodrome 1*, dont les images contiennent* 1280×720 *pixels.*

de faux positifs par rapport aux résultats sans atténuation de l'illumination, la représentation compensée étant plus performante que la représentation brute.

Enfin, la table 6.2 présente les temps de calcul moyens par image associés aux différentes techniques d'atténuation de l'illumination. En plus de permettre les meilleures performances en détection de changements, cette table montre que les techniques de représentation basées sur les coordonnées chromatiques classiques sont également les plus rapides à utiliser. Au contraire, celles basées sur les coordonnées chromatiques logarithmiques sont environ quatre fois plus lentes, du fait de l'utilisation de fonctions exponentielles et logarithmiques dont l'évaluation est relativement longue. À titre de comparaison, cette table présente également le temps de calcul obtenu lorsqu'aucune atténuation de l'illumination n'est effectuée. Notons que ce dernier est non nul, car l'implémentation utilisée effectue, en même temps que l'atténuation de l'illumination, des traitements secondaires mais nécessaires, qui doivent être exécutés même lorsqu'aucune atténuation de l'illumination n'est exigée.

Dans la suite de ce chapitre, sauf mention contraire, les résultats présentés sont obtenus en utilisant l'atténuation de l'illumination correspondant à la représentation compensée à base de coordonnées chromatiques classiques.

6.3 Algorithmes de modélisation d'apparence

L'apparence d'un objet observé dans une vidéo est rarement constante, en particulier dans les vidéos aériennes, où de nombreuses perturbations peuvent survenir. Pour illustrer ce point, la figure 6.13 présente un ensemble de vignettes correspondant au même point physique et

(a) *Vignettes* **(b)** *Emplacement du point observé* **(c)** *Schéma*

FIGURE 6.13 – *Cette figure illustre le problème des variations d'apparence (e.g. dues au passage de nuages, aux effets de l'illumination, aux angles de vue, etc) à l'aide d'un ensemble de vignettes (a) correspondant à une zone fixe (b) de la scène observée dans la vidéo* Aérodrome 2 *(carré rouge). Dans le cas de l'utilisation d'un algorithme de modélisation pixélique, ces variations d'apparence peuvent générer des non-détections en présence de certains types de changements, tels que celui illustré par le schéma (c).*

extraites de la vidéo *Aérodrome 2*. Ces vignettes montrent que l'apparence du point observé peut varier énormément, même au sein d'une seule vidéo. Pour aborder ce problème, nous avons vu au chapitre 2 qu'il est nécessaire d'effectuer une modélisation d'apparence pour les objets de la scène observée. Cependant, dans le cadre de la détection de changements, tous les algorithmes de modélisation d'apparence ne sont pas équivalents.

Pour l'évaluation des différents algorithmes de modélisation d'apparence présentés à la section 4.3, nous avons utilisé la vidéo *Aérodrome 2* pour générer un modèle 3D d'apparence à l'aide de chaque algorithme. Nous avons ensuite effectué une détection de changements sur les vidéos *Aérodrome 1* et *Aérodrome 3*, en effectuant une atténuation de l'illumination ainsi qu'un lissage temporel des résultats (voir la section 5.1.1). Nous avons enfin comparé l'ensemble des détections finales à la vérité-terrain, afin de calculer les courbes ROC pour chaque algorithme de modélisation.

Comparaison des algorithmes La figure 6.14 compare les courbes ROC obtenues à l'aide de différents algorithmes de modélisation d'apparence. Cette figure présente les performances associées aux trois algorithmes présentés à la section 4.3 : les deux algorithmes de modélisation pixéliques, par gaussienne unique et par mélange de gaussiennes (notamment utilisé pour la détection de changements par [34, 91]), ainsi que l'algorithme de modélisation contextuelle par analyse en composantes principales (ACP) incrémentale. D'autre part, les performances associées à deux autres algorithmes de modélisation contextuelle sont également présentées pour comparaison. Le premier algorithme effectue une modélisation par décomposition incrémentale en valeurs singulières (SVD incrémentale), tel que proposé par Brand [21], qui formule la mise à jour incrémentale de la matrice de covariance à l'aide de rotations appliquées aux vecteurs singuliers (voir [21]). Cet algorithme a également fait l'objet d'une adaptation similaire à celle effectuée pour l'ACP incrémentale, afin d'éliminer l'oubli progressif mis en œuvre dans la technique d'origine et de prendre en compte les possibles données manquantes dues aux effets géométriques. Le second algorithme effectue une modélisation que nous désignerons par ACP par blocs, qui exploite une ACP exacte et non plus incrémentale. Pour cela, il est nécessaire de définir des blocs contenant quelques modèles d'apparence, sur chacun desquels une ACP exacte est calculée de manière indépendante. En pratique, ces blocs sont définis en exploitant la structure arborescente du Quad-Tree augmenté, en coupant l'arbre un niveau hiérarchique au dessus du niveau de la branche la plus profonde. Il découle de cette définition que chacun des

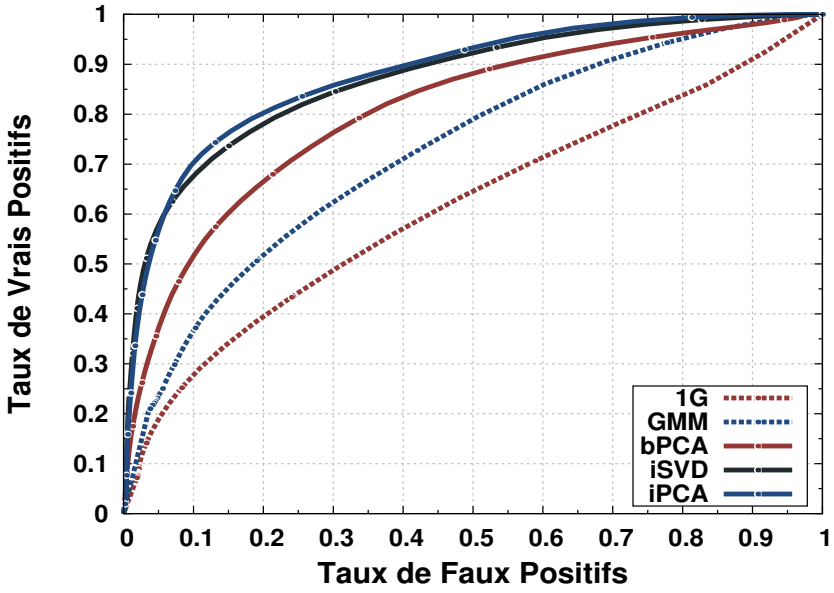


FIGURE 6.14 – *Cette figure compare les performances moyennes obtenues sur les vidéos* Aérodrome 1 *et* Aérodrome 3*, à l'aide de différents algorithmes de modélisation d'apparence. Les algorithmes utilisés effectuent une modélisation par gaussienne unique (1G), par mélange de gaussiennes (GMM), par ACP incrémentale (iPCA), par SVD incrémentale (iSVD) et par ACP par blocs (bPCA).*

blocs contient entre quatre ou seize cellules de Quad-Tree, contenant chacune un modèle d'apparence [1]. Par ailleurs, les trois algorithmes de modélisation contextuelle utilisent un nombre identique de composantes principales (ou de vecteurs singuliers pour la modélisation par SVD incrémentale), ici $N_{\text{CP}} = 20$.

La figure 6.14 montre que l'algorithme de modélisation par gaussienne unique donne les moins bonnes performances, ce qui est conforme aux attentes puisque cette méthode n'exploite pas l'information contextuelle et ne permet pas non plus de capturer correctement le comportement multi-modal des observations (voir la figure 6.13a). La modélisation par mélange de gaussiennes permet de capturer correctement ces comportements multi-modaux, à l'aide des trois gaussiennes de la distribution. En revanche, le fait que cette modélisation soit effectuée de manière indépendante en chaque pixel peut générer des erreurs de détection. Par exemple, un changement tel que celui illustré à la figure 6.13c, qui est composé d'apparences déjà observées dans les vidéos de référence mais qui présente un changement en termes de structure, sera ignoré alors qu'il devrait être détecté comme changement d'intérêt potentiel. Au contraire, ce type de changements structurels sont détectés correctement par les algorithmes exploitant le contexte pour la modélisation des apparences, qui sont donc associées à de meilleures performances. Parmi ces algorithmes, celui exploitant une ACP par blocs donne des performances moindres. Ceci est dû au fait que l'analyse est effectuée sur des blocs indépendants dans l'image, ce qui peut générer des non-détections dans certains cas. En effet, un changement donné est souvent plus facilement détectable au niveau de sa bordure qu'au niveau de son intérieur, qui peut être plus homogène et moins texturé. En particulier, lorsqu'un changement recouvre entièrement un bloc donné, la manifestation de ce changement est généralement imperceptible pour ce bloc, pour lequel aucun changement n'est donc détecté. Par conséquent, l'algorithme d'ACP par bloc peine à détecter les changements complètement et ne détecte le plus souvent que leurs bordures, ce qui mène à un taux de faux négatifs supérieur aux méthodes exploitant les images globales. Enfin, les algorithmes de SVD incrémentale et d'ACP incrémentale donnent des performances

1. Ce nombre relativement faible de cellule par bloc est nécessaire pour limiter l'occupation mémoire, qui est critique avec cette approche, comme le montre la table 6.3.

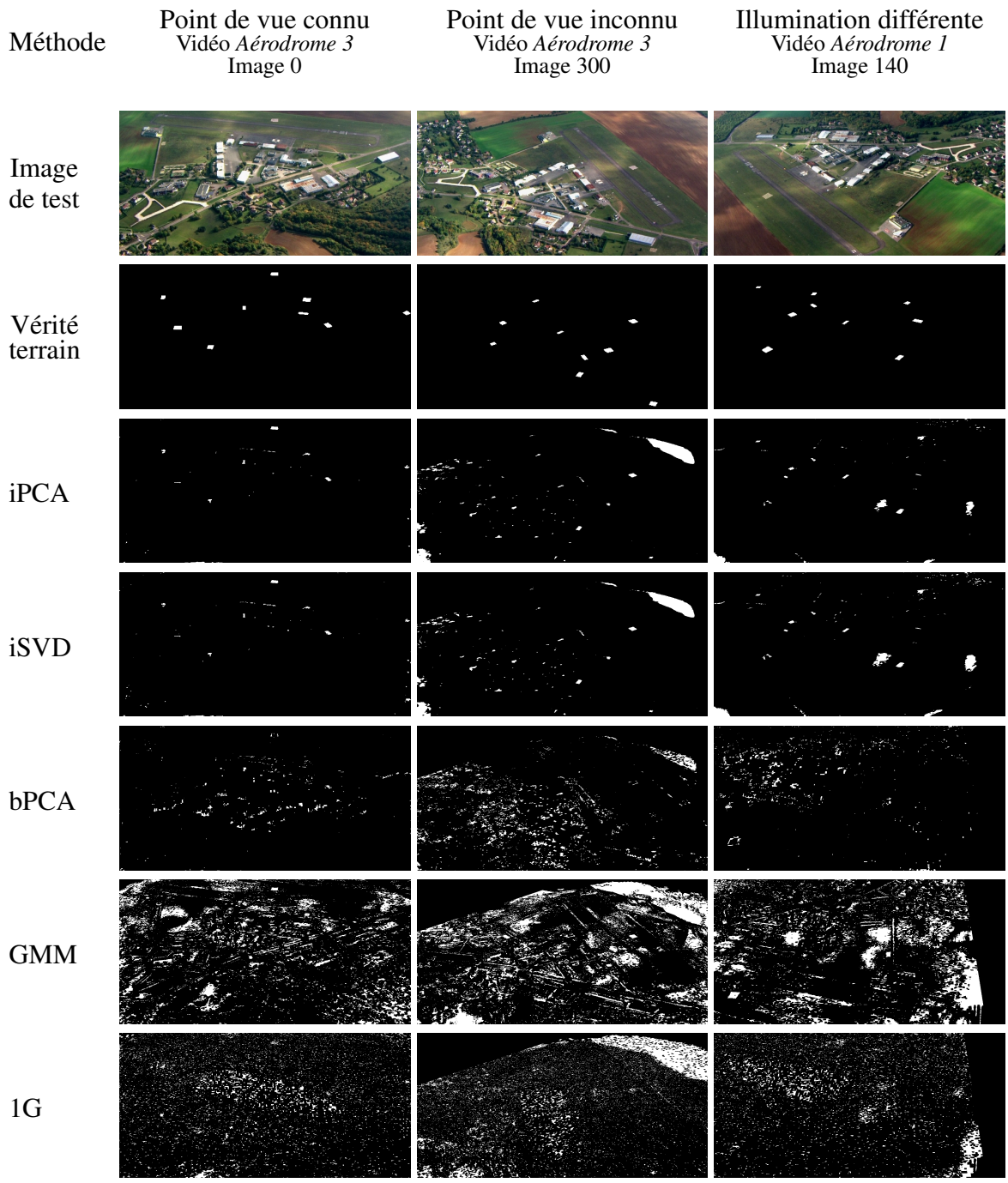


FIGURE 6.15 – *Cette figure permet une comparaison visuelle des résultats obtenus sur les vidéos* Aérodrome 1 *et* Aérodrome 3*, à l'aide de différents algorithmes de modélisation d'apparence. Les algorithmes utilisés effectuent une modélisation par gaussienne unique (1G), par mélange de gaussiennes (GMM), par ACP incrémentale (iPCA), par SVD incrémentale (iSVD) et par ACP par blocs (bPCA).*

équivalentes, avec une légère supériorité de la méthode par ACP incrémentale.

La figure 6.15 présente une comparaison visuelle des résultats bruts, c'est-à-dire sans consolidation temporelle, obtenus à l'aide des différents algorithmes de modélisation d'apparence comparés à la figure 6.14. Les résultats issus des algorithmes effectuant une modélisation pixélique, c'est-à-dire les algorithmes par gaussienne unique et par mélange de gaussiennes, génèrent un grand nombre de faux positifs et ne détectent généralement les changements que partiellement. L'algorithme utilisant une ACP par blocs commet nettement moins de faux positifs que les deux algorithmes pixéliques. Cependant, à cause de l'indépendance des blocs considérés, les changements recouvrant un bloc entier peuvent ne pas être détectés sur ce bloc, menant à un certain nombre de faux négatifs. Parmi les algorithmes comparés, ceux utilisant une modélisation par SVD incrémentale et ACP incrémentale, qui utilisent l'information contextuelle des images totales, sont associées aux meilleures performances. En effet, elles génèrent relativement peu de faux positifs et une bonne proportion des changements sont détectés, même si selon la valeur du seuil utilisé, ils ne sont pas toujours complètement détectés.

Enfin, la table 6.3 compare les performances de ces algorithmes en termes de temps de calcul et d'occupation mémoire du modèle 3D d'apparence. Les mesures de temps de calcul ont été mesurés sur un PC standard de 2.4 *Ghz*. Ces temps de calcul sont en pratique dominés par l'algorithme de lancer de rayon, qui, dans l'implémentation utilisée, a été parallélisé sur quatre cœurs. Malgré la simplicité de ces méthodes, les temps de détection moyens par image sont importants dans le cas des modélisation par gaussienne unique et par mélange de gaussiennes. Ceci peut s'expliquer par l'utilisation intensive faite par ces algorithmes de la fonction exponentielle, dont l'évaluation est relativement longue, ainsi que par une moins bonne utilisation du cache, du fait que les pixels sont traités indépendamment. Dans le cas de la modélisation par ACP incrémentale, cette table montre que l'utilisation d'un nombre plus ou moins important de composantes principales n'influe que de manière négligeable sur les temps de détection. En revanche, ce nombre de composantes principales a une influence plus significative sur les temps de modélisation et sur l'occupation mémoire. Par ailleurs, cette table montre que, dans le cas où $N_{\text{CP}} = 3$, l'occupation mémoire du modèle par ACP incrémentale et celle du modèle par mélange de gaussiennes sont équivalentes. Ceci démontre que la modélisation d'apparence par mélange de gaussiennes donne non seulement de moins bons résultats que l'approche par ACP incrémentale, mais est également associée à des temps de calculs et une occupation mémoire supérieurs. Par ailleurs, l'occupation mémoire associée à la modélisation par ACP par blocs est très importante, du fait du stockage des matrices de covariance réelles [2] au lieu de leur approximation par les quelques premiers vecteurs propres. Enfin, l'approche par SVD incrémentale est associée à des temps de modélisation nettement plus longs que ceux de l'ACP incrémentale.

Dimension du sous-espace de modélisation La précision des algorithmes de modélisation contextuelle, tels que celui basé sur une ACP incrémentale, dépendent de la dimension du sous-espace recherché. En effet, plus ce sous-espace est de dimension importante, plus fine sera la modélisation des observations. Dans le cas de l'algorithme par ACP incrémentale, cette précision de modélisation dépend du nombre N_{CP} de composantes principales conservées durant la génération du modèle 3D d'apparence. La table 6.4 présente l'évolution de l'énergie cumulative $E_{N_{\text{CP}}}$ associée au modèle final en fonction de N_{CP}. Ainsi, ces valeurs montrent qu'en utilisant un nombre de composantes principales de $N_{\text{CP}} = 40$ (sur un maximum de $3 \times 142\,000$ dans le cas de la vidéo *Aérodrome 2*, ce qui est impossible à atteindre en pratique), le modèle final parvient à expliquer environ 50% de la variance totale observée dans la vidéo de référence. La figure 6.16 montre cependant que, pour des valeurs de N_{CP} supérieures à 20, l'amélioration des performances de détection de changements n'est que minime. Notons par ailleurs que l'énergie cumulative E_3 associée aux trois premières composantes principales augmente avec N_{CP}. Cela est dû à l'approximation mise en œuvre à chaque étape de la modélisation incrémentale,

2. Ce stockage tient compte de la symétrie des matrices de covariance.

Algorithme	Temps de modélisation (h)	Temps moyen de détection (s)	Taille du modèle final (Mo)
1G ($N_{modes} = 1$)	1.3	9.0	75
GMM ($N_{modes} = 3$)	2.1	13.0	97
bPCA ($N_{\mathrm{CP}} = 20$)	1.5	4.5	960
iSVD ($N_{\mathrm{CP}} = 20$)	5.2	4.5	176
iPCA			
$N_{CP} = 3$	1.4	4.47	96
$N_{CP} = 5$	1.4	4.48	113
$N_{CP} = 10$	1.6	4.51	158
$N_{CP} = 20$	2.1	4.56	246
$N_{CP} = 30$	2.6	4.61	335
$N_{CP} = 40$	3.3	4.65	423

TABLE 6.3 – *Tailles des modèles finaux, temps de modélisation des données de référence et temps de détection moyens par image de test, associés à chacun des algorithmes de modélisation d'apparence : gaussienne unique (1G), mélange de gaussiennes (GMM), ACP par blocs (bPCA), SVD incrémentale (iSVD) et ACP incrémentale (iPCA) pour différentes valeurs de N_{CP}.*

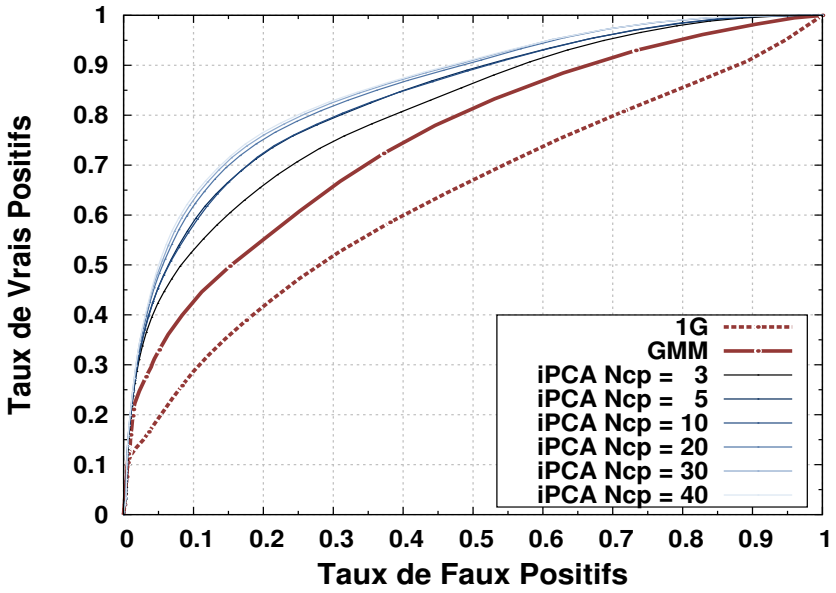


FIGURE 6.16 – *Cette figure compare les performances moyennes obtenues sur les vidéos* Aérodrome 1 *et* Aérodrome 3*, pour plusieurs valeurs du nombre N_{CP} de composantes principales conservées dans l'algorithme de modélisation par ACP incrémentale. À titre de comparaison, le graphique présente également les performances obtenues à l'aide de la modélisation par gaussienne unique (1G) et par mélange de gaussiennes (GMM), qui correspondent respectivement à une modélisation à l'aide de* 1 *et* 3 *modes.*

N_{CP}	3	5	10	20	30	40
E_3	28.69%	29.20%	29.22%	29.23%	29.25%	29.26%
$E_{N_{CP}}$	28.69%	36.88%	43.84%	48.00%	49.59%	50.36%

TABLE 6.4 – *Énergies cumulatives des représentations issues de l'algorithme de modélisation par ACP incrémentale, en fonction du nombre N_{CP} de composantes principales utilisées. E_3 représente l'énergie cumulative associée aux trois premières composantes principales et $E_{N_{CP}}$ représente l'énergie cumulative associée à l'ensemble des composantes principales utilisées.*

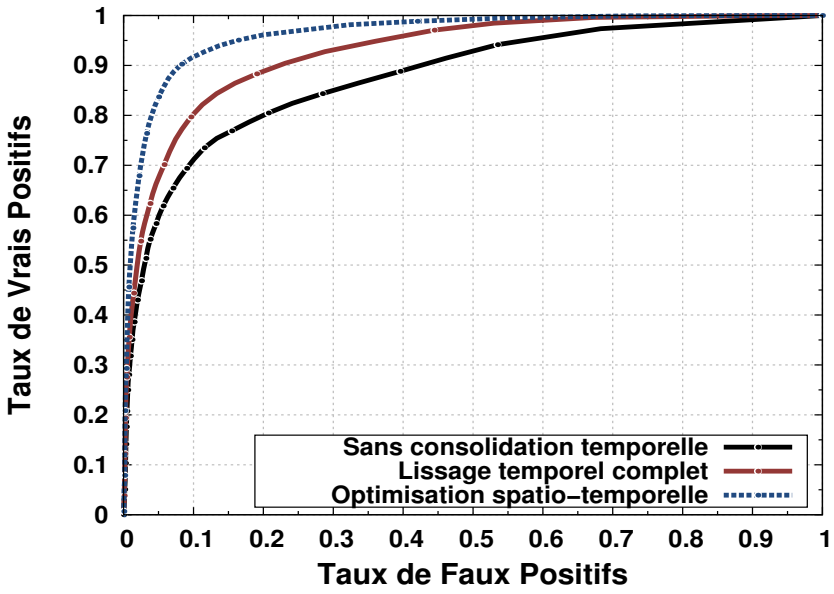


FIGURE 6.17 – *Cette figure compare les courbes ROC associées aux performances de détection de changements obtenues après application des deux techniques de consolidation temporelle.*

au cours desquelles la quantité d'information conservée est d'autant plus importante que N_{CP} est grand. Enfin, la figure 6.16 permet également de vérifier que pour une même valeur de la dimension du sous-espace de modélisation, égale à 3, l'algorithme de modélisation par ACP incrémentale donne bien de meilleurs résultats que celui par mélange de gaussiennes, ce qui confirme l'intérêt d'utiliser une modélisation contextuelle plutôt que pixélique.

6.4 Évaluation des techniques de consolidation

Cette section présente les résultats d'évaluation des méthodes de consolidation présentées au chapitre 5. Les évaluations portent sur les résultats et temps d'exécution intermédiaires lorsqu'applicable, et sur les performances finales en détection de changements.

6.4.1 Influence de la consolidation temporelle

Nous avons présenté à la section 5.1 plusieurs méthodes de consolidation temporelle, permettant l'exploitation en-ligne de la redondance présente dans la vidéo de test. Cette section analyse les résultats d'évaluation de ces méthodes.

Pour commencer, la figure 6.17 compare les courbes ROC associées aux méthodes de lissage temporel complet (voir section 5.1.1) et d'optimisation spatio-temporelle (voir section 5.1.2) avec la courbe ROC obtenue lorsqu'aucune consolidation temporelle n'est effectuée. Cette figure montre clairement que l'exploitation de la redondance dans la vidéo de test, à l'aide de l'une ou l'autre des méthodes, débouche sur un gain de performance. D'autre part, elle montre

Méthode	Facteur de réduction	Temps par image (ms)
Lissage temporel complet	1	1573
Lissage temporel hybride	1	799
Lissage temporel hybride	1/4	93
Optimisation spatio-temporelle	1/4	9908

TABLE 6.5 – *Temps d'exécution moyens par image des algorithmes de consolidation temporelle, exprimés en millisecondes, selon le facteur de sous-échantillonnage appliqué aux dimensions de l'image.*

également que la méthode d'optimisation spatio-temporelle débouche sur des résultats nettement meilleurs que la méthode de lissage temporel. Ceci s'explique par le fait que cette méthode incorpore une modélisation du comportement spatio-temporel des changements, qui est plus précise que le critère très simple de la méthode de lissage temporel.

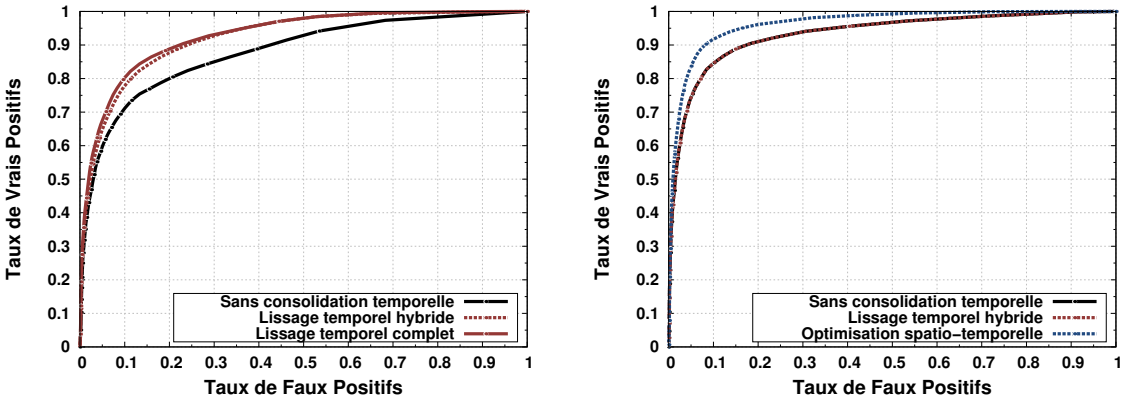
Cependant, cette différence de performance pourrait aussi être expliquée par le fait que les conditions d'exécution de ces deux méthodes sont différentes. En effet, la première travaille en coordonnées images, sur des images de faible résolution et avec une fenêtre temporelle limitée, tandis que la seconde travaille en coordonnées spatiales, sur des images en pleine résolution et avec une fenêtre temporelle infinie. Par conséquent, pour effectuer une comparaison juste entre les méthodes, nous avons introduit la méthode de lissage temporel hybride (voir section 5.1.3), appliquant le même traitement que la méthode de lissage temporel complet mais dans les mêmes conditions que la méthode d'optimisation spatio-temporelle (coordonnées images et fenêtre temporelle limitée).

La figure 6.18a compare les courbes ROC obtenues sur les images en pleine résolution sans consolidation temporelle et avec les méthodes de lissage temporel hybride et complet. Les deux méthodes de lissage temporel donnent de meilleurs résultats que lorsqu'aucune consolidation temporelle n'est effectuée. D'autre part, bien que les courbes des deux méthodes de lissage temporel soient relativement proches, celle associée au lissage temporel complet est légèrement supérieure. Ceci confirme l'intuition selon laquelle la consolidation temporelle est plus performante lorsqu'elle est appliquée en coordonnées spatiales et à l'aide d'une fenêtre temporelle infinie.

La figure 6.18b compare les courbes ROC obtenues sur les images sous-échantillonnées sans consolidation temporelle et avec les méthodes de lissage temporel hybride et d'optimisation spatio-temporelle. De manière générale, le fait de réduire les dimensions de l'image a deux conséquences pratiques sur les courbes ROC. La première est une amélioration globale des performances de détection, puisque les détections associées à de petites zones, qui sont généralement dues au bruit, ont tendance à disparaître. La seconde conséquence est un rapprochement des courbes associées aux différentes méthodes, puisque le nombre total de pixels, et donc la différence de performance potentielle entre deux méthodes, diminue. Cette dernière conséquence, combinée avec la simplicité du critère de la méthode de lissage temporel, permettent d'expliquer le fait que les résultats de la méthode de lissage temporel hybride soient quasiment identiques à ceux obtenus sans consolidation temporelle. En revanche, là encore, la méthode d'optimisation spatio-temporelle est associée à de meilleures performances, ce qui confirme que l'amélioration des résultats est bien due à l'algorithme utilisé.

Plus généralement, ces résultats montrent qu'exploiter la redondance spatio-temporelle dans la vidéo de test permet une amélioration substantielle des performances de détection de changements. Cependant, cette amélioration s'accompagne également d'une augmentation des temps de traitement par image, comme le montre la table 6.5. Ces temps de traitement ont été mesurés sur un PC standard de 2.4 *Ghz* exécutant une implémentation mono-thread.

Pour des raisons de temps de calcul lors de la génération des courbes ROC, les résultats



(a) *Images en pleine résolution* **(b)** *Images sous-échantillonnées*

FIGURE 6.18 – *Cette figure compare les courbes ROC associées aux performances de détection de changements obtenues après application des deux techniques de consolidation temporelle, selon les conditions d'application. Le graphique de gauche (a) présente les courbes pour les méthodes appliquées en pleine résolution et celui de droite (b) présente celles pour les méthodes appliquées aux images sous-échantillonnées.*

d'évaluation présentés dans les autres sections de ce chapitre ont été générés, pour la plupart, en utilisant la consolidation par lissage temporel. L'utilisation d'une consolidation par optimisation spatio-temporelle mène cependant à une amélioration des performances.

6.4.2 Influence de l'algorithme de binarisation

La section 5.2 a montré comment exploiter le fait que les changements d'intérêt correspondent à des structures artificielles, dont les frontières sont bien définies, en utilisant l'algorithme d'extraction de régions extrêmes maximales à stabilité maximale (MSER). Cet algorithme intervient durant l'étape de binarisation du masque de changements, c'est-à-dire pour la conversion de la carte de scores de détection, qui sont à valeurs continues, en un masque de changements binaire. Après l'atténuation de l'illumination par pré-traitement et la capacité d'apprentissage de la modélisation d'apparence par ACP incrémentale, cette technique de binarisation exploitant les MSER constitue un troisième mécanisme permettant de limiter les faux positifs dûs aux effets de l'illumination. En effet, la notion de MSER est pertinente pour distinguer les changements d'intérêt potentiel, dont les frontières sont généralement bien définies, des variations d'illumination, dont les frontières sont généralement plus floues.

Pour évaluer l'influence de cette technique de binarisation, nous avons calculé les courbes ROC associées aux performances de détection de changements obtenues en effectuant l'étape de binarisation grâce à un simple seuillage ou grâce à l'algorithme d'extraction de MSER. Pour cela, nous avons effectué la modélisation d'apparence par ACP incrémentale à l'aide de la vidéo *Aérodrome 2*, puis nous avons détecté les changements sur la vidéo *Aérodrome 1*, avec atténuation de l'illumination et consolidation par un lissage temporel.

Comme pour celles présentées précédemment dans ce chapitre, la courbe ROC correspondant au cas de la binarisation par seuillage a été générée en faisant varier le seuil de détection, afin d'explorer l'ensemble des couples de valeurs du taux de vrais positifs et du taux de faux positifs. Dans le cas de la binarisation par extraction de MSER, la courbe ROC a été générée en faisant varier le paramètre Δ, qui caractérise la localité du taux de variation de l'aire des régions considérées (voir la définition à la section 5.2). Ce paramètre permet bien d'explorer les couples de valeurs des taux d'erreurs. En effet, de petites valeurs de Δ sont associées à un taux de variation de l'aire très localisé, qui est par conséquent peu performant pour différencier les régions aux frontières très nettes de celles aux frontières plus floues. Dans ce cas, le nombre de faux

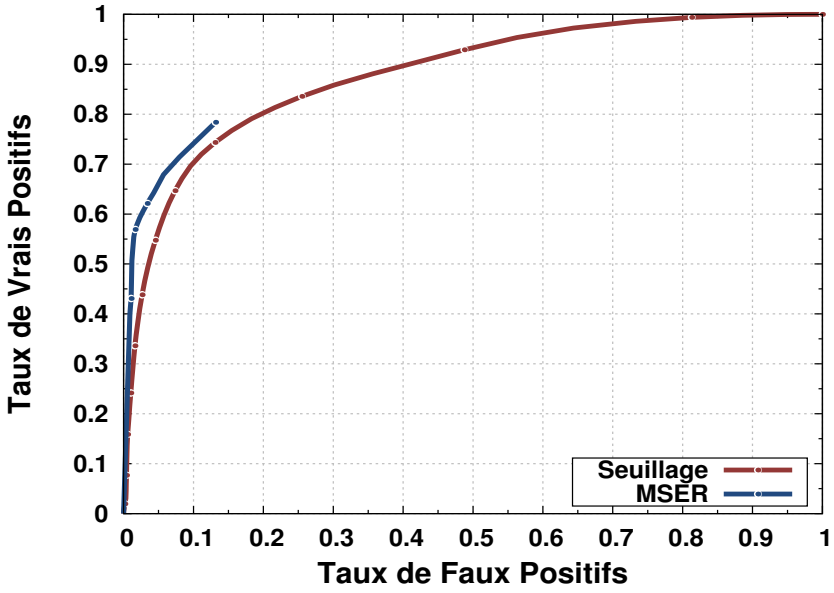


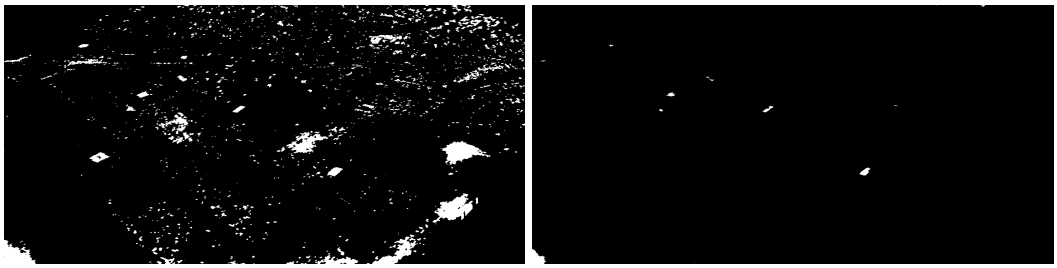
FIGURE 6.19 – *Cette figure compare les courbes ROC associées aux performances de détection de changements obtenues en estimant le masque de changements à l'aide d'un seuillage ou d'une binarisation par extraction de MSER.*

positifs est important mais peu de faux négatifs sont générés. Au contraire, pour des valeurs importantes de Δ, le taux de variation de l'aire est calculé de manière plus globale, permettant ainsi d'effectuer une discrimination plus stricte des régions, ce qui débouche sur une réduction du nombre de faux positifs mais une augmentation du nombre de faux négatifs. Par ailleurs, l'algorithme d'extraction de MSER nécessitant une quantification des valeurs d'entrée, il n'est pas possible de calculer le taux de variation de l'aire des régions avec des valeurs arbitrairement faibles de Δ. Ceci a pour conséquence qu'il n'est pas possible d'atteindre des valeurs élevées du taux de faux positifs à l'aide de ce paramètre, ce qui mène à la génération de courbes ROC tronquées dans le cas de la binarisation par extraction de MSER. Notons cependant que ceci n'est pas gênant, puisque nous nous intéressons surtout à la partie de la courbe correspondant à des valeurs faibles du taux de faux positifs.

La figure 6.19 compare les performances associées à la binarisation par seuillage et par extraction de MSER. Ces courbes montrent que la mise en œuvre de la binarisation par extraction de MSER se traduit bien par une amélioration des performances, par rapport à la binarisation par simple seuillage. La figure 6.20 illustre cette amélioration en comparant visuellement les masques de changements issus de chacune des deux méthodes. Dans le cas de la binarisation par seuillage, une valeur de seuil trop élevée fait perdre une bonne partie des changements pertinents, tandis qu'une valeur trop faible génère un grand nombre de faux positifs. Il n'est de plus pas possible de trouver une valeur de seuil intermédiaire, qui permettrait de conserver l'ensemble des changements pertinent tout en éliminant les fausses alarmes. Au contraire, la binarisation par extraction de MSER permet d'éliminer une large majorité de ces fausses alarmes tout en conservant l'ensemble des changements de la vérité-terrain. Cette méthode de binarisation donne donc de meilleures performances de détection de changements que la méthode de seuillage. Elle est toutefois associée à des temps de calculs plus élevés, comme le montre la table 6.6.

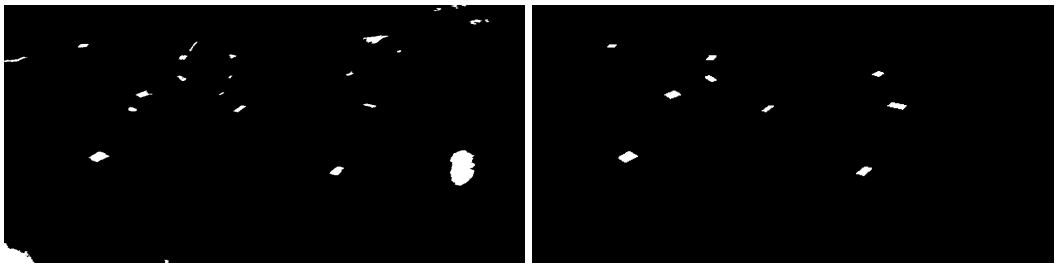
6.4.3 Influence du retour interactif de pertinence

Le mécanisme de retour interactif de pertinence présenté à la section 5.3 permet de consolider les résultats de détection en fonction d'annotations fournies par l'utilisateur. D'autre part, cette technique constitue le dernier mécanisme permettant de filtrer les faux positifs résiduels



Seuil égal à 0.09 Seuil égal à 0.56

(a) *Estimations par seuillage*



Delta égal à 7/255

(b) *Estimation par extraction de MSER* **(c)** *Vérité-terrain*

FIGURE 6.20 – *Cette figure compare, par rapport à la vérité-terrain (c), les masques de changements estimés sur l'image* 140 *de la vidéo* Aérodrome 1 *à l'aide d'une opération de seuillage (a) ou par extraction de MSER (b).*

Méthode de binarisation	Temps par image (ms)
Seuillage	4
Extraction de MSER	417

TABLE 6.6 – *Temps d'exécution moyens par image des techniques de binarisation, exprimés en millisecondes, obtenus sur des images de taille* 1280×720 *pixels*

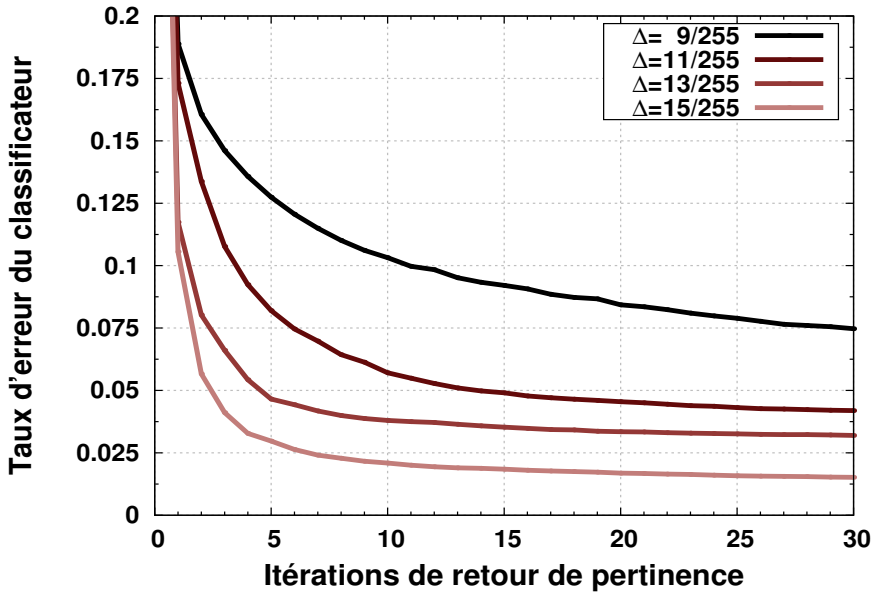


FIGURE 6.21 – *Cette figure présente les courbes d'évolution du taux d'erreurs de classification, c'est-à-dire de la moyenne du taux de pixels non-pertinents classés comme pertinents (taux de faux positifs de classification) et du taux de pixels pertinents classés comme non-pertinents (taux de faux négatifs de classification), en fonction du nombre d'itérations du mécanisme de retour de pertinence.*

résultant non seulement des effets de l'illumination, mais plus généralement de toute variation parasite ne présentant pas un intérêt pour l'analyste image. Ce mécanisme de retour interactif permet en effet d'adapter, dans une certaine mesure, les résultats de détection de changements aux besoins de l'utilisateur.

Performances hors-ligne Afin d'évaluer cette technique, nous avons commencé par mesurer les performances obtenues sans prendre en compte l'aspect en-ligne du mécanisme. Pour cela, nous avons effectué la détection de changements sur la vidéo *Aérodrome 1*, à l'aide de la technique d'atténuation de l'illumination et avec lissage temporel. Les cartes de scores obtenues ont ensuite été converties en masques de changements binaires, à l'aide de la technique de binarisation par extraction de MSER, pour des valeurs du paramètre Δ de $\frac{9}{255}$, $\frac{11}{255}$, $\frac{13}{255}$ et $\frac{15}{255}$ (voir section 5.2). Par la suite, nous avons extrait les descripteurs associés à l'ensemble des régions détectées dans la vidéo *Aérodrome 1* et nous avons analysé la vérité-terrain associée pour affecter une classe binaire (changement ou non-changement) à chacune des régions. Enfin, nous avons simulé une annotation interactive en exploitant ces classes issues de la vérité-terrain. Pour cela, 5% des régions détectées ont été sélectionnées aléatoirement et annotées en 30 itérations d'annotation incrémentale [3].

La figure 6.21 présente l'évolution du taux d'erreur de classification, c'est-à-dire de la moyenne entre le taux de faux positifs de classification et le taux de faux négatifs de classification, en fonction du nombre d'itérations de la simulation. Notons que ce taux d'erreurs de classification permet une analyse des performances de manière conditionnelle à la détection de changements, car la classification ne s'applique qu'aux régions détectées par celle-ci. Or, avant la première itération du mécanisme de retour interactif de pertinence, aucune annotation n'a encore été fournie. Par conséquent, toutes les régions détectées sont classées dans la catégorie des changements, ce qui débouche sur un taux de faux positifs de classification égal à 1 et un taux de faux négatifs de classification égal à 0. À l'itération 0, le taux d'erreurs de classification vaut donc 0.5. Ensuite, les annotations fournies lors de la première itération permettent une nette réduction du taux d'erreurs de classification et les itérations suivantes permettent progressivement de le réduire davantage. Pour les valeurs de Δ de $\frac{11}{255}$, $\frac{13}{255}$ et $\frac{15}{255}$, nous pouvons voir qu'au bout de 30 itérations, le taux d'erreurs de classification atteint une valeur inférieure à 5%, mais que la convergence est quasiment atteinte au bout d'une dizaine d'itérations.

Ces performances satisfaisantes du point de vue du problème de classification débouchent de manière logique sur de bonnes performances du point de vue du problème de la détection de changements, à ceci près que le taux de vrais positifs après retour interactif ne peut être supérieur à celui obtenu sans ce retour interactif. Cette limite, évoquée à la section 5.3.1, vient du fait que pour des raisons d'efficacité, notre mécanisme de retour interactif de pertinence effectue l'analyse des régions détectées par les algorithmes précédents, sans effectuer de nouvelle détection. La figure 6.22 compare les performances obtenues pour quatre points de fonctionnement, correspondant aux quatre valeurs considérées pour le paramètre Δ de la binarisation par extraction de MSER, avec ou sans utilisation du mécanisme de retour interactif de pertinence. Cette comparaison montre que les annotations fournies lors de la première itération permettent d'initialiser le classificateur, ce qui mène à une forte diminution du taux de faux positifs, mais également à une diminution du taux de vrais positifs, qui résulte d'une prédiction incorrecte de la classe de certaines régions correspondant à des changements pertinents. Cependant, les annotations fournies lors des itérations suivantes permettent d'affiner la classification. Ceci mène ainsi à une augmentation progressive du taux de vrais positifs, qui néanmoins ne pourra jamais dépasser celui associé à l'itération 0, tout en continuant de réduire progressivement le taux de faux positifs.

3. Par conséquent, 0.16% des régions ont été annotées à chaque itération.

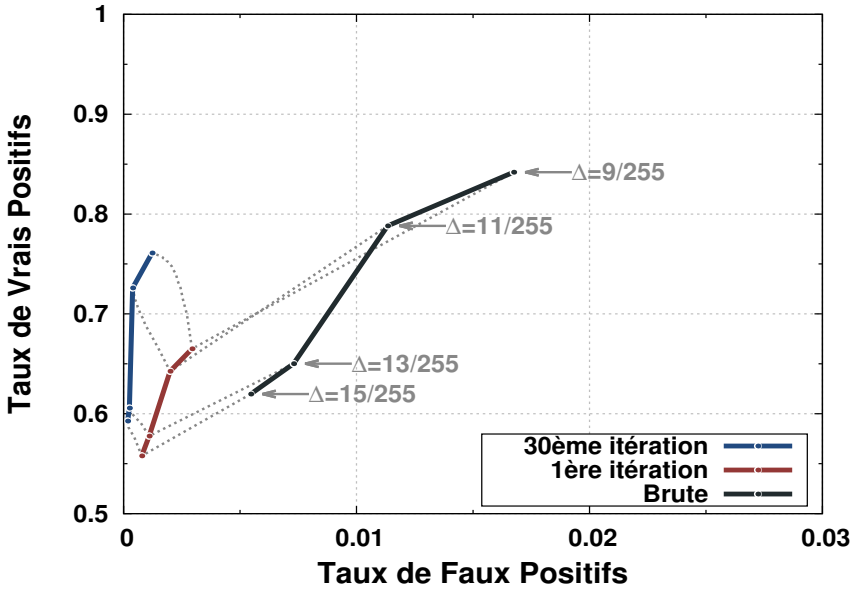


FIGURE 6.22 – *Cette figure compare les performances de détection de changements associées à quelques points de fonctionnement, avec et sans utilisation du mécanisme de retour interactif de pertinence.*

Performances en-ligne En pratique, notre approche de détection de changements vise une détection en ligne des changements dans la vidéo de test, ce qui implique que le mécanisme de retour interactif de pertinence est appliqué de manière incrémentale. Ceci signifie que les annotations saisie par l'utilisateur sur une image donnée ne peuvent pas influencer les résultats obtenus pour les images précédentes de la vidéo considérée, ce qui a nécessairement un impact sur les performances idéales présentées ci-dessus, dans le cas hors-ligne.

Pour évaluer l'impact de cette contrainte sur les performances du mécanisme de retour de pertinence, nous avons comparé les résultats obtenus avec différentes politiques d'annotation. Dans le premier cas, la politique d'annotation a consisté à entraîner le classificateur à l'aide des régions détectées dans les 34 premières images de la vidéo *Aérodrome 1* (qui en contient 214). La seconde politique a visé l'annotation des régions détectées dans une image sur trois parmi les 100 premières images. Enfin, les troisième, quatrième et cinquième politiques ont visé l'annotation des régions détectées respectivement dans une image sur six, une image sur douze, et une image sur vingt-quatre, parmi les 199 premières images. Ainsi, pour les trois premières politiques d'annotation, le nombre d'images d'apprentissage est égal à 34, et il est inférieur pour les deux dernières. D'autre part, pour prendre en compte le fait que l'utilisateur n'annotera probablement pas la totalité des régions détectées, seulement 33% des régions détectées sur les images d'apprentissage ont été effectivement utilisées pour entraîner le classificateur.

La figure 6.23 détaille le taux d'erreurs de classification obtenu sur chaque image d'évaluation, c'est-à-dire en excluant l'union des images d'apprentissage utilisées dans chaque cas. Pour les deux premières politiques, où l'annotation a été effectuée sur un ensemble d'images regroupées au début de la vidéo, de bonnes performances sont obtenues sur les images proches des ensembles d'apprentissage (images 34 à 50 pour la première, et images 100 à 120 pour la seconde). Cependant, conformément à l'intuition, les performances obtenues sur les images de la fin de la vidéo, plus éloignées des ensembles d'apprentissage, sont médiocres voire mauvaises (images 80 à 214 pour la première politique, et 130 à 214 pour la seconde). Au contraire, les trois dernières politiques d'annotation, utilisant des images d'apprentissage uniformément réparties dans la vidéo, donnent des performances satisfaisantes tout au long de cette vidéo. Naturellement, plus les images d'apprentissage sont nombreuses, meilleures sont les performances obtenues, comme le montre la comparaison de leurs performances (images 34 à 120). Cependant, les performances obtenues avec ces trois dernières politiques restent relativement proches,

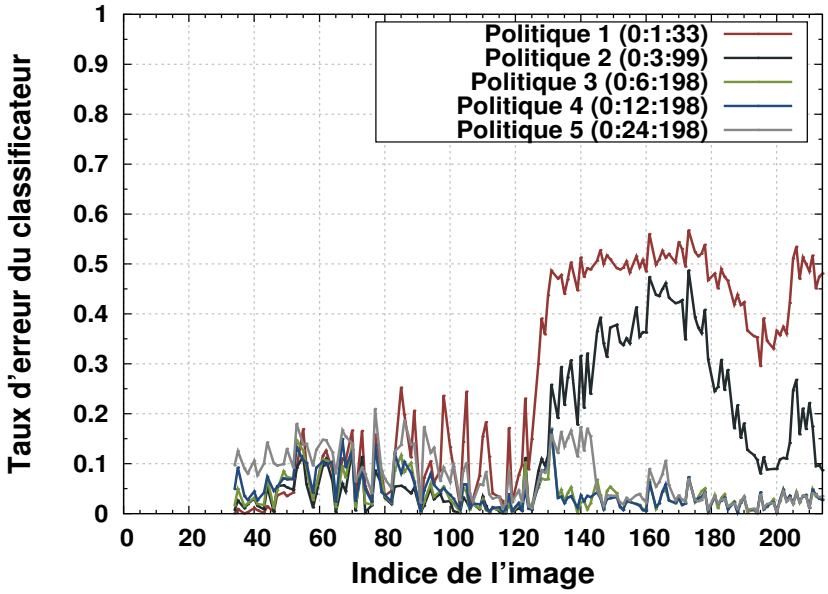


FIGURE 6.23 – *Cette figure compare le taux d'erreurs de classification (moyenne entre taux de faux positifs de classification et de faux négatifs de classification) associés à différentes politiques d'annotation, dans le cas d'un retour interactif de pertinence en ligne. La notation* $0:1:33$ *signifie que l'apprentissage du classificateur a été effectué à l'aide des régions détectées sur les images dont les indices sont inclus dans* $\{0,1,2,\cdots,33\}$. *La notation* $0:3:99$ *correspond à l'ensemble d'indices* $\{0,3,6,\cdots,99\}$ *et la notation* $0:6:198$ *correspond à* $\{0,6,12,\cdots,198\}$. *De même,* $0:12:198$ *correspond à* $\{0,12,24,\cdots,192\}$, *et* $0:24:198$ *correspond à* $\{0,24,48,\cdots,192\}$. *D'autre part, l'évaluation a été effectuée sur l'ensemble total des indices, privé de l'union des ensembles d'apprentissage.*

et deviennent même quasiment identiques pour les dernières images de la vidéo (images 150 à 214).

D'autre part, les performances maximales sont obtenues lorsqu'un maximum d'annotations fournies par l'utilisateur sont utilisées. Cependant, ces annotations demandant un effort, il est peu réaliste de supposer que 100% des régions détectées seront annotées. Par conséquent, nous avons évalué l'impact sur les performances du mécanisme de retour interactif en ligne dans le cas de différentes hypothèses sur le taux d'annotation des régions. La figure 6.24 compare les performances obtenues avec des taux d'annotations de 8%, 16%, 33%, 66% et 100%. Bien que les meilleures performances soient naturellement obtenues avec un taux d'annotation de 100% des régions, ce graphique montre que les différences de performances sont en réalité minimes et qu'elles deviennent même négligeables pour les dernières images de la vidéo.

Enfin, la figure 6.25 présente les performances en détection de changements obtenues grâce au mécanisme en ligne de retour interactif de pertinence. Ce graphique montre ainsi que même avec un taux d'annotation de 8%, il est possible de réduire le taux de faux positifs environ d'un facteur 10 tout en ne réduisant le taux de vrais positifs que d'environ 15%. De plus, ces performances peuvent être améliorées en fournissant plus d'annotations, ce qui permet à la fois de diminuer encore le taux de faux positifs et d'augmenter le taux de vrais positifs.

Ainsi, ceci montre que, dans le cadre d'un mécanisme incrémental de retour interactif de pertinence, il n'est pas nécessaire de fournir des annotations pour chaque image de la vidéo, mais qu'il est important de le faire de manière uniforme à mesure que la vidéo est acquise. Toutefois, afin d'accélérer la convergence du classificateur, il peut être préférable d'adopter une politique d'annotation dégressive. En effet, un nombre d'annotation important au début de la vidéo peut permettre de fiabiliser plus rapidement les prédictions du classificateur. Par la suite, il est nécessaire de mettre à jour le classificateur régulièrement afin de lui permettre de s'adapter aux variations dans la vidéo, mais le nombre d'annotations peut être largement diminué.

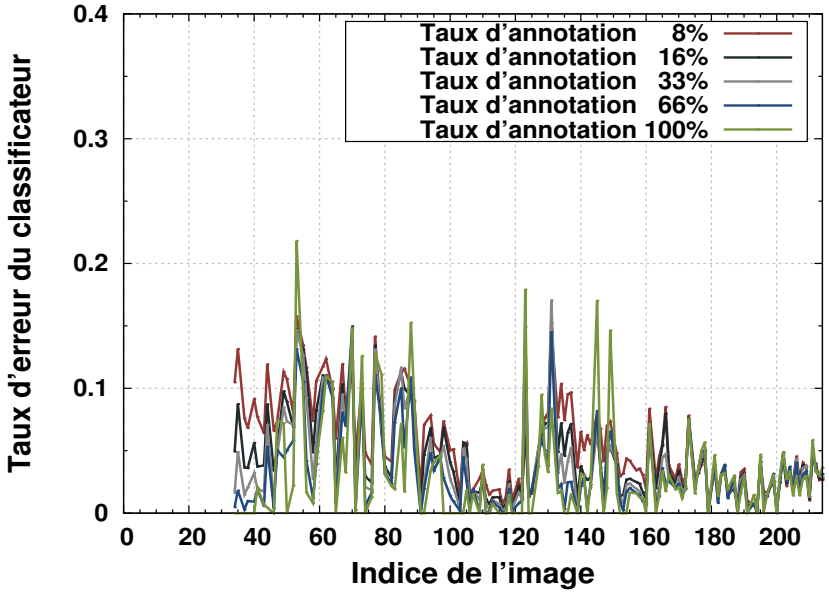


FIGURE 6.24 – *Cette figure compare les taux d'erreur de classification associés à différentes hypothèses sur l'effort d'annotation fourni par l'utilisateur. Le graphique compare les performances obtenues respectivement lorsque* 8% *(en rouge),* 16% *(en noir),* 33% *(en gris),* 66% *(en bleu) et* 100% *(en vert) des régions détectées sur les images d'apprentissage sont annotées par l'utilisateur.*

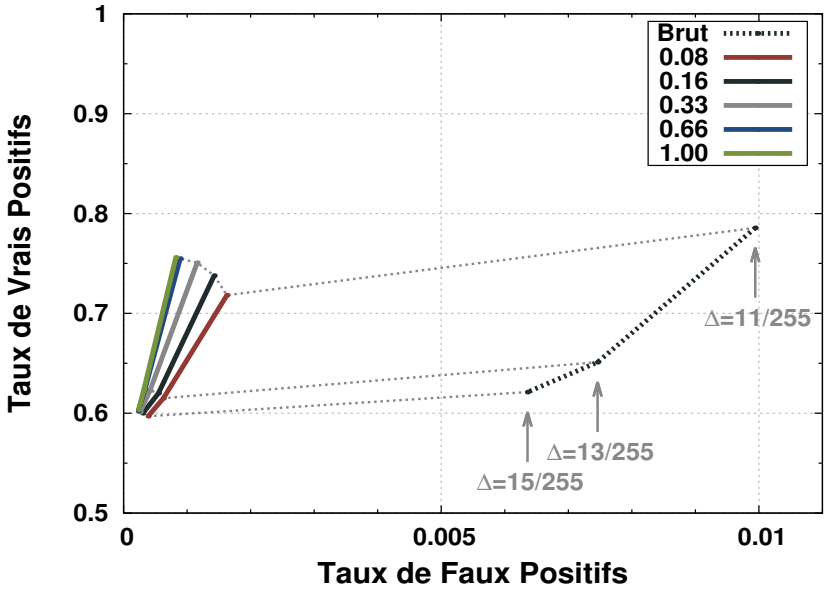


FIGURE 6.25 – *Cette figure compare les performances en détection de changements associées à différentes hypothèses sur l'effort d'annotation fourni par l'utilisateur. Le graphique compare les performances obtenues sans mécanisme de retour interactif (en pointillés) avec celles obtenues respectivement lorsque* 8% *(en rouge),* 16% *(en noir),* 33% *(en gris),* 66% *(en bleu) et* 100% *(en vert) des régions détectées sur les images d'apprentissage sont annotées par l'utilisateur.*

Descripteurs (ms)	Prédiction (ms)	Mise à jour (ms)
646	191	351

TABLE 6.7 – *Temps d'exécution moyens par image, exprimés en millisecondes, des différentes tâches mises en œuvre dans le mécanisme de retour interactif de pertinence : l'extraction des descripteurs de régions, la prédiction des classes et la mise à jour du classificateur à l'aide des annotations de l'utilisateur.*

Temps de calcul Enfin, la table 6.7 présente les temps d'exécution moyens par image pour les différentes tâches mises en œuvre dans le mécanisme de retour interactif de pertinence. Dans le cas de la tâche de mise à jour, la mesure a été effectuée dans l'hypothèse où toutes les régions détectées dans une image donnée sont annotées par l'utilisateur et utilisées pour la mise à jour du classificateur. Même si cette mesure dépend en réalité du nombre de régions utilisées, la valeur fournie permet néanmoins de donner un ordre de grandeur de l'efficacité du mécanisme de retour interactif. Ainsi, la tâche la plus coûteuse en temps de calcul, dont le temps d'exécution reste toutefois inférieur à la seconde, correspond à l'extraction des descripteurs. D'autre part, les temps de mise à jour et de prédiction sont relativement courts, ce qui, du point de vue de l'utilisateur, permet une bonne fluidité du mécanisme de retour interactif.

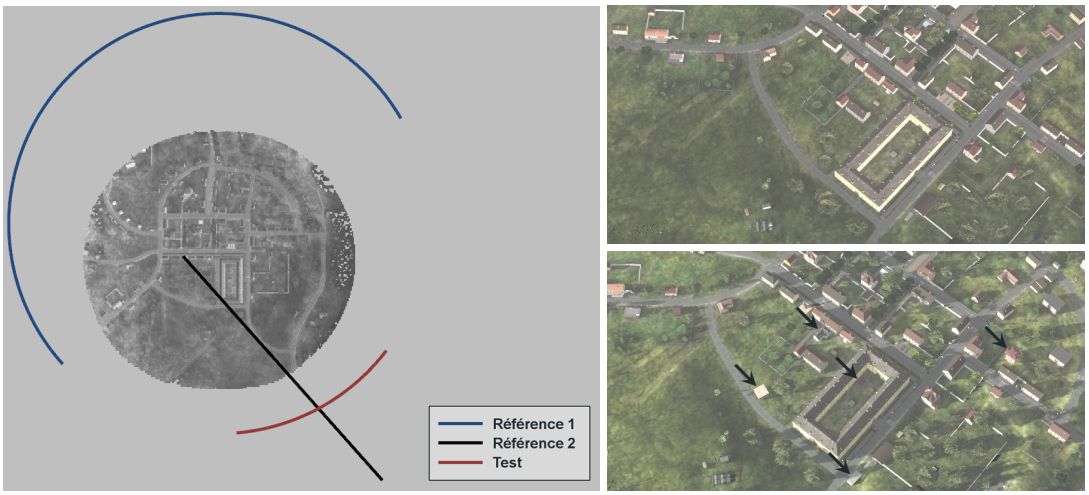
6.5 Discussion générale des résultats

Les résultats présentés dans les sections précédentes montrent que l'approche proposée dans le cadre de cette thèse permet d'atteindre de bonnes performances en détection de changements sur les données d'évaluation utilisées. En effet, la modélisation d'apparence effectuée permet une localisation très précise des changements dans la scène observée et le nombre de fausses alarmes est limité grâce à l'exploitation des techniques de pré-traitement et de consolidation développées dans le cadre de cette thèse. Pour conclure ce chapitre, cette section propose une discussion générale vis-à-vis de la détection de changements par modélisation d'apparence, ainsi que sur les limites de notre approche.

Diversité des apparences de référence Le principe de la détection de changements par modélisation d'apparence repose sur l'idée selon laquelle une meilleure connaissance de la scène observée mène à de meilleures performances. Ceci signifie donc que plus la quantité et la diversité des observations de référence utilisées pour générer le modèle 3D d'apparence sont importantes, plus la détection de changements sera précise. Pour mettre ce point en évidence, nous avons eu recours à des données synthétiques, les données réelles n'étant pas en nombre et en diversité suffisante.

Pour cela, nous avons utilisé deux vidéos de référence et une vidéo de test, dont les trajectoires sont présentées à la figure 6.26a. Pour analyser l'influence de la diversité des apparences dans les vidéos de référence sur les performances, nous avons considéré deux types de variations : des variations de l'illumination et des variations de points de vue. Plus précisément, la première vidéo de référence a été acquise sous les mêmes conditions d'illumination que la vidéo de test, mais selon des points de vue très différents. Au contraire, la seconde vidéo de référence a été acquise sous des conditions d'illumination différentes, mais selon des points de vue proches de ceux de la vidéo de test. La figure 6.26b permet de comparer le contenu des images de la seconde vidéo de référence avec celles de la vidéo de test.

Par la suite, nous avons généré le modèle 3D d'apparence à l'aide des images issues des deux vidéos de référence puis nous avons effectué la détection de changements sur la vidéo de test, avec lissage temporel. Notons que dans ce cas précis, nous n'avons pas effectué d'atténuation de l'illumination, afin de mettre en évidence le gain de performance dû à l'exploitation de



(a) *Trajectoires* **(b)** *Contenu des vidéos*

FIGURE 6.26 – *Cette figure présente les données utilisées pour montrer l'intérêt d'exploiter des données de référence contenant une bonne diversité de conditions d'acquisition, ici en termes de points de vue (a) et d'illumination (b). Les images illustrant le contenu des vidéos, et notamment les différentes conditions d'illumination, correspondent à la seconde vidéo de référence (en haut) et à la vidéo de test (en bas), où les changements sont signalés par des flèches noires.*

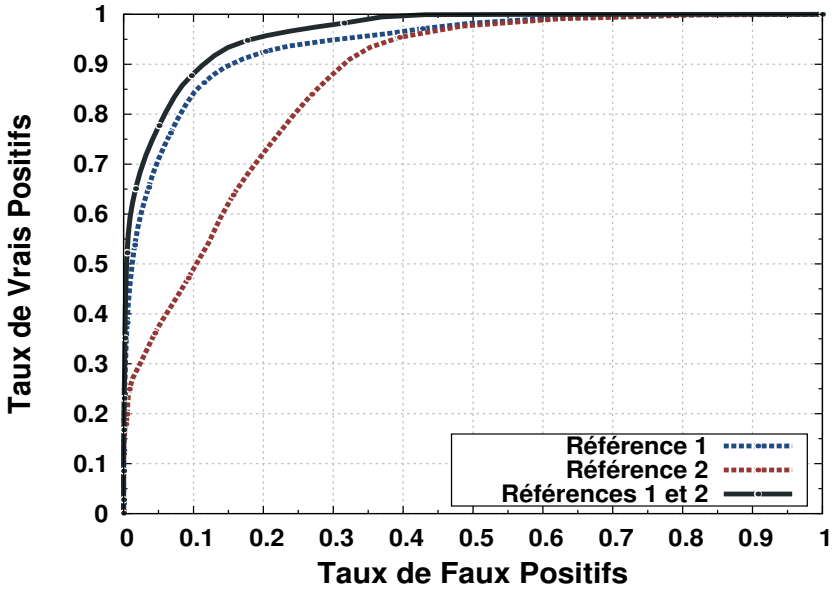
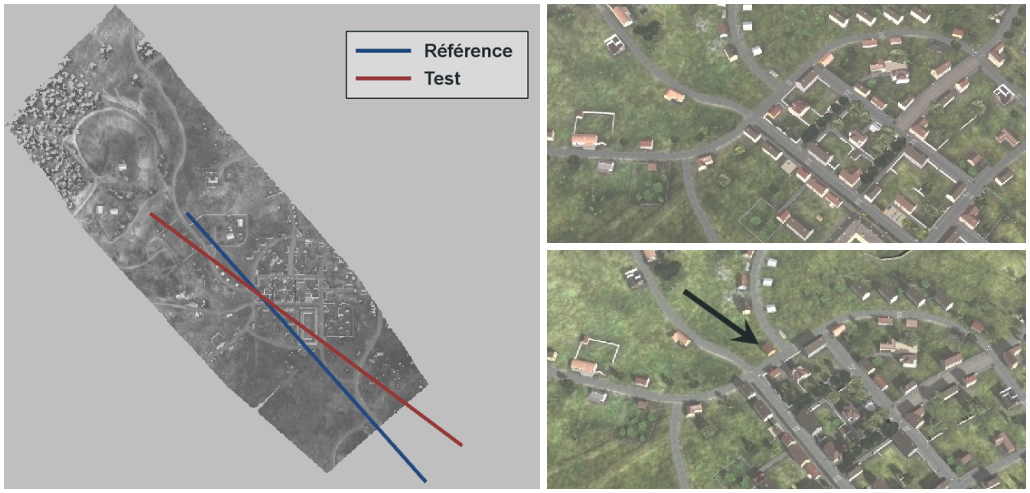


FIGURE 6.27 – *Cette figure compare les courbes ROC associées aux performances de détection de changements obtenues grâce à l'exploitation combinée ou indépendante de deux vidéos de référence acquises sous des conditions différentes.*



(a) *Trajectoires* **(b)** *Contenu des vidéos*

FIGURE 6.28 – *Cette figure présente les données utilisées pour évaluer les performances en précision et rappel. Le schéma de gauche (a) présente les trajectoires des vidéos utilisées et les images de droite (b) illustrent le contenu des vidéos de référence (en haut) et de test (en bas), où le changements est signalé par une flèche noire.*

la diversité dans les vidéos de référence. La figure 6.27 présente les performances de détection de changements obtenues en exploitant les deux vidéos de référence, en les comparant avec celles obtenues lorsque le modèle 3D d'apparence est généré avec une seule des deux vidéos de référence. Ces courbes montrent que l'exploitation combinée des deux vidéos de référence disponibles permet un gain de performances par rapport à l'exploitation indépendante de chacune des deux vidéos de référence. Par conséquent, ceci démontre que l'exploitation d'un maximum de données de référence, acquises sous des conditions aussi diverses que possibles, permet de maximiser les performances de détection de changements.

Pertinence des détections Comme nous l'avons mentionné dans le chapitre 2, l'analyse des performances de détection de changement à l'aide des courbes ROC permet de mesurer la précision de localisation des changements. Cependant, dans le cadre d'un système d'assistance à l'analyse vidéo, il peut être intéressant de quantifier la pertinence des détections, en analysant les mesures de précision et de rappel (voir section 2.6) lorsque la fréquence d'apparition des changements est faible et qu'une majorité d'images ne contiennent aucun changement. En effet, en exploitation opérationnelle, le système de détection de changements peut être amené à analyser plusieurs flux vidéo en parallèle afin de lever des alertes lorsqu'une zone d'intérêt potentiel est détectée. Par conséquent, il est souhaitable que les détections soient les plus pertinentes possible, afin d'éviter que les opérateurs ne soient submergés d'alertes inutiles.

Pour évaluer cette pertinence de détection, nous avons utilisé une vidéo synthétique contenant 620 images, générées à l'aide d'une scène virtuelle dans laquelle un unique changement a été inséré. Ce changement a été positionné de manière à ce qu'il ne soit visible que sur un tronçon de la vidéo représentant environ 35% du nombre total d'images. Pour finir, une seconde vidéo, utilisée comme référence, a été générée à partir de la scène sans changement, sous des conditions d'éclairage différentes et selon une trajectoire légèrement différente. La figure 6.28 présente les trajectoires et le contenu des vidéos utilisées pour cette évaluation de la pertinence des détections. La détection de changements a été effectuée en utilisant les techniques d'atténuation de l'illumination et de consolidation par optimisation spatio-temporelle.

Pour quantifier les résultats, nous avons utilisé deux critères d'évaluation. Le premier de ces critères correspond à celui utilisé jusqu'à présent, qui effectue une analyse pixélique des régions détectées par rapport à la vérité-terrain. Au contraire, le second critère considère les

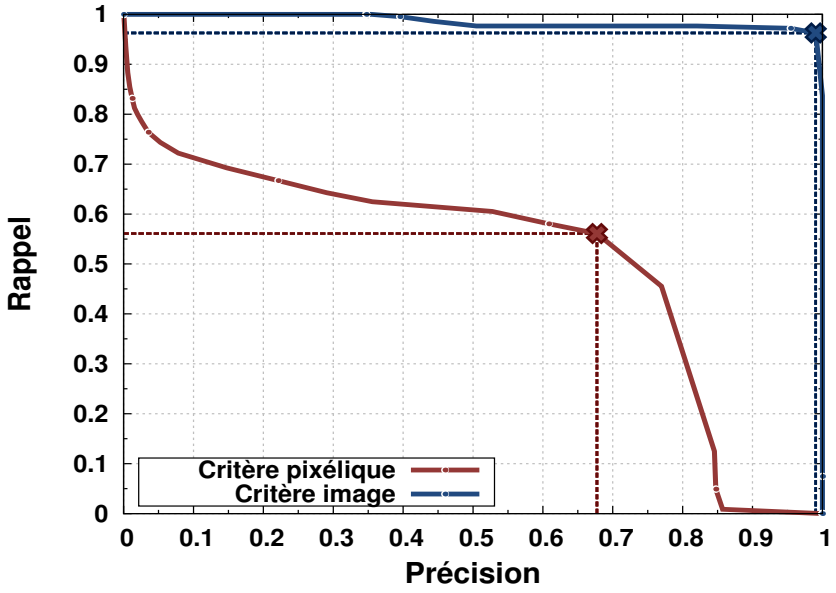


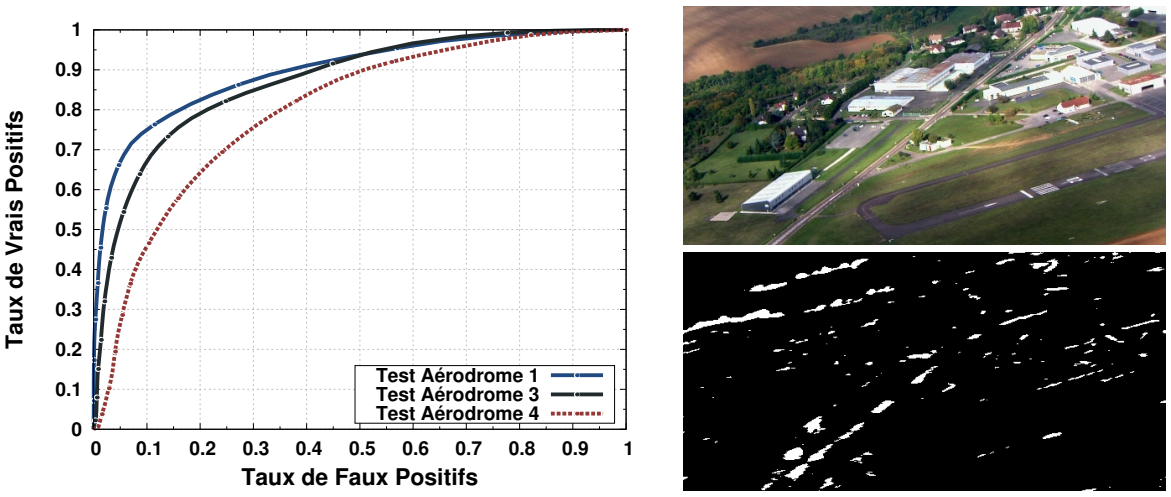
FIGURE 6.29 – *Cette figure compare les performances de détection de changements en précision et rappel calculées à l'aide d'un critère d'évaluation pixélique (en rouge) et d'un critère considérant les images entières (en bleu).*

images de manière globale, et analyse la présence ou l'absence de changements sur l'image considérée sans se préoccuper de la localisation de ces changements. Ainsi, ce second critère considère qu'une image donnée contient un changement lorsqu'au moins un de ses pixels est détecté comme étant un changement [4]. Un tel critère est intéressant en pratique, car dans certains scénarios opérationnels, c'est le plus adapté pour mesurer la pertinence de la focalisation d'attention. Par exemple, dans le cas de l'assistance à l'analyse de nombreux flux vidéos en parallèle, la focalisation automatique d'attention peut consister, à un instant donné, à présenter à l'analyste image un unique flux vidéo parmi les différents flux analysés. Le système de détection de changement doit donc évaluer la pertinence d'une image entière, sans se préoccuper de la localisation des changements présents dans cette image.

La figure 6.29 compare les performances de détection de changements en précision et rappel, calculées à l'aide de ces deux critères d'évaluation. Ces courbes montrent de très bonnes performances de détection. En effet, la courbe correspondant au critère pixélique montre qu'il est possible de trouver un point de fonctionnement, celui mis en évidence par une croix rouge, permettant d'atteindre un taux de plus de 65% de détections correspondant bien à un changement pertinent et un taux de plus de 55% de changements pertinents détectés. Dans le cas du critère image, il est possible de trouver un point de fonctionnement pour lequel environ 99% des images sélectionnées contiennent effectivement un changement pertinent et plus de 96% des images contenant des changements sont correctement sélectionnées.

Pour illustrer ces résultats sur un exemple concret, considérons une vidéo d'une heure acquise à 5 images par seconde, contenant donc 18 000 images. Considérons de plus un cas plus réaliste que l'exemple ci-dessus, dans lequel 10% des images de la vidéo contiennent effectivement un changement pertinent. Dans un tel cas, une précision de 99% et un rappel de 96% sont équivalent à la génération de 17 faux positifs et 72 faux négatifs, soit moins d'un faux positif toutes les 3 minutes et 96% des changements correctement détectés. Ces résultats montrent donc que la pertinence des détections générées par notre approche est tout à fait satisfaisante, voire excellente dans le cas d'une évaluation avec le critère image.

4. Notons que ce critère nécessite donc de sur-estimer le seuil de détection, afin de réduire au maximum les fausses alarmes.



(a) *Courbes ROC* **(b)** *Inspection visuelle*

FIGURE 6.30 – *Le graphique de gauche (a) présente les courbes ROC associées aux performances de détection de changements obtenues sur les vidéos* Aérodrome 1*,* Aérodrome 3 *et* Aérodrome 4*. À droite (b) sont présentées une image de la vidéo* Aérodrome 4 *avec le masque de changements correspondant.*

Limites Comme l'ont montré les résultats exposés jusqu'à présent, l'approche présentée dans ce manuscrit permet d'obtenir des performances intéressantes dans le cadre du scénario opérationnel envisagé à la section 1.2.3. Cependant, la pertinence de cette approche montre ses limites si nous nous éloignons des hypothèses de ce scénario opérationnel.

Résolution au sol Pour commencer, la résolution au sol des données joue un rôle important au sein notre approche, car, comme indiqué à la section 4.2.1.2, la résolution du modèle 3D d'apparence dépend directement de la résolution au sol maximale dans les vidéos de référence. Par conséquent, si la résolution au sol dans la vidéo de test est trop différente de celle des vidéos de référence, ceci peut causer de mauvaises performances à plusieurs niveaux.

D'une part, ceci peut perturber l'algorithme de géo-localisation par asservissement visuel et mener à des imprécisions de géo-localisation. En effet, si la résolution au sol dans la vidéo de test est supérieure à celle des vidéos de référence, alors l'image de rendu issue du modèle 3D d'apparence sera sous-résolue et contiendra des artefacts (voir figure 4.9), du fait de l'approximation de la scène observée sous forme de carte d'élévation. Par conséquent, la transformation de recalage estimée entre une image de test donnée et l'image de rendu du modèle 3D associée sera imprécise, voire aberrante, ce qui pourra conduire à une divergence de l'erreur de géo-localisation. Si, au contraire, la résolution au sol dans la vidéo de test est inférieure à celle des vidéos de référence, il est possible que l'étendue du modèle 3D d'apparence ne soit pas suffisante pour permettre un recalage fiable entre une image de test donnée et l'image de rendu associée. Cependant, si le modèle 3D d'apparence est suffisamment étendu, il est probable que les performances de géo-localisation seraient moins affectées que dans le cas précédent.

D'autre part, une différence de résolution importante peut également rendre difficilement comparable les modèles d'apparence et les observations de test. En effet, si la résolution au sol de la vidéo de test est supérieure à celle des vidéos de référence, l'approximation de la scène observée peut causer des erreurs de mise en correspondance, du fait d'une mauvaise représentation du relief (e.g. bâtiments, arbres) ou du contenu de la scène (e.g. champs, routes). De plus, une résolution au sol trop différente entre les observations de référence et de test rend les modèles d'apparence calculés sur la vidéo de référence peu adaptés pour la détection de changements dans la vidéo de test, puisqu'alors ils ne décrivent plus le même contenu. Par conséquent,

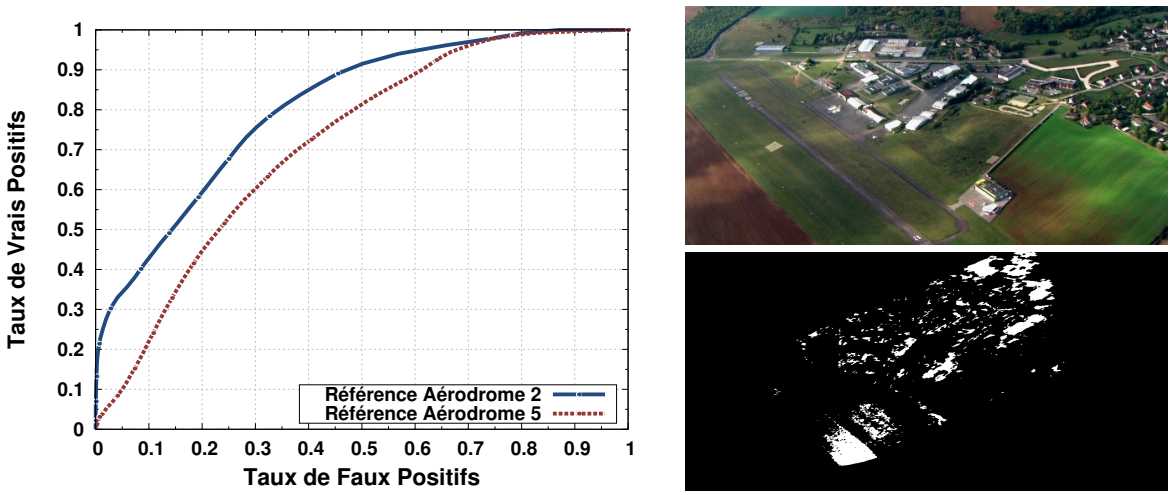
une différence trop importante de résolution au sol peut causer de mauvaises performances en détection de changements.

Pour illustrer ce point, nous avons utilisé la vidéo *Aérodrome 2* comme vidéo de référence, puis nous avons comparé les performances en détection de changements obtenues sur les vidéos *Aérodrome 1* et *Aérodrome 3*, acquises à une résolution au sol comparable avec celle de la vidéo de référence, avec les performances obtenues sur la vidéo *Aérodrome 4*, acquise selon une trajectoire plus proche de la scène et avec un zoom plus important. Cette détection de changements a été effectuée en appliquant l'atténuation de l'illumination ainsi que la consolidation par lissage temporel. La figure 6.30a présente les courbes ROC obtenues ainsi que l'exemple d'une image et du masque de changements associé. Les performances associées à la vidéo *Aérodrome 4* sont inférieures à celles obtenues sur les vidéos *Aérodrome 1* et *Aérodrome 3*. Comme le montre l'exemple de la figure 6.30b, un nombre important de faux positifs sont générés dans le masque de changements, qui sont principalement dûs à l'approximation de la scène et aux petits détails qui n'apparaissent pas à une résolution inférieure.

Pour aborder ce problème, trois solutions peuvent être envisagées. La première consiste, de manière similaire à l'approche proposée par Crispell et al. [34], à essayer de maximiser la résolution du modèle 3D d'apparence indépendamment de la résolution des données de référence. En effet, ceci permettrait de rendre imperceptible l'approximation de représentation de la scène, pour une large gamme de résolution au sol dans les données de test. D'autre part, il peut être envisagé de réduire artificiellement la taille de l'image de test, pour qu'elle corresponde approximativement à la taille des images de référence. Pour connaître le facteur de réduction à appliquer, une solution possible peut consister à effectuer régulièrement, mais pas forcément à chaque image, un recalage préalable entre l'image de test et une image de référence proche. Ceci peut permettre d'estimer le facteur de mise à l'échelle entre les observations de test et celles de référence, qui pourrait par la suite être utilisé pour réduire l'image de test avant son exploitation. Enfin, afin de permettre une comparaison adaptée entre les modèles d'apparence et les observations de test, il pourrait être intéressant de mettre en œuvre une approche multi-échelle. Par exemple, un modèle d'apparence pourrait être calculé à chaque niveau hiérarchique du Quad-Tree augmenté, ce qui, par la suite, pourrait permettre de choisir l'ensemble de modèles d'apparence adapté pour effectuer la détection de changements à l'aide d'une image de test donnée.

Délai temporel Pour finir, le délai temporel entre les acquisitions de référence et de test est un facteur essentiel pour assurer un certain niveau de pertinence dans les détections. En effet, si la vidéo de référence, utilisée pour analyser une vidéo de test donnée, a été acquise à une date trop éloignée de cette dernière, il est probable qu'elle mène à une mauvaise modélisation des variations normales dans la scène observée par la vidéo de test. Ceci a souvent pour conséquence de générer un grand nombre de fausses alarmes.

Pour illustrer ce point, nous avons cherché à détecter les changements dans la vidéo *Aérodrome 1*, en utilisant comme vidéo de référence d'abord une partie de la vidéo *Aérodrome 2*, acquise à quelques minutes d'intervalle, puis la vidéo *Aérodrome 5*, acquise plus d'un mois et demi après la vidéo de test. Cette détection de changements a été effectuée avec l'atténuation de l'illumination ainsi que la consolidation par lissage temporel. De plus, pour permettre une comparaison objective, nous n'avons pas utilisé la totalité de la vidéo *Aérodrome 2* mais seulement certaines images, en nombre égal à celles de la vidéo *Aérodrome 5* et sélectionnées pour correspondre à la même gamme de points de vues. La figure 6.31a présente les courbes ROC obtenues ainsi que l'exemple d'une image et du masque de changements associé. Comme attendu, les performances associées à l'utilisation comme référence de la vidéo *Aérodrome 5* sont nettement inférieures à celles obtenues avec la vidéo *Aérodrome 2*. Ceci est dû à la présence de nombreuses fausses alarmes, générées à cause d'une mauvaise modélisation des variations nor-



(a) *Courbes ROC* **(b)** *Inspection visuelle*

FIGURE 6.31 – *Le graphique de gauche (a) présente les courbes ROC associées aux performances de détection de changements obtenues sur la vidéo* Aérodrome 1*, en utilisant les vidéos* Aérodrome 2 *ou* Aérodrome 5 *comme référence. À droite (b) sont présentées une image de la vidéo* Aérodrome 1 *avec le masque de changements estimé lorsque la vidéo* Aérodrome 5 *est utilisée comme référence. Notons que le masque de changements présenté ne représente qu'une portion limitée de l'image de test, car les observations de référence disponibles dans la vidéo* Aérodrome 5 *ne permettaient pas de couvrir l'ensemble de cette image. Copyright © 2010 - 2012 Cassidian - All rights reserved.*

males de la scène observée dans la vidéo de test. Comme le montre l'exemple de la figure 6.31b, ces fausses alarmes consistent pour la plupart à des changements de végétation dans la scène.

Afin d'améliorer la robustesse de l'approche proposée, une solution possible peut consister à améliorer la modélisation des variations normales de la scène. Ceci peut par exemple être effectué en intégrant, dans la base de données de référence, des vidéos contenant les changements de végétation observés dans la vidéo de test et, plus généralement, toute variation d'apparence considérée comme non pertinente pour la détection de changements. Comme nous l'avons vu plus haut, ceci peut par exemple être fait en diversifiant les conditions d'acquisition des vidéos de référence utilisées. Cette diversification permet en effet d'affiner la précision du modèle 3D d'apparence, en identifiant les modes de variation les plus caractéristiques de la scène. Bien que ceci n'ait pas été testé dans le cas ci-dessus par manque de données, il est probable qu'une diversification des conditions d'acquisition des vidéos de référence mène à une amélioration des performances, comme dans l'exemple présenté au début de cette section.

Cependant, de manière plus générale, le critère de détection de changements basé sur les apparences possède une sensibilité intrinsèque vis-à-vis des variations de conditions d'acquisition. Par conséquent, les performances peuvent être considérablement réduites lorsqu'il est impossible de modéliser correctement les variations normales de la scène dans la vidéo de test. Certes, nous avons montré dans ce manuscrit que, dans certains cas tels que les variations de points de vue ou d'illumination, cette sensibilité pouvait être atténuée sans l'aide de données de référence spécifiques. Toutefois, pour les scénarios applicatifs où les conditions d'acquisition sont souvent très différentes (e.g. délais temporels considérables, observations jour / nuit, etc) entre les données de référence et les données de test, d'autres critères de détection de changements peuvent être plus adaptés. Une piste intéressante pour cela peut par exemple consister à exploiter l'information mutuelle [49], afin d'effectuer une comparaison indépendante de l'apparence des objets.

Chapitre 7

Conclusion et perspectives

COMME l'ont justifié les chapitres 1 et 2, la détection de changements dans des vidéos aériennes est un problème vaste, qui met en jeu de nombreuses problématiques. Bien sûr, celle de la comparaison des données est au cœur de toute approche de détection de changements. Cependant, nous avons vu que d'autres jouent également un rôle essentiel, en particulier les problématiques de géo-localisation, d'atténuation de l'illumination, de reconstruction 3D, d'indexation ou encore de compression des observations.

Dans le cadre de cette thèse, nous avons choisi d'orienter nos travaux vers une approche semi-automatique, visant à assister un opérateur dans sa tâche d'analyse vidéo. Cette approche met donc ponctuellement cet opérateur à contribution, notamment pour la géo-localisation des données de référence (section 3.1.1) et pour la consolidation interactive des résultats de détection (section 5.3). Cependant, afin de minimiser l'effort requis de la part de l'opérateur, nous nous sommes concentrés sur la maximisation de la quantité d'information extraite des données disponibles, en proposant des algorithmes innovants tels que la géo-localisation par asservissement visuel (section 3.1.2), la consolidation des détections par optimisation spatio-temporelle (section 5.1.2) ou encore la binarisation par extraction de MSER (section 5.2).

Bilan des travaux Plus généralement, l'ensemble des solutions et des méthodes développées dans le cadre de cette thèse forment une approche complète et cohérente pour la détection de changements dans des vidéos aériennes.

Ainsi, les algorithmes de pré-traitements, utilisés pour préparer la détection effective des changements, sont regroupés au chapitre 3. Un premier ensemble d'algorithmes vise la géo-localisation des vidéos considérées, c'est-à-dire l'estimation de leurs paramètres d'acquisition. Pour cela, nous avons proposé deux algorithmes distincts, qui permettent de répondre aux différents besoins survenant dans le cas des données de référence ou de test. Ainsi, un algorithme semi-automatique d'interpolation de poses a été proposé pour la géo-localisation hors-ligne des vidéos de référence, pour lesquelles aucun modèle de la scène n'est encore disponible. Pour les vidéos de test, un algorithme d'asservissement visuel a été proposé pour permettre une géo-localisation rapide et incrémentale, exploitant le modèle généré à partir des vidéos de référence. D'autre part, le second ensemble d'algorithmes de pré-traitements vise à atténuer les effets de l'illumination, qui peuvent générer un grand nombre d'erreurs de détection. Pour cela, nous avons choisi de convertir les observations dans une représentation invariante aux variations d'illumination. En effet, cette technique, qui exploite les moyennes des observations de référence en chaque point de la scène, est très rapide et permet de traiter simplement une gamme importante de variations dues à l'illumination.

La méthode de détection de changements que nous avons développée a ensuite été décrite en détails au chapitre 4. Cette méthode consiste à estimer un modèle 3D d'apparence à partir des observations contenues dans les vidéos de référence. Pour cela, elle repose sur la combinaison de deux techniques de modélisation, qui permettent toutes deux d'effectuer la majeure

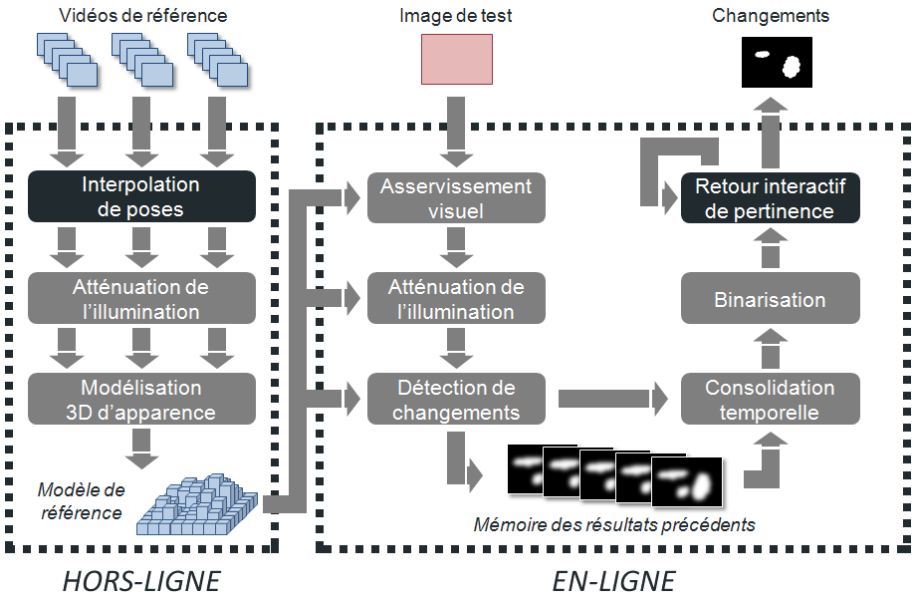


FIGURE 7.1 – *Cette figure présente le schéma synthétisant le fonctionnement général de notre approche de détection de changements dans des vidéos aériennes. Les tâches nécessitant une intervention de l'utilisateur (géo-localisation par interpolation de poses et consolidation par retour interactif de pertinence) sont mise en évidence par un bloc sombre.*

partie des calculs de manière hors-ligne. La première technique consiste à calculer un modèle tri-dimensionnel de la scène, afin de pouvoir gérer correctement et simplement les effets géométriques dûs aux changements de points de vue, qui surviennent fréquemment dans les vidéos aériennes. La seconde technique consiste a effectuer une modélisation des apparences observées dans la scène, ce qui permet d'exploiter la redondance présente dans les vidéos afin d'assurer une bonne robustesse au bruit et aux perturbations diverses relatives aux apparences des objets de la scène. Ce modèle 3D d'apparence peut ensuite être exploité de manière incrémentale pour détecter les changements dans une vidéo de test.

Enfin, un certain nombre d'algorithmes permettant la consolidation des résultats de détection de changements ont été présentés au chapitre 5. Pour cela, nous avons cherché à modéliser la connaissance a priori relative aux changements d'intérêt pour l'analyste image, ce qui a mené à trois pistes de consolidation. La première piste de consolidation que nous avons exploré consiste à exploiter la redondance spatio-temporelle présente dans la vidéo de test pour améliorer la détection de changements fixes dans la scène. Les deux algorithmes développés pour cela permettent un traitement incrémental de la vidéo de test considérée, et exploitent un lissage temporel ou une optimisation spatio-temporelle des scores de détection. La deuxième piste de consolidation permet d'améliorer les résultats de détection correspondant aux objets dont les frontières sont bien définies (e.g. structures artificielles, personnes, etc), et utilise l'algorithme d'extraction de MSER pour effectuer une analyse fine de la carte des scores de détection. Pour finir, afin d'augmenter la flexibilité de notre approche en permettant à l'analyste image d'adapter les résultats de détection à ses besoins, nous avons développé un mécanisme de retour interactif de pertinence, permettant de filtrer les fausses alarmes résiduelles.

Un schéma synthétisant le fonctionnement général de notre approche de détection de changements est présenté à la figure 7.1.

Par ailleurs, nous avons vu que l'évaluation des méthodes de détection de changements dans le cadre de vidéos aériennes est un problème délicat, du fait de la difficulté d'obtenir des données réelles pertinentes. Pour contourner ce problème, nous avons proposé une technique permettant d'insérer, par réalité augmentée, des changements d'intérêt dans des vidéos aériennes

initialement exemptes de changements significatifs. Outre le fait qu'elle permet de conserver la complexité des données réelles, en termes de bruit, de sur-exposition ou sous-exposition des pixels et autres perturbations diverses, cette technique permet d'obtenir très simplement la vérité-terrain, ce qui est d'un intérêt considérable.

Grâce à ces données d'évaluation, les résultats de notre approche ont pu être analysés de manière quantitative et systématique au chapitre 6, qui a montré des performances très satisfaisantes malgré la complexité des données. Par ailleurs, ces données ont également permis d'effectuer une comparaison objective de nos choix d'algorithmes intermédiaires avec d'autres solutions existantes dans la littérature. Ces expérimentations ont ainsi montré que les performances obtenues à l'aide de notre approche sont supérieures à celles obtenues à l'aide d'autres méthodes comparables, parmi celles que nous avons pu implémenter.

Perspectives Naturellement, malgré ces bonnes performances, l'approche de détection de changements présentée dans ce manuscrit est perfectible. En effet, un certain nombre de pistes d'amélioration auraient pu être explorées si nous avions disposé de plus de temps. Les principales pistes envisagées sont évoquées ci-dessous.

Accélération des algorithmes Pour commencer, bien que nous ayons pris soin d'effectuer une implémentation rapide et efficace des différents algorithmes proposés, une accélération matérielle pourrait être envisagée pour certains d'entre eux. Une telle accélération pourrait en effet permettre d'atteindre une exécution temps réel de l'approche de détection de changements que nous avons développée, ce qui constitue généralement l'objectif ultime des techniques de traitement en ligne.

Plus particulièrement, l'algorithme de lancer de rayon est utilisé de manière intensive tout au long de notre approche, afin d'effectuer la mise en correspondance des pixels considérés avec les cellules du modèle 3D d'apparence. Or, cet algorithme est tout à fait adapté à un portage sur GPU puisque le même traitement est répété de manière indépendante pour chaque pixel.

De la même façon, les deux algorithmes de consolidation temporelle, pourraient être de bons candidats à un portage sur GPU. Notamment, l'accélération de l'algorithme de consolidation par optimisation spatio-temporelle le rendrait considérablement plus attractif, au vu des excellentes performances de détection de changements qu'il permet d'ores et déjà d'obtenir.

Extension de la géo-localisation automatique La géo-localisation des images considérées constitue une tâche essentielle pour la détection de changements, mais applicable à de nombreux autres problèmes opérationnels. Les algorithmes développés pour cela ouvrent donc un grand nombre de pistes à explorer pour étendre leurs fonctionnalités.

Ainsi, une extension intéressante pour notre algorithme d'interpolation de poses concerne la minimisation de l'effort requis de la part de l'utilisateur pour la calibration des images-clés. Pour cela, un premier pas pourrait consister à déterminer automatiquement le nombre minimal et les indices associés des images-clés, pour la géo-localisation d'une vidéo donnée. Ceci pourrait par exemple être fait en analysant la similarité entre les images pour trouver celles donnant le meilleur compromis entre le nombre d'images-clés et l'intensité de la distorsion entre images-clés et images intermédiaires. Une autre piste concerne l'exploitation de vidéos déjà géo-localisées sur la même zone, pour minimiser le nombre de nouvelles images à calibrer manuellement. En effet, l'utilisation du tenseur trifocal n'est pas limitée à l'interpolation dans une vidéo, mais peut également servir entre deux vidéos différentes si les images considérées sont suffisamment similaires. Enfin, diverses manières d'améliorer la précision de la géo-localisation pourrait être envisagées, par exemple en exploitant les méta-données (coordonnées GPS, mesures d'orientations, etc) pour initialiser l'estimation, ou en imposant une certaine continuité entre les paramètres d'acquisition des images successives. Notons que, d'un point de vue opé-

rationnel, l'automatisation de ces traitements présenterait un intérêt considérable pour la fouille de données géo-localisées.

D'autre part, de nombreuses extensions de notre algorithme d'asservissement visuel peuvent également être envisagées. En particulier, la transformation de recalage, qui est utilisée pour ajuster les paramètres d'acquisition, peut être estimée selon de nombreuses méthodes, dont certaines pourraient notamment permettre d'améliorer la précision de l'algorithme. De plus, un cas intéressant serait d'effectuer un recalage basé sur l'information mutuelle [35, 123], ce qui pourrait permettre d'adapter l'algorithme à des modèles 3D qui ne seraient pas forcément photo-réalistes (e.g. maquette numérique d'un avion issue des bureaux d'études). Une autre piste d'exploration concerne l'amélioration de la robustesse d'estimation des paramètres de calibration. Enfin, l'algorithme présenté dans ce manuscrit utilise, pour guider l'estimation des paramètres d'acquisition, le rendu du modèle 3D basé sur la moyenne des observations de référence. Ceci peut poser problème lorsque la moyenne des observations de référence est trop différente des observations contenues dans la vidéo de test, et mener vers un échec du recalage et donc de la géo-localisation. Puisque le modèle 3D contient également des modèles d'apparence, il pourrait être intéressant de chercher à les exploiter afin d'améliorer la robustesse de la méthode.

Approfondissement de la modélisation d'apparence Par ailleurs, la modélisation d'apparence est au cœur de l'approche de détection de changements décrite dans ce manuscrit, et les travaux réalisés ont permis d'identifier diverses pistes pouvant permettre l'amélioration des performances.

Pour commencer, il pourrait être intéressant d'exploiter la structure arborescente du modèle 3D d'apparence, afin de pouvoir utiliser les modèles d'apparence les plus adaptés aux observations de test considérées. En effet, une telle approche hiérarchique (*coarse-to-fine* dans la littérature), présenterait un double avantage. D'une part, elle pourrait permettre de traiter plus rapidement les grandes zones uniformes dans le modèle 3D d'apparence. En effet, dans la version présentée ici, les modèles d'apparence n'existent qu'au niveau des feuilles du Quad-Tree augmenté. Les grandes zones uniformes sont donc considérées comme un ensemble de modèles d'apparence quasiment identiques mais très localisés, ce qui pourrait être optimisé. D'autre part, le second avantage serait la possibilité de traiter correctement les vidéos de test dont la résolution au sol diffère de celle des vidéos de référence. En effet, il serait alors possible d'effectuer la détection de changements en sélectionnant le modèle d'apparence qui correspond le mieux à la résolution de l'observation de test considérée.

Par ailleurs, nous avons montré que la modélisation par Analyse incrémentale en Composantes Principales (ACP incrémentale) donnait les meilleures performances parmi les algorithmes testés. Cet algorithme présente également le double intérêt d'être bien adapté au traitement de données vidéo, puisqu'il permet de traiter les vidéos de référence sans garder l'ensemble des observations en mémoire, et de pouvoir être formulé de manière robuste aux données manquantes, qui surviennent fréquemment du fait des occultations dues aux effets géométriques. Cependant, la technique de l'ACP reste relativement basique. Il a ainsi été montré à plusieurs reprises que d'autres techniques permettaient de donner de meilleures performances, dans le cadre de la détection de changements dans des images satellites, notamment l'analyse des corrélations canoniques (CCA, pour *Canonical Correlation Analysis* dans la littérature) [85] ou l'Analyse en Composantes Indépendantes [72]. Des versions incrémentales existent pour certaines de ces techniques (voir notamment [110] pour la CCA), toutefois, aucune application à la modélisation d'apparence n'a semble-t-il été publiée.

Enfin, il pourrait être intéressant d'étudier des critères différents de l'apparence, pour effectuer la détection de changements. En effet, les apparences des objets d'une scène données sont sujettes à de nombreuses variations, rendant la détection de changements très sensibles à de nombreuses perturbations (voir par exemple la section 6.5). Cette sensibilité pourrait être

contournée en utilisant un critère plus stable, analysant par exemple la correspondance des contours ou l'information mutuelle.

Détection de changements hors-ligne Pour finir, les travaux réalisés montrent que les performances de détection de changements pourraient être améliorées si la détection de changements était effectuée de manière hors-ligne, c'est-à-dire une fois que la vidéo de test complète est disponible. En effet, ceci ouvrirait de nombreuses possibilités d'exploitation, dont une liste non-exhaustive d'exemples sont mentionnés ci-dessous.

En premier lieu, nous avons vu que l'erreur de reprojection associée à l'estimation hors-ligne des trajectoires d'acquisition des vidéos était plus faible que celle associée à leur estimation en-ligne. Nous avons montré à la section 6.2.1 que ceci pouvait avoir un impact sur les performances de détection de changements.

En second lieu, une détection hors-ligne des changements pourrait permettre d'employer une généralisation du cadre de la consolidation par optimisation spatio-temporelle, afin d'effectuer la comparaison de l'ensemble des images de référence avec l'ensemble des images de test. Cette généralisation pourrait donc permettre une exploitation plus poussée de la redondance dans la vidéo de test, qui pourrait mener à un important gain de performances.

Enfin, une détection de changements hors-ligne pourrait permettre la mise en œuvre de techniques d'apprentissage actif, que nous avons évoquées à la section 5.3.1, dans le cadre du mécanisme de retour interactif de pertinence. Ces techniques permettent d'optimiser l'ordre d'annotation des détections par l'utilisateur, de manière à maximiser la quantité d'information qui en résulte. Associées à une détection de changements hors-ligne, ces techniques d'apprentissage actif pourraient ainsi déboucher sur d'excellentes performances.

Démonstrations Annexes

LES démonstrations et calculs intermédiaires relatifs aux algorithmes présentés dans ce manuscrit n'ont pas tous été fournis dans les chapitres correspondants par souci de place et de clarté d'exposition. Les détails de ces calculs pouvant malgré tout être intéressants, ils sont regroupés dans les sections annexes qui suivent.

La première partie de ces calculs est relative à l'algorithme de géo-localisation par asservissement visuel. Ils permettent de démontrer l'expression analytique de l'homographie de recalage entre deux images dont les paramètres d'acquisition sont connus. Cette expression analytique permet ensuite de déduire l'expression des termes de linéarisation lorsque les paramètres d'acquisition sont proches.

La seconde partie concerne les calculs relatifs à la technique d'atténuation de l'illumination, à l'aide des coordonnées chromatiques logarithmiques. Ils permettent de démontrer la paramétrisation de la matrice de projection en fonction d'un unique paramètre angulaire.

Sommaire

A.1 Expressions analytiques pour l'asservissement visuel

Les trois sections suivantes démontrent les expressions analytiques sur lesquelles se base l'algorithme d'asservissement visuel présenté à la section 3.1.2. La section A.1.1 dérive l'expression analytique de la matrice de recalage entre deux images observant un plan. La section A.1.2 fournit l'expression des termes de linéarisation de cette matrice dans le cas restreint où les paramètres de calibration des deux caméras sont connus. Enfin, la section A.1.3 fournit l'expression des termes de linéarisation dans le cas plus général où les paramètres de calibration sont inconnus.

A.1.1 Expression analytique de la matrice de recalage

Nous reprenons ici les mêmes notations que celles introduites dans la section 3.1.2. D'autre part, nous désignons l'expression de la distance orthogonale d'un point $\mathbf{M}$ au plan π par $\pi(\mathbf{M}) = \langle \mathbf{n}_\pi | \mathbf{M} \rangle - d_\pi$. De plus, pour $i \in \{1,2\}$, nous désignerons par $\mathbf{C}_i$ le vecteur position de la caméra $\mathcal{C}_i$ et par $\mathcal{R}_i = \{\mathbf{V}x_i, \mathbf{V}y_i, \mathbf{V}z_i\}$ la base de vecteurs orthonormés représentant le système de coordonnées liée à la caméra $\mathcal{C}_i$. Ces vecteurs sont orientés selon la convention utilisée en vision artificielle : $\mathbf{V}z_i$ est orienté selon l'axe optique de $\mathcal{C}_i$, $\mathbf{V}x_i$ est orienté de la gauche vers la droite de l'image I_i et $\mathbf{V}y_i$ du haut vers le bas de I_i. De plus, les vecteurs d'orientation, les vecteurs positions ainsi que le vecteur $\mathbf{n}_\pi$ et tous les vecteurs 3D considérés dans la suite, sauf mention contraire, sont supposés exprimés dans le repère terrestre canonique $\mathcal{R}_0$.

Nous pouvons alors exprimer la matrice de rotation transformant les coordonnées du repère terrestre canonique $\mathcal{R}_0$ vers le repère $\mathcal{R}_i$ à l'aide des angles d'Euler $\{\psi_i, \theta_i, \phi_i\}$ dans la convention ZYX (ou convention de Tait-Bryan) :

$$\begin{aligned} \mathrm{R}_{\mathcal{R}_i \leftarrow \mathcal{R}_0}(\psi_i, \theta_i, \phi_i) &= \mathrm{PERM}_{Z\leftarrow X,\ -X\leftarrow Y,\ -Y\leftarrow Z} \cdot \mathrm{R}_X(\phi_i) \cdot \mathrm{R}_Y(\theta_i) \cdot \mathrm{R}_Z(\psi_i) \\ &= \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi_i & -sin\phi_i \\ 0 & sin\phi_i & cos\phi_i \end{bmatrix} \cdot \begin{bmatrix} cos\theta_i & 0 & sin\theta_i \\ 0 & 1 & 0 \\ -sin\theta_i & 0 & cos\theta_i \end{bmatrix} \cdot \begin{bmatrix} cos\psi_i & -sin\psi_i & 0 \\ sin\psi_i & cos\psi_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{A.1}$$

Notons que dans cette expression, la matrice de permutation $\mathrm{PERM}_{Z\leftarrow X,\ -X\leftarrow Y,\ -Y\leftarrow Z}$ permet la conversion dans la convention utilisée en vision artificielle, tandis que les matrices $\mathrm{R}_X(\phi_i)$, $\mathrm{R}_Y(\theta_i)$ et $\mathrm{R}_Z(\psi_i)$ représentent la rotation dans la convention d'Euler ZYX.

Nous souhaitons obtenir l'expression de la matrice de recalage $\mathrm{H}^f_{2\leftarrow 1}$, qui lie les coordonnées d'un pixel (u_1, v_1) dans l'image I_1 aux coordonnées du pixel (u_2, v_2) dans l'image I_2 de la façon suivante :

$$\begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} \propto \mathrm{H}^f_{2\leftarrow 1} \cdot \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \\ \omega_4 & \omega_5 & \omega_6 \\ \omega_7 & \omega_8 & \omega_9 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} \propto \begin{bmatrix} \frac{\omega_1}{\omega_9} & \frac{\omega_2}{\omega_9} & \frac{\omega_3}{\omega_9} \\ \frac{\omega_4}{\omega_9} & \frac{\omega_5}{\omega_9} & \frac{\omega_6}{\omega_9} \\ \frac{\omega_7}{\omega_9} & \frac{\omega_8}{\omega_9} & 1 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} \tag{A.2}$$

Notons ici que la matrice de recalage $\mathrm{H}^f_{2\leftarrow 1}$ est définie à une mise à l'échelle près, du fait de l'utilisation des coordonnées homogènes [52]. Pour permettre une comparaison non-ambigüe entre deux matrices de recalages, il est donc préférable d'utiliser la représentation canonique de l'homographie, qui est telle que l'élément en position $(3,3)$ est égal à 1. L'utilisation de cette représentation canonique introduit alors une division des autres éléments par ω_9.

Pour obtenir l'expression de $\mathrm{H}^f_{2\leftarrow 1}$, intéressons nous à l'expression du point $\mathbf{M}_0$ appartenant au plan π et se projetant dans l'image I_1 aux coordonnées image (u_1, v_1). La direction $\mathbf{V}r_1$ du rayon lumineux issu du pixel (u_1, v_1) de la caméra $\mathcal{C}_1$ est donnée par l'expression :

$$\mathbf{V}r_1 = \frac{u_1 - ox_1}{fx_1} \cdot \mathbf{V}x_1 + \frac{v_1 - oy_1}{fy_1} \cdot \mathbf{V}y_1 + \mathbf{V}z_1 \tag{A.3}$$

En combinant la contrainte $\pi(\mathbf{M}_0) = 0$, représentant le fait que le point $\mathbf{M}_0$ appartient au plan π, avec la nécessaire colinéarité entre les vecteurs $\mathbf{V}r_1$ et $\mathbf{M}_0 - \mathbf{C}_1$, nous trouvons l'expression suivante pour $\mathbf{M}_0$:

$$\mathbf{M}_0 = \mathbf{C}_1 - \frac{\pi(\mathbf{C}_1)}{\pi(\mathbf{V}r_1) + d_\pi} \cdot \mathbf{V}r_1 = \frac{[\pi(\mathbf{V}r_1) + d_\pi] \cdot \mathbf{C}_1 - \pi(\mathbf{C}_1) \cdot \mathbf{V}r_1}{\pi(\mathbf{V}r_1) + d_\pi} \tag{A.4}$$

Nous pouvons alors exprimer le vecteur $\mathbf{M}_0 - \mathbf{C}_2$ en fonction des paramètres de $\mathcal{C}_1$:

$$\mathbf{M}_0 - \mathbf{C}_2 = \frac{1}{Z} \cdot (u_1 \cdot \mathbf{V}u_1 + v_1 \cdot \mathbf{V}v_1 + \mathbf{V}c_1) \tag{A.5}$$

où
$$\begin{aligned}
Z &= \pi(\mathbf{V}r_1) + d_\pi \\
\mathbf{V}u_1 &= \frac{-1}{fx_1} \cdot [\langle \mathbf{n}_\pi | \mathbf{V}x_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}x_1] \\
\mathbf{V}v_1 &= \frac{-1}{fy_1} \cdot [\langle \mathbf{n}_\pi | \mathbf{V}y_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}y_1] \\
\mathbf{V}c_1 &= -[\langle \mathbf{n}_\pi | \mathbf{V}w_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}w_1] \\
\mathbf{V}w_1 &= \mathbf{V}z_1 - \frac{ox_1}{fx_1} \cdot \mathbf{V}x_1 - \frac{oy_1}{fy_1} \cdot \mathbf{V}y_1 \\
\mathbf{V}_B &= \mathbf{C}_2 - \mathbf{C}_1
\end{aligned}$$

Pour finir, la contrainte selon laquelle le point $\mathbf{M}_0$ se projette dans l'image I_2 aux coordonnées (u_2, v_2) donne la relation suivante :

$$\begin{aligned}
u_2 &= ox_2 + fx_2 \cdot \frac{\langle \mathbf{V}x_2 | \mathbf{M}_0 - \mathbf{C}_2 \rangle}{\langle \mathbf{V}z_2 | \mathbf{M}_0 - \mathbf{C}_2 \rangle} \\
&= \frac{\langle ox_2 \cdot \mathbf{V}z_2 + fx_2 \cdot \mathbf{V}x_2 | u_1 \cdot \mathbf{V}u_1 + v_1 \cdot \mathbf{V}v_1 + \mathbf{V}c_1 \rangle}{\langle \mathbf{V}z_2 | u_1 \cdot \mathbf{V}u_1 + v_1 \cdot \mathbf{V}v_1 + \mathbf{V}c_1 \rangle} \\
&= \frac{\omega_1 \cdot u_1 + \omega_2 \cdot v_1 + \omega_3}{\omega_7 \cdot u_1 + \omega_8 \cdot v_1 + \omega_9}
\end{aligned} \tag{A.6}$$

$$\begin{aligned}
v_2 &= oy_2 + fy_2 \cdot \frac{\langle \mathbf{V}y_2 | \mathbf{M}_0 - \mathbf{C}_2 \rangle}{\langle \mathbf{V}z_2 | \mathbf{M}_0 - \mathbf{C}_2 \rangle} \\
&= \frac{\langle oy_2 \cdot \mathbf{V}z_2 + fy_2 \cdot \mathbf{V}y_2 | u_1 \cdot \mathbf{V}u_1 + v_1 \cdot \mathbf{V}v_1 + \mathbf{V}c_1 \rangle}{\langle \mathbf{V}z_2 | u_1 \cdot \mathbf{V}u_1 + v_1 \cdot \mathbf{V}v_1 + \mathbf{V}c_1 \rangle} \\
&= \frac{\omega_4 \cdot u_1 + \omega_5 \cdot v_1 + \omega_6}{\omega_7 \cdot u_1 + \omega_8 \cdot v_1 + \omega_9}
\end{aligned} \tag{A.7}$$

Il est alors possible d'identifier les éléments $\{\omega_k\}_{k \in [\![1,9]\!]}$ de la matrice $\mathrm{H}^f_{2 \leftarrow 1}$, débouchant sur les expressions analytiques suivantes :

$$\begin{aligned}
\omega_1 &= \langle ox_2 \cdot \mathbf{V}z_2 + fx_2 \cdot \mathbf{V}x_2 | \mathbf{V}u_1 \rangle &&= -\tfrac{1}{fx_1} \cdot \langle ox_2 \cdot \mathbf{V}z_2 + fx_2 \cdot \mathbf{V}x_2 | \langle \mathbf{n}_\pi | \mathbf{V}x_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}x_1 \rangle \\
\omega_2 &= \langle ox_2 \cdot \mathbf{V}z_2 + fx_2 \cdot \mathbf{V}x_2 | \mathbf{V}v_1 \rangle &&= -\tfrac{1}{fy_1} \cdot \langle ox_2 \cdot \mathbf{V}z_2 + fx_2 \cdot \mathbf{V}x_2 | \langle \mathbf{n}_\pi | \mathbf{V}y_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}y_1 \rangle \\
\omega_3 &= \langle ox_2 \cdot \mathbf{V}z_2 + fx_2 \cdot \mathbf{V}x_2 | \mathbf{V}c_1 \rangle &&= -\langle ox_2 \cdot \mathbf{V}z_2 + fx_2 \cdot \mathbf{V}x_2 | \langle \mathbf{n}_\pi | \mathbf{V}w_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}w_1 \rangle \\
\omega_4 &= \langle oy_2 \cdot \mathbf{V}z_2 + fy_2 \cdot \mathbf{V}y_2 | \mathbf{V}u_1 \rangle &&= -\tfrac{1}{fx_1} \cdot \langle oy_2 \cdot \mathbf{V}z_2 + fy_2 \cdot \mathbf{V}y_2 | \langle \mathbf{n}_\pi | \mathbf{V}x_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}x_1 \rangle \\
\omega_5 &= \langle oy_2 \cdot \mathbf{V}z_2 + fy_2 \cdot \mathbf{V}y_2 | \mathbf{V}v_1 \rangle &&= -\tfrac{1}{fy_1} \cdot \langle oy_2 \cdot \mathbf{V}z_2 + fy_2 \cdot \mathbf{V}y_2 | \langle \mathbf{n}_\pi | \mathbf{V}y_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}y_1 \rangle \\
\omega_6 &= \langle oy_2 \cdot \mathbf{V}z_2 + fy_2 \cdot \mathbf{V}y_2 | \mathbf{V}c_1 \rangle &&= -\langle oy_2 \cdot \mathbf{V}z_2 + fy_2 \cdot \mathbf{V}y_2 | \langle \mathbf{n}_\pi | \mathbf{V}w_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}w_1 \rangle \\
\omega_7 &= \langle \mathbf{V}z_2 | \mathbf{V}u_1 \rangle &&= -\tfrac{1}{fx_1} \cdot \langle \mathbf{V}z_2 | \langle \mathbf{n}_\pi | \mathbf{V}x_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}x_1 \rangle \\
\omega_8 &= \langle \mathbf{V}z_2 | \mathbf{V}v_1 \rangle &&= -\tfrac{1}{fy_1} \cdot \langle \mathbf{V}z_2 | \langle \mathbf{n}_\pi | \mathbf{V}y_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}y_1 \rangle \\
\omega_9 &= \langle \mathbf{V}z_2 | \mathbf{V}c_1 \rangle &&= -\langle \mathbf{V}z_2 | \langle \mathbf{n}_\pi | \mathbf{V}w_1 \rangle \cdot \mathbf{V}_B + \pi(\mathbf{C}_1) \cdot \mathbf{V}w_1 \rangle
\end{aligned} \tag{A.8}$$

$$\mathbf{h}_{Id}(\mathbf{x}_1^i,\mathbf{x}_2^i) = \begin{bmatrix} \frac{fx_2}{fx_1} \\ 0 \\ ox_2 - fx_2 \cdot \frac{ox_1}{fx_1} \\ 0 \\ \frac{fy_2}{fy_1} \\ oy_2 - fy_2 \cdot \frac{oy_1}{fy_1} \\ 0 \\ 0 \end{bmatrix}$$

$$\mathrm{J}_H(\mathbf{x}_1,\mathbf{x}_2^i,\mathbf{n}_\pi,d_\pi) = \begin{bmatrix} 0 & -\frac{ox_2+\frac{ox_1}{fx_1}\cdot fx_2}{fx_1} & \frac{oy_1}{fy_1}\cdot\frac{fx_2}{fx_1} & \frac{fx_2}{fx_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle}{\pi(\mathbf{C}_1)} & 0 & \frac{ox_2\cdot\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle - fx_2\cdot\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{fx_1\cdot\pi(\mathbf{C}_1)} \\ -\frac{fx_2}{fy_1} & 0 & \frac{ox_2}{fy_1} & \frac{fx_2}{fy_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle}{\pi(\mathbf{C}_1)} & 0 & \frac{ox_2}{fy_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle}{\pi(\mathbf{C}_1)} \\ \frac{oy_1}{fy_1}\cdot fx_2 & (1+\frac{ox_1^2}{fx_1^2})\cdot fx_2 & -\frac{ox_1}{fx_1}\cdot\frac{oy_1}{fy_1}\cdot fx_2 & fx_2\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{\pi(\mathbf{C}_1)} & 0 & \frac{ox_1}{fx_1}\cdot fx_2\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{\pi(\mathbf{C}_1)} \\ \frac{fy_2}{fx_1} & -\frac{oy_2}{fx_1} & 0 & 0 & \frac{fy_2}{fx_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle}{\pi(\mathbf{C}_1)} & \frac{oy_2}{fx_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle}{\pi(\mathbf{C}_1)} \\ 0 & -\frac{ox_1}{fx_1}\cdot\frac{fy_2}{fy_1} & \frac{oy_2+\frac{oy_1}{fy_1}\cdot fy_2}{fy_1} & 0 & \frac{fy_2}{fy_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle}{\pi(\mathbf{C}_1)} & \frac{oy_2\cdot\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle - fy_2\cdot\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{fy_1\cdot\pi(\mathbf{C}_1)} \\ -\frac{ox_1}{fx_1}\cdot fy_2 & \frac{ox_1}{fx_1}\cdot\frac{oy_1}{fy_1}\cdot fy_2 & -(1+\frac{oy_1^2}{fy_1^2})\cdot fy_2 & 0 & fy_2\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{\pi(\mathbf{C}_1)} & \frac{oy_1}{fy_1}\cdot fy_2\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{\pi(\mathbf{C}_1)} \\ 0 & -\frac{1}{fx_1} & 0 & 0 & 0 & \frac{1}{fx_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle}{\pi(\mathbf{C}_1)} \\ 0 & 0 & \frac{1}{fy_1} & 0 & 0 & \frac{1}{fy_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle}{\pi(\mathbf{C}_1)} \end{bmatrix}$$

TABLE A.1 – *Expressions analytiques des termes de linéarisation de l'homographie de recalage dans le cas restreint.*

A.1.2 Linéarisation dans le cas restreint

Dans le cas restreint, les paramètres intrinsèques sont supposés connus et seuls les paramètres extrinsèques, c'est-à-dire la position et l'orientation des caméras, sont estimés. La variation de ces paramètres extrinsèques $d\mathbf{x}$ est définie par rapport à la caméra $\mathcal{C}_1$. Ainsi, la variation de position de la caméra $\mathcal{C}_1$ à la caméra $\mathcal{C}_2$ est exclusivement représentée par le vecteur $\mathbf{V}_B = \mathbf{C}_2 - \mathbf{C}_1 = dx\cdot\mathbf{V}x_1 + dy\cdot\mathbf{V}y_1 + dz\cdot\mathbf{V}z_1$, tandis que la variation d'orientation de la caméra $\mathcal{C}_2$ par rapport à la caméra $\mathcal{C}_1$ est représentée par les produits scalaires entre les vecteurs du repère $\mathcal{R}_2$ et ceux du repère $\mathcal{R}_1$. Plus précisément, la matrice de rotation $\mathrm{R}_{2\leftarrow1}$ du repère $\mathcal{R}_1$ vers le repère $\mathcal{R}_2$ s'exprime de la façon suivante :

$$\begin{aligned} \mathrm{R}_{2\leftarrow1}(d\psi,d\theta,d\phi) &= \mathrm{R}_X(d\phi)\cdot\mathrm{R}_Y(d\theta)\cdot\mathrm{R}_Z(d\psi) \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(d\phi) & -sin(d\phi) \\ 0 & sin(d\phi) & cos(d\phi) \end{bmatrix} \cdot \begin{bmatrix} cos(d\theta) & 0 & sin(d\theta) \\ 0 & 1 & 0 \\ -sin(d\theta) & 0 & cos(d\theta) \end{bmatrix} \cdot \begin{bmatrix} cos(d\psi) & -sin(d\psi) & 0 \\ sin(d\psi) & cos(d\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{V}x_2^T \\ \mathbf{V}y_2^T \\ \mathbf{V}z_2^T \end{bmatrix} \cdot \begin{bmatrix} \mathbf{V}x_1 & \mathbf{V}y_1 & \mathbf{V}z_1 \end{bmatrix} \end{aligned} \tag{A.9}$$

Lorsque les angles $d\psi$, $d\theta$ et $d\phi$ sont petits, la linéarisation au premier ordre de la matrice $\mathrm{R}_{2\leftarrow1}(d\psi,d\theta,d\phi)$ est donnée par l'expression suivante :

$$\mathrm{R}_{2\leftarrow1}(d\psi,d\theta,d\phi) = \begin{bmatrix} \mathbf{V}x_2^T \\ \mathbf{V}y_2^T \\ \mathbf{V}z_2^T \end{bmatrix} \cdot \begin{bmatrix} \mathbf{V}x_1 & \mathbf{V}y_1 & \mathbf{V}z_1 \end{bmatrix} \approx \begin{bmatrix} 1 & -d\psi & d\theta \\ d\psi & 1 & -d\phi \\ -d\theta & d\phi & 1 \end{bmatrix} \tag{A.10}$$

Dans la suite, nous utilisons la représentation canonique de la matrice de recalage $\mathrm{H}^f_{2\leftarrow1}$, qui met en jeu les rapports $\{\frac{\omega_k}{\omega_9}\}_{k\in[\![1,8]\!]}$. Nous utilisons également la fonction $\mathrm{vec}(\cdot)$ désignant la transformation d'une matrice 3×3 en un vecteur colonne à 8 dimensions, l'élément d'indice $(3,3)$ étant ignoré. Il est alors possible de déterminer l'expression de $\mathrm{vec}\left(\mathrm{H}^f_{2\leftarrow1}(\mathbf{x}_1,\mathbf{x}_2,\mathbf{n}_\pi,d_\pi)\right)$ linéarisée au premier ordre par rapport à $d\mathbf{x}$:

$$\mathrm{vec}\left(\mathrm{H}^f_{2\leftarrow1}(\mathbf{x}_1,\mathbf{x}_2,\mathbf{n}_\pi,d_\pi)\right) \approx \mathbf{h}_{Id}(\mathbf{x}_1^i,\mathbf{x}_2^i) + \mathrm{J}_H(\mathbf{x}_1,\mathbf{x}_2^i,\mathbf{n}_\pi,d_\pi)\cdot d\mathbf{x} \tag{A.11}$$

$$
\mathrm{J}_H(\mathbf{x}_1,\mathbf{n}_\pi,d_\pi) = \begin{bmatrix}
0 & -2\cdot\frac{ox_1}{fx_1} & \frac{oy_1}{fy_1} & \frac{\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle}{\pi(\mathbf{C}_1)} & 0 & \frac{ox_1\cdot\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle - fx_1\cdot\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{fx_1\cdot\pi(\mathbf{C}_1)} & \frac{1}{fx_1} & 0 & 0 & 0 \\
-\frac{fx_1}{fy_1} & 0 & \frac{ox_1}{fy_1} & \frac{fx_1}{fy_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle}{\pi(\mathbf{C}_1)} & 0 & \frac{ox_1}{fy_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle}{\pi(\mathbf{C}_1)} & 0 & 0 & 0 & 0 \\
\frac{oy_1}{fy_1}\cdot fx_1 & (1+\frac{ox_1^2}{fx_1^2})\cdot fx_1 & -ox_1\cdot\frac{oy_1}{fy_1} & fx_1\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{\pi(\mathbf{C}_1)} & 0 & ox_1\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{\pi(\mathbf{C}_1)} & -\frac{ox_1}{fx_1} & 0 & 1 & 0 \\
\frac{fy_1}{fx_1} & -\frac{oy_1}{fx_1} & 0 & 0 & \frac{fy_1}{fx_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle}{\pi(\mathbf{C}_1)} & \frac{oy_1}{fx_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle}{\pi(\mathbf{C}_1)} & 0 & 0 & 0 & 0 \\
0 & -\frac{ox_1}{fx_1} & 2\cdot\frac{oy_1}{fy_1} & 0 & \frac{\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle}{\pi(\mathbf{C}_1)} & \frac{oy_1\cdot\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle - fy_1\cdot\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{fy_1\cdot\pi(\mathbf{C}_1)} & 0 & \frac{1}{fy_1} & 0 & 0 \\
-\frac{ox_1}{fx_1}\cdot fy_1 & \frac{ox_1}{fx_1}\cdot oy_1 & -(1+\frac{oy_1^2}{fy_1^2})\cdot fy_1 & 0 & fy_1\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{\pi(\mathbf{C}_1)} & oy_1\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}w_1\rangle}{\pi(\mathbf{C}_1)} & 0 & -\frac{oy_1}{fy_1} & 0 & 1 \\
0 & -\frac{1}{fx_1} & 0 & 0 & 0 & \frac{1}{fx_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}x_1\rangle}{\pi(\mathbf{C}_1)} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{fy_1} & 0 & 0 & \frac{1}{fy_1}\cdot\frac{\langle \mathbf{n}_\pi|\mathbf{V}y_1\rangle}{\pi(\mathbf{C}_1)} & 0 & 0 & 0 & 0
\end{bmatrix}
$$

TABLE A.2 – *Expression analytique de la matrice Jacobienne associée à la linéarisation de l'homographie de recalage dans le cas général.*

$$
\begin{aligned}
\text{avec} \qquad \mathbf{h}_{Id}(\mathbf{x}_1^i,\mathbf{x}_2^i) &= \mathrm{vec}\left(\mathrm{H}^f_{2\leftarrow 1}(\mathbf{x}_1,\mathbf{x}_2,\mathbf{n}_\pi,d_\pi)\right)\Big|_{\mathbf{x}_2^e=\mathbf{x}_1^e} \\
\mathrm{J}_H(\mathbf{x}_1,\mathbf{x}_2^i,\mathbf{n}_\pi,d_\pi) &= \left[j_H^{(k,l)}(\mathbf{x}_1,\mathbf{x}_2^i,\mathbf{n}_\pi,d_\pi)\right]_{k\in[\![1,8]\!],\ l\in[\![1,6]\!]} \\
j_H^{(k,l)}(\mathbf{x}_1,\mathbf{x}_2^i,\mathbf{n}_\pi,d_\pi) &= \frac{\partial(\frac{\omega_k}{\omega_9})}{\partial(d\mathbf{x}_l)}\Big|_{\mathbf{x}_2^e=\mathbf{x}_1^e} = \frac{\frac{\partial\omega_k}{\partial(d\mathbf{x}_l)}\cdot\omega_9 - \omega_k\cdot\frac{\partial\omega_9}{\partial(d\mathbf{x}_l)})}{\omega_9^2}\Big|_{\mathbf{x}_2^e=\mathbf{x}_1^e}
\end{aligned}
$$

Les expressions analytiques des termes $\mathbf{h}_{Id}(\mathbf{x}_1^i,\mathbf{x}_2^i)$ et $\mathrm{J}_H(\mathbf{x}_1,\mathbf{x}_2^i,\mathbf{n}_\pi,d_\pi)$ sont données à la table A.1 en fonction des différents paramètres du problème.

A.1.3 Linéarisation dans le cas général

Dans le cas général, nous nous intéressons à l'estimation conjointe des paramètres extrinsèques et des paramètres intrinsèques. Nous définissons alors la variation des paramètres d'acquisition $d\mathbf{x}$ comme dans le cas restreint, en ajoutant en plus les variations de paramètres de calibration. Ainsi, la variation de position est représentée par rapport à la caméra $\mathcal{C}_1$ par le vecteur $\mathbf{V}_B = \mathbf{C}_2 - \mathbf{C}_1 = dx\cdot\mathbf{V}x_1 + dy\cdot\mathbf{V}y_1 + dz\cdot\mathbf{V}z_1$ et la variation d'orientation est représentée par la matrice $\mathrm{R}_{2\leftarrow 1}$ définie à l'équation A.9. De plus, nous ajoutons à ces variations de paramètres extrinsèques les variations des quatre paramètres intrinsèques de la façon suivante :

$$
d\mathbf{x} = \begin{bmatrix} d\psi & d\theta & d\phi & dx & dy & dz & dfx & dfy & dox & doy \end{bmatrix}^T \tag{A.12}
$$

$$
\begin{aligned}
\text{avec} \qquad dfx &= fx_2 - fx_1 \\
dfy &= fy_2 - fy_1 \\
dox &= ox_2 - ox_1 \\
doy &= oy_2 - oy_1
\end{aligned}
$$

De la même manière que pour le cas restreint, nous souhaitons déterminer l'expression de $\mathrm{vec}\left(\mathrm{H}^f_{2\leftarrow 1}(\mathbf{x}_1,\mathbf{x}_2,\mathbf{n}_\pi,d_\pi)\right)$ linéarisée au premier ordre par rapport à $d\mathbf{x}$:

$$
\mathrm{vec}\left(\mathrm{H}^f_{2\leftarrow 1}(\mathbf{x}_1,\mathbf{x}_2,\mathbf{n}_\pi,d_\pi)\right) \approx \mathrm{vec}\left(\mathrm{ID}\right) + \mathrm{J}_H(\mathbf{x}_1,\mathbf{n}_\pi,d_\pi)\cdot d\mathbf{x} \tag{A.13}
$$

$$
\begin{aligned}
\text{avec} \qquad \mathrm{J}_H(\mathbf{x}_1,\mathbf{n}_\pi,d_\pi) &= \left[j_H^{(k,l)}(\mathbf{x}_1,\mathbf{n}_\pi,d_\pi)\right]_{k\in[\![1,8]\!],\ l\in[\![1,10]\!]} \\
j_H^{(k,l)}(\mathbf{x}_1,\mathbf{n}_\pi,d_\pi) &= \frac{\partial(\frac{\omega_k}{\omega_9})}{\partial(d\mathbf{x}_l)}\Big|_{\mathbf{x}_2=\mathbf{x}_1} = \frac{\frac{\partial\omega_k}{\partial(d\mathbf{x}_l)}\cdot\omega_9 - \omega_k\cdot\frac{\partial\omega_9}{\partial(d\mathbf{x}_l)})}{\omega_9^2}\Big|_{\mathbf{x}_2=\mathbf{x}_1}
\end{aligned}
$$

L'expression analytique de $\mathrm{J}_H(\mathbf{x}_1,\mathbf{n}_\pi,d_\pi)$ est donnée à la table A.2 en fonction des différents paramètres du problème.

A.2 Expression de la projection invariante à l'illumination

La méthode introduite par Finlayson et al. [44] permet de convertir une observation en couleurs (R,G,B) vers un espace uni-dimensionnel invariant par rapport à l'illumination. Cette conversion est effectuée grâce à une projection, opérant sur le vecteur des coordonnées chromatiques logarithmiques (χ_R,χ_G,χ_B) et pouvant être caractérisée à l'aide d'un unique paramètre angulaire. Dans cette section, nous formulons l'expression de cette projection en fonction de ce paramètre angulaire.

Remarquons d'abord que les vecteurs représentant les coordonnées chromatiques logarithmiques (χ_R,χ_G,χ_B) appartiennent tous au plan κ d'équation $\chi_R+\chi_G+\chi_B=0$, qui passe par l'origine et est orthogonal au vecteur $\mathbf{V}_u=(1,1,1)$, puisque :

$$\chi_R+\chi_G+\chi_B = log(\frac{R}{\sqrt[3]{R\cdot G\cdot B}})+log(\frac{G}{\sqrt[3]{R\cdot G\cdot B}})+log(\frac{B}{\sqrt[3]{R\cdot G\cdot B}}) = log(\frac{R\cdot G\cdot B}{R\cdot G\cdot B}) = 0 \qquad \text{(A.14)}$$

Soit $(\mathbf{V}_{\chi_R},\mathbf{V}_{\chi_G},\mathbf{V}_{\chi_B})$ la base canonique de l'espace des coordonnées chromatiques logarithmiques. Déterminons une deuxième base $(\mathbf{V}_u,\mathbf{V}_v,\mathbf{V}_w)$, composée du vecteur $\mathbf{V}_u$ orthogonal au plan κ, et de deux vecteurs $(\mathbf{V}_v,\mathbf{V}_w)$ formant une base du plan κ. Soit $\mathbf{V}_v$ le projeté orthogonal de $\mathbf{V}_{\chi_B}$ sur le plan κ. Le symbole $\times$ représentant le produit vectoriel, il peut être démontré aisément que :

$$\mathbf{V}_v = \frac{2}{\sqrt{6}}\cdot\mathbf{V}_{\chi_B}-\frac{1}{\sqrt{6}}\cdot\mathbf{V}_{\chi_R}-\frac{1}{\sqrt{6}}\cdot\mathbf{V}_{\chi_G} \qquad \text{(A.15)}$$

$$\mathbf{V}_w = \mathbf{V}_u\times\mathbf{V}_v = \frac{1}{\sqrt{2}}\cdot\mathbf{V}_{\chi_R}-\frac{1}{\sqrt{2}}\cdot\mathbf{V}_{\chi_G} \qquad \text{(A.16)}$$

Considérons le vecteur $\mathbf{V}_{illum}(\xi)=cos(\xi)\cdot\mathbf{V}_v+sin(\xi)\cdot\mathbf{V}_w$. Désignons par $\pi(\xi)$ le plan orthogonal à $\mathbf{V}_{illum}(\xi)$ et passant par l'origine. Alors la projection orthogonale, sur le plan $\pi(\xi)$, d'un vecteur $\boldsymbol{\chi}=(\chi_R,\chi_G,\chi_B)$ appartenant au plan κ peut s'exprimer de la façon suivante :

$$\mathbf{h}_\pi(\xi) = \boldsymbol{\chi}-\mathbf{V}_{illum}(\xi)\cdot\mathbf{V}_{illum}(\xi)^T\cdot\boldsymbol{\chi} = \mathrm{P}_{illum}(\xi)\cdot\boldsymbol{\chi} \qquad \text{(A.17)}$$

$$\text{avec} \qquad \mathrm{P}_{illum}(\xi) = \mathrm{ID}-\mathbf{V}_{illum}(\xi)\cdot\mathbf{V}_{illum}(\xi)^T$$

Il a été démontré par Finlayson et al. [44] que sous certaines conditions, les coordonnées chromatiques logarithmiques (χ_R,χ_G,χ_B) associées à un pixel d'une image donnée obéissent à la relation suivante :

$$(\chi_R,\chi_G,\chi_B) = \left(log(\frac{R}{\sqrt[3]{R\cdot G\cdot B}}),log(\frac{G}{\sqrt[3]{R\cdot G\cdot B}}),log(\frac{B}{\sqrt[3]{R\cdot G\cdot B}})\right) = \boldsymbol{\rho}+\frac{s}{T}\cdot\mathbf{V}_{illum}(\xi) \qquad \text{(A.18)}$$

pour une valeur donnée de ξ, liée à la caméra utilisée pour acquérir l'image considérée. Dans cette relation, $\boldsymbol{\rho}$ est un vecteur dépendant de la caméra et de la réflectance de la surface observée, s est un scalaire dépendant de la caméra, et T représente la température de corps noir de la source lumineuse éclairant la surface observée. Ainsi, s'il est possible de déterminer la valeur ξ caractérisant la direction de variation des coordonnées chromatiques logarithmiques en fonction de l'illumination, alors la matrice de projection $\mathrm{P}_{illum}(\xi)$ permet d'éliminer la composante d'illumination. En pratique, la valeur du paramètre ξ correspondant à la caméra utilisée peut être obtenue par la méthode de calibration décrite par Finlayson et al. [44].

English synthesis

THIS appendix provides an English synthesis of the work carried out during this PhD. In order to limit the size of this synthesis, the demonstrations, figures and tables will not be repeated here: they will only be discussed and links to their correspondents in the main document will be given.

This synthesis is organized as follows. Section B.1 discusses the operational difficulties encountered in the task of video analysis in aerial videos. A brief introduction to the problem of change detection is then provided, and motivations are given to justify our general change detection approach for aerial videos with arbitrary trajectories. Section B.2 then presents the various pre-processing algorithms which we developed in the context of this PhD. These algorithms correspond to two kinds of pre-processing operations: geo-localization and illumination attenuation. Next, section B.3 describes the algorithms related to our change detection approach. This approach is based on 3D appearance modeling, which consists of an offline 3D model generation from the reference videos, followed by an online change detection on the test video. Finally, section B.4 presents several consolidation algorithms, which enable the use of prior knowledge in order to improve the accuracy of the results. As a generalization of this idea, we also introduce a relevance feedback mechanism, allowing the user to provide annotations in order to adjust the change detection results to match his own needs. The good performance of the proposed algorithms is demonstrated, in the associated sections, using both real and synthetical data.

Contents

B.1 Introduction

B.1.1 Operational context

Over the last decade, video monitoring has received a growing interest in many remote sensing and computer vision applications [46, 64, 68], such as site or street surveillance, resource monitoring from aerial platforms, survivors rescue, military intelligence and so on. Therefore, video analysis, which is still supported for the largest part by human operators, has become an essential task for numerous applications. In parallel, video devices have become cheaper and their frame-rates, resolutions and memory capacities have dramatically increased.

Hence, technological progress, combined with the increasing number of acquisition devices deployed in operational scenarios, has led to an explosion of the volume of data being acquired. As a consequence, nowadays, human operators in charge of video analysis often have to stay focused for extensive time periods, while paying close attention to multiple video streams in parallel. Moreover in most applications, the target events or objects, which the operators are trying to identify, are often unusual and therefore happen or appear very rarely in the video streams (e.g. breaking in into private site, burglary or aggression, avalanche survivor, etc). Hence, video analysis requires a continuous concentration even though most of the data is uninteresting, which makes the analysis of large data volumes very demanding.

Analysis difficulties Besides the difficulties resulting from the large data volumes, the task of video analysis itself is not an obvious one. Indeed, the good understanding of the observed scene may be disturbed by the poor quality of the images from the video streams. This poor image quality usually lead to considerable variations of the appearance of objects in the scene, which might be due for instance to image noise, to the low dynamic range of the sensor which causes under or over-exposed observations, to illumination effects, and so on. When considering moving cameras, in particular in aerial observation, these appearance variations become even more significant, due to various causes such as motion blur or meteorological conditions (e.g. rain, fog, wind, etc). Moreover, aerial acquisition often suffer from a lack of context, which result from the narrowness of the field of view (referred to as the *soda straw effect* [64]). This lack of context lead to a difficulty to perceive the relative position of objects in the observed scene and can also disturb the good understanding of the observations. Finally, the frequent and considerable viewpoint variations encountered in aerial observation may also be an obstacle to the task of video analysis, as they make data comparison difficult.

These difficulties have two consequences in practice. Firstly, they can lead to erroneous analysis results, which may have serious repercussions depending on the stakes associated with each operational application. Secondly, as the thorough analysis of the large amount of acquired data is unrealistic, it is common for the largest part of this data to be simply stored and finally erased without being processed.

Interest of a semi-automatic approach Following the mention of the difficulties encountered with video analysis, one may wonder how to improve the effectiveness and efficiency of operators, or in other words, how to maximize the amount of information extracted from the available acquisitions, in order to use them to their full potential. For that purpose, the solution consisting of a fully automatic system seems unrealistic, due to the limited maturity of automatic image understanding.

Hence, it seems more promising to employ a semi-automatic approach, which would occasionally request user input in order to improve the reliability of the results. Such semi-automatic approaches combine the benefits from automatic processing, able to execute tedious tasks quickly and systematically, with the benefits from human analysis, able of a great accuracy in classification tasks. Therefore, they are very well suited to the problem of analyzing large

volumes of video data, since this requires fast exclusion of the largest part of the data, which is uninteresting, but also accurate detection of targeted events or objects.

Moreover, in many applications (e.g. detection of breaking-in, detection of improvised explosive device, etc), the distinction between interesting and uninteresting data can be made by detecting the presence or absence of significant changes with respect to a given reference. Hence, the problem of comparing several acquisitions, or more generally the problem of change detection, which represents a low-level semantic task (i.e. detection of generic changes), constitute a pre-requisite to higher-level semantic tasks (e.g. understanding, classification of changes). Hence, the problem of change detection has a considerable interest in the field of video analysis.

B.1.2 Change detection problem

The term of *change detection* has several meanings in the scientific literature, hence it is important to define what we mean by this in this PhD thesis. We define the problem of change detection as the task aiming at detecting significant changes by comparing two or more radiometric acquisitions, which were acquired at different dates. This sentence emphasizes three important points, which help to define the scope of our problem with respect to other close or connected problems. Firstly, we are interested in the comparison of acquisitions acquired at different dates, on the contrary to problems aiming at the analysis of a single acquisition (e.g. abrupt behavior change [4], moving object tracking [119], background subtraction [102], etc). Secondly, we are interested in the comparison of radiometric acquisitions (i.e. measurements of luminous information), therefore composed of continuous values even though they might be quantified, on the contrary to problems which involves discreet data obtained from a prior classification (e.g. detection of category changes [73]). Finally, we are interested in the detection of significant changes, which introduces a subjectivity related to the targeted application, on the contrary to problems interested to all forms of changes (e.g. video compression [9]) or to very subtle changes (e.g. detection of steganography in images [70]).

The following presents in more details the problem of change detection and discusses the assumptions taken in this PhD thesis to address this problem in the context of aerial videos.

Description of the core problem As a simplification, the problem of change detection can be related to the popular game called ”spot the differences“, whose objective is to identify the differences between two images. The goal of change detection is very similar, but this more general problem addresses the general case where pixel-wise alignment between the images cannot be assumed and where the differences due to the targeted changes are not the only ones.

More formally, the problem of change detection [95] consists of comparing a new acquisition, referred to as the *test* acquisition, with respect to some *reference* data. The objective of this comparison is two-fold. Firstly, this enables detecting the presence or absence of changes between the test and reference data. Secondly, this may also aim at the localization of these changes in the acquisitions. Such a localization is obtained through the estimation of the *change mask*, which associates to each element of the test acquisition (e.g. pixel, group of pixel) a binary label indicating if this element is irrelevant or of possible interest. Both tasks were addressed during this PhD and in the following, the term of change detection will design both of them.

In practice, the problem of change detection consists of several sub-problems. The main one involves the actual data comparison, which aims at the identification of changes of possible interest. Another important sub-problem, when the acquisitions are captured from different viewpoints, involves the task of geometric alignment, which aims at finding areas representing the same content in test and reference data. Moreover, in the case where the test and/or reference data are composed of several acquisitions (e.g. videos, time series, etc), two additional problems may arise. The first one may be referred to as the formation of a reference, which aims at finding or generating an optimal reference for the comparison with a given test acquisition. The

second one involves the combination of the results obtained on successive test acquisitions, which are redundant and therefore include a latent consistency. Hence, using this consistency may benefit to the overall accuracy and/or computational load. A detailed discussion of these sub-problems, along with a review of the state of the art, has been provided in chapter 2 (in French). Figures 2.2, 2.4, 2.6, 2.7, 2.8 and 2.9 summarize the different approaches used in the scientific literature for each sub-problem of change detection. You can also refer to [18] for a brief literature review in English.

Moreover, the techniques used for the estimation of the change masks may benefit from different opportunities depending on the nature of the data being considered. For instance, a reference and test images are more informative for the change detection problem than a reference and test pixels and, therefore, a higher accuracy may be obtained if the spatial structure is used correctly [2, 84]. Similarly, working with videos provides a temporal dimension in addition to the two spatial dimensions available when working on simple images, which must enable an improvement of the detection accuracy. This point will be demonstrated in section B.4.1.2.

Since using the largest amount of available information lead to the best change detection accuracy, as many reference acquisitions as possible should be used. However, this raises the question of how to address changes which appeared within the reference acquisitions. Indeed, when the considered reference observations are acquired at different times (e.g. frames of a video, images of a time serie, etc), changes may have occurred in the observed scene and may therefore exist between the different reference acquisitions. In this PhD, we chose to consider any change existing in the reference acquisitions (e.g. illumination variations, moving vehicles on a road, etc) as irrelevant to the problem of change detection. This choice is justified by the fact that changes within the reference data indicates geographical areas whose normal appearance is varying frequently. As a consequence, similar appearance variations on these geographical areas in the test data are probably of no interest to the image analyst.

This shows that the relevancy of a given change is quite difficult to define. The following provides a detailed discussion on this topic.

Categories of changes Besides the resolution of the different sub-problems related to the problem of change detection, the main difficulty is related to the fact that any two views of a same scene may contain a large number of differences. Figure 1.1 illustrates this difficulty with two aerial images taken 45 days apart and from different viewpoints. It is fairly easy to see that both represent the same scene. However, the difference image, obtained after a simple registration and a thresholded difference, demonstrates that many differences exist between the intensities of corresponding pixels. Hence, the task of change detection is to identify significant changes while ignoring irrelevant ones.

This notion of irrelevant changes, which will be referred to in the following as *variations* or *variability*, may depend on the targeted application. However, many types of variations are common to most applications. Firstly, such variations may be due to uninteresting changes in the observed scene, such as illumination conditions, direction or intensity changes for shadows, repetitive movements in the scene (e.g. tree leaves, water, etc), seasonal variations, and so on. Secondly, variations may also occur during the image acquisition process, such as geometrical effects due to viewpoint variations, lens flares, acquisition noise, transmission (e.g. packet drops) or compression artifacts, etc. These irrelevant variations are usually of no interest in applications using change detection and they should therefore be ignored by the detection. Nevertheless, despite their irrelevance, those variations may still result in extreme visual differences, which makes them difficult to handle automatically.

The notion of significant or relevant change also depend on the targeted application. For instance, detection of changes in the appearance of vegetation, of moving vehicles or of pedestrians may be of interest in such applications as crops or forests monitoring, aerial video-surveillance or search and rescue. In this PhD, we first studied the change detection prob-

lem in the general case and we then focused our work on a more accurate definition of the kind of changes to be detected. Indeed, having a clear definition of the targeted changes is interesting because this enables the integration, in the detection algorithm, of prior knowledge related to their characteristics, which may lead to a significant improvement of accuracy (see section B.4). Hence, after studying a general change detection approach, we focused on the detection of changes corresponding to stationary artificial structures or objects (e.g. buildings, fields, parked vehicles, stationary pedestrians, etc). This restriction to artificial structures or objects is interesting because they usually have clear and well-defined boundaries in the images, which makes them easier to distinguish from other irrelevant variations (e.g. illumination effects). Moreover, in the context of change detection in aerial acquisitions, the changes to be detected are usually stationary, on the contrary to moving objects, which can be monitored and tracked more effectively using other techniques. Of course, these restrictions necessarily narrow the range of direct applications for the proposed change detection approach, however, these applications remain both numerous and diverse (e.g. crop monitoring, cartography or observation database update, search and rescue, situation awareness after natural catastrophes, surveillance of terrorist activities, etc).

Working hypotheses The various topics mentioned above make change detection between several aerial videos a broad and challenging problem. In order to address it, we used a few working hypotheses in order to restrain the range of difficulties. Hence, we assumed that the videos to be compared are acquired within a short time period, which, as mentioned above, lead to less intense and less numerous irrelevant variations in the observed scene. Furthermore, we assume that the videos to be compared correspond to similar ground resolutions, which ensures that the acquisitions are composed of comparable observations.

As a more concrete example, this PhD thesis considers an operational scenario involving a generic aerial observation platform (e.g. airplane, helicopter, Unmanned Aerial Vehicle, aerostat balloon, etc). This platform makes frequent acquisitions over a given geographical area, with a time delay ranging from a few hours to a few days. The ground resolution of each acquisition is approximately the same, but the trajectories or viewpoints are arbitrary.

B.1.3 Motivations and contributions

Motivations Inspection of the state of the art in change detection (see chapter 2, in French) leads to several important conclusions, which shape the design of a change detection framework for aerial videos acquired from cameras moving along arbitrary trajectories.

Incremental and asymmetrical modeling Firstly, although many approaches for appearance modeling exist, only very few of them are adapted to the specific properties of our application. Indeed, as input videos may be huge, the algorithm cannot keep all data points in memory and only compute the appearance model once all observations are available. As a consequence, the algorithm should (i) update appearance models incrementally, (ii) implement a compressed representation of the observed data and (iii) enable asymmetric learning of appearances. This last point relates to the fact that the change detection algorithm should be able to distinguish between two classes, namely *changed* and *unchanged* pixels, even though no sample from the *changed* class was available during the training phase. These three properties are very close to the constraints met in the field of background subtraction, making the associated appearance modeling algorithms well suited to our application. However, pixel-wise appearance models, such as Gaussian Mixture Models [34, 91, 102] or pixel-wise codewords [61], may be ineffective at detecting changes involving groups of pixels, referred to as structural changes in the following, especially in the context of severely varying appearances (see section B.3.2.2 for more details). Moreover, the number of images (several thousands of video frames) is very

small with respect to the dimension of the considered feature space (several hundreds of thousands dimensions). As a consequence, our framework includes a global appearance modeling approach based on the Incremental PCA algorithm presented in [67], which is able to detect both intensity and structural changes, while being robust to the small number of learning samples compared to the dimensionality of the problem.

Redundancy exploitation Secondly, video data is highly redundant and leads to a considerable overlapping between the successive images. Exploitation of this overlapping is both necessary and beneficial as it helps improving detection accuracy. Indeed, on the one hand, ignoring the overlapping between reference video frames, particularly with the problem of the formation of a reference, can lead to a slow and inefficient detection framework. Conversely, exploitation of this overlapping may enable most of the computation to be done offline, making the online processing of a given test image much more efficient. Moreover, the techniques developed in the context of background subtraction [59, 61, 67, 102] showed that exploitation of this redundancy, via appearance modeling, led to the improvement of accuracy when detecting foreground objects. The same conclusion applies to approaches addressing change detection in videos, making appearance modeling approaches good candidates for the exploitation of redundancy in reference videos, provided that they are adapted to the context of change detection (e.g. progressive forgetting of old data should be avoided). Furthermore, redundancy is also present in test videos and its exploitation similarly leads to an improvement of the detection performance [16, 120]. Temporal consolidation of change detection results is therefore included in our framework.

Tri-dimensional representation Thirdly, every technique of appearance modeling requires a form of scene representation in order to organize the appearance models. For that purpose, approaches based on a tri-dimensional representation [24, 32, 34, 91] are better suited than 2D ones [14, 25, 63, 111] to our objective of detecting changes in aerial videos acquired from arbitrary trajectories. Indeed, 3D approaches enable a simpler and more accurate handling of geometrical effects (including parallax, occlusions and so on), which are challenging issues for 2D approaches. This results in a more accurate modeling of reference appearances, therefore improving the accuracy of change detection. Furthermore, a 3D representation of the scene observed in a given reference video may be based on a structure which is independent of this video. Consequently, this representation may be more easily used for the merging or the joint exploitation of several reference acquisitions, possibly heterogeneous (e.g. several aerial videos and/or satellite images). On the contrary, 2D approaches, for instance those using a mosaic representation [63, 79], are usually based on structures which are dependent on the reference data and may therefore be more cumbersome to reuse with additional data. However, a significant drawback of 3D approaches is that they require an accurate knowledge of the relief in the scene. In order to alleviate this problem, our change detection framework uses a constrained form of 3D model implemented as a Height Map, which provides satisfactory results even when the knowledge of relief is inaccurate. More precisely, we show in section B.3.1.3 that, in regions where heights are small with respect to the distance from the aircraft to the scene, namely in most rural and light-urban regions, information from a Digital Terrain Model is sufficient to provide good change detection accuracy.

Handling of illumination variations Fourthly, appropriate handling of illumination effects is a challenging issue. Generative approaches [90, 93], which try to simulate illumination conditions, are often limited by the lack of realism of the simulated effects or by the complexity of the models involved. Approaches based on the detection and exclusion of illumination effects [11, 59, 71, 82, 112, 116] lack flexibility and usually cannot cope with multiple kinds of effects (e.g. hard shadows, soft shadows, highlights, etc). Finally, due to the complex convolution of

illumination with scene-related information, approaches exclusively attempting a direct attenuation of illumination [38, 44, 47] necessarily make a trade-off between false-alarms, generated by illumination variations, and non-detections, caused by wrong attenuation of targeted changes in the scene. In order to address this issue, we use a combination of several mechanisms. Firstly, we use a direct attenuation technique, such as [38], in order to filter the largest part of the illumination variations. Secondly, we use a learning technique, such as incremental PCA [67], which uses acquisitions acquired under various illumination conditions to learn what irrelevant changes caused by illumination look like. Thirdly, we estimate the change mask using a binarization method based on the notion of Maximally Stable Extremal Regions (*MSER*, [75]), which enables a finer analysis of the continuous detection scores than the more frequently used thresholding approach [34]. Finally, we use a relevance feedback approach in order to allow the user to eliminate any residual false alarm.

Semi-automatic approach Semi-automatic approaches [101, 115], on the contrary to fully automatic approaches, occasionally request user input in order to improve robustness. For a data analysis assistance system, this kind of approach is appropriate since the user already takes part in the analysis. However, keeping such a system lightweight and practical is important for this system to be suited to operational use.

Contributions The technical contributions will be described in more details in the following sections. However for clarity, they are also listed below:

- Design and development of a semi-automatic change detection framework, able to perform the offline analysis of several reference videos and the online detection of changes in a given test video. This framework, which was developed in a modular way, enables the comparison of various algorithms for a given operation and the analysis of the impact of each operation on the global change detection accuracy.
- Introduction of a new type of 3D model combining a Quad-Tree structure, whose resolution adapts to the available observations, with a height map, which stores the third spatial dimension [1]. This kind of 3D model is more appropriate than an Octree to the problem of change detection in aerial acquisitions, since it enables a good robustness to viewpoint extrapolation and a lower memory load, which allows large scenes to be modeled.
- Design and development of a semi-automatic method for the offline estimation of the acquisition parameters associated with the reference videos. This method is based on manual annotations for a few key-frames from the videos and on the automatic propagation of these annotations to the remaining video frames. This propagation is done by automatically estimating the trifocal tensor defined between two key-frames and a given intermediate frame, which is then able to accurately transfer control points from the key-frames to the intermediate frame. In particular, this method is able to process videos where intrinsic parameters vary from one frame to the next, which may be the case when optical stabilization of the acquisition is used.
- Design and conception of a method of visual servoing, which requires a 3D model of the scene, for the online estimation of the acquisition parameters associated with a test video. This method refines the estimation of acquisition parameters, based on the registering transformation measured between the real image and a rendering of the scene from the current estimate of the acquisition parameters. This method may be adapted to match various constraints related to accuracy, execution times, or varying intrinsic parameters.
- Design and development of a method of temporal consolidation of the change detection results, able to improve accuracy. This method uses a specific modeling of the problem

1. Note that we did not study 3D reconstruction techniques in this PhD. The heights used by our 3D model were obtained from the SRTM3 digital terrain model, distributed freely by NASA.

under the framework of belief propagation, and performs an optimization of the spatio-temporal consistency between successive results on the test video frames.
- Introduction of a relevance feedback method, using a specific region descriptor, able to adjust the detection results to the needs of the user. In particular, this method enables to eliminate the large majority of residual false alarms, while still detecting most of the targeted changes.
- Design and implementation of a descriptor of regions in an image, which is used by our relevance feedback method. This descriptor combines several criteria, based for instance on the shape of the considered region, on the intensity and color of the observations it contains, etc. This descriptor is able to differentiate with a good accuracy between regions corresponding to changes of possible interest or those corresponding to irrelevant variations.

Moreover, our work and contributions led to the following products and scientific publications:

- Article [14] published and presented orally at the 2011 conference *International Geoscience And Remote Sensing Symposium* (IGARSS),
- Article [15] published and presented orally at the 2012 conference *International Geoscience And Remote Sensing Symposium* (IGARSS),
- Article [16] published at the 2012 conference *International Geoscience And Remote Sensing Symposium* (IGARSS),
- Patent [13] in the process of being registered,
- Article [18] describing the content of the patent, waiting to be submitted to journal *IEEE Transactions on Geoscience and Remote Sensing*,
- Publication [12] of a synthetical dataset for benchmarking change detection techniques between pairs of aerial images,
- Technical report [17] published in the internal journal of Télécom ParisTech,
- Development of a demonstrator based on Qt, OpenGL and OpenCV, allowing the demonstration of the developed functionalities.

B.1.4 Evaluation data

The objective and quantitative evaluation of algorithms requires that some dataset and the corresponding ground-truth be available. In the context of change detection in aerial videos, these two things are difficult to obtain because they require expensive resources and are very time-consuming. On the first hand, the ideal dataset should indeed be acquired under various conditions of acquisition in order to analyze the influence of various factors (e.g. illumination, haze, viewpoints, ground resolutions etc) and should also contain relevant and statistically representative changes to be detected, which depends on the targeted application. On the other hand, obtaining the ground-truth in the context of change detection requires manual annotations from an image analyst, which represent a tremendous and tedious work especially when using videos. Moreover, the resulting ground-truth is not objective and may vary with the annotator and with the time of annotation [95].

In the literature, the authors have used various approaches in order to avoid this considerable problem: some of them only present qualitative evaluation results [92, 103], some present quantitative results on synthetical data [90, 91] and others evaluate their algorithms in the context of background subtraction [33]. We think this last approach is questionable, because even though the two fields are close, the working hypothesis of background subtraction are quite different from those of change detection (e.g. single video stream, focus on most recent reference video frames, etc). The following presents the approach chosen during this PhD in order to evaluate our algorithms, quantitatively and in the context of change detection, using both synthetical and real videos.

The quantitative evaluation process for a given algorithm usually has two main objectives: to determine the accuracy of the results and to quantify the robustness to various perturbations. In the first case, the ground-truth corresponding to the data must be available, it may therefore be useful to use synthetical data since the corresponding ground-truth is usually easy to obtain. On the other hand, in the second case, it is better to work with real data in order to test the algorithms using realistic perturbations.

Synthetical aerial videos In some experiments, we used synthetical videos in order to evaluate our algorithms. These synthetical videos were generated using Virtual Battle Station 2 (VBS2), which is a realistic rendering software developed by Bohemia Interactive Simulations. This software contains a large library of object models and enable the dynamic and realistic simulation of vast geographical scenes. Moreover, this software enable realistic simulation of meteorological or illumination conditions. Finally, VBS2 enables the automatic and fast extraction of ground-truth, related for instance to acquisition trajectories, digital elevation models and true change masks. Figure 6.1 shows an example of synthetical video generated using VBS2.

Real aerial videos The real aerial videos used in this thesis to evaluate our algorithm were obtained through an acquisition campaign led by Cassidian, in collaboration with Astrium Satellites and EADS Innovation Works. These videos were acquired using an aircraft equipped with an EO/IR observation turret. This turret recorded the visible video stream using an HD stabilized camera (1280×720 pixels). The specific dataset used to evaluate our algorithms is composed of five aerial videos acquired over the Darois aerodrome, near Dijon in France. Figure 6.2 presents the acquisition trajectories of these videos, overlapped on a map of the region.

This acquisition campaign only involved aerial means and did not involve ground support in order to produce changes to be detected. As a consequence, in order to evaluate our change detection approach, we developed an augmented reality method in order to insert virtual changes in the raw, change-free, videos. This method allows the generation of an infinite number of different videos but also the fast and automatic extraction of ground-truth. Basically, this method uses the geo-localization of the video frames (see section B.2.1) to warp new rectangular textures in the raw videos. Figure 6.3 shows an exemple of aerial video frame before and after insertion of virtual changes (highlighted by black arrows), which demonstrates that these changes cannot be trivially identified as virtual changes.

B.2 Pre-processing

As mentioned in [95], most change detection methods rely on data pre-processing in order to filter common types of irrelevant changes, before making the change detection decision. In our method, we use two steps of data pre-processing in order to attenuate geometric effects (ray-casting errors and 3D effects, such as parallax or occlusions) and radiometric variations (such as shadows and highlights).

B.2.1 Geo-localization

Knowledge of the camera poses for each frame, both for the reference and test videos, is necessary in a change detection approach based on 3D appearance modeling, because it is crucial to map image intensities correctly into the 3D model. However, today, acquisition devices recording camera poses in an accurate and frame-synchronized manner are rare, heavy and expensive. Moreover, modern aerial video acquisition devices often integrate a stabilization system, which might result in a frame-wise modification of the calibration parameters of the camera.

In this section, we therefore describe two algorithms allowing the accurate estimation of extrinsic and intrinsic acquisition parameters for each frame of a video. The first one implements a semi-automatic offline estimation scheme, which first requires the manual calibration of a few video frames and then automatically interpolates the poses of the remaining video frames using the trifocal tensor. The second one is based on Visual Servoing and aims at the online estimation of camera poses directly from the images, by using a 3D model of the scene.

B.2.1.1 Pose interpolation

A standard technique, called camera resectionning [52, algorithm 7.1], can be used for the estimation of the acquisition parameters associated with a given image. This algorithm is based on the minimization of the geometric error associated with a set of correspondences between 2D points in the image and 3D points in the scene. For a given reference image, this set of 2D $\leftrightarrow$ 3D correspondences may be obtained by choosing landmarks in the scene (e.g. building corners, particular rocks, trees, etc), retrieving their 3D coordinates (e.g. GPS coordinates) and finally, by localizing their projections in the image. For a given set of landmarks, this 2D localization is usually done manually, which is tedious but possible for a few images. However, in the context of videos, this manual approach is not realistic. Hence, the main idea of our pose interpolation approach is to manually calibrate a few keyframes from the video and then to propagate pose estimation to the other frames of the video using the trifocal tensor.

As a consequence, we developed [18] an algorithm able to propagate automatically the annotation provided manually on a few keyframes to the remaining video frames. This propagation mechanism uses an interpolation based on the automatic estimation of the trifocal tensor [52, chapter 15], which represents the constraints relating three views of a single scene. This trifocal tensor has the interesting property to enable the accurate localization of the projection in an image of a point in the scene, given the projections of this point in the two other images (see figure 3.1a). As shown in figure 3.1b, this mechanism may then be used to estimate the acquisition parameters of all frames of a video. The pseudo-code of this method is provided by algorithm B.1.

Notice that an accurate estimation of the trifocal tensor is critical to ensure that the estimated acquisition parameters are correct and usable. This accuracy depends on the quality and the number of SIFT point matches between the three images, which in turn depends on the similarity between these three images. In other words, in order to maximize the estimation accuracy, the three images have to be as close as possible in the video. This means that, asymptotically, the maximum accuracy is obtained when all video frames are calibrated manually. Since this is impossible to achieve in practice, this leads to a trade-off between annotation effort and estimation accuracy.

The algorithmic complexity of this method is linear with the number of images to process. However in practice, its processing time is dominated by the trifocal tensor estimation algorithm, which is quite slow.

This method assumes that the set of all video frames is available and requires information given by the user. This makes this approach appropriate for the geo-localization of reference videos, but inappropriate for test videos which should be processed incrementally.

B.2.1.2 Visual servoing

For the incremental geo-localization of the test video frames, we proposed to use the visual servoing approach [33]. The idea of this technique is to guide the estimation of the acquisition parameters using the registration between the considered image and a rendering of a 3D model of the scene, using the current estimation of these acquisition parameters. This technique consists of two steps: the coarse estimation of an initial value for the acquisition parameters, and the subsequent refinement of this coarse estimation. For the first step, we developed a prediction

Inputs: Set of images $\{I_t\}_{t\in\mathbb{N}}$, keyframe indices $K \subset \mathbb{N}$ and set of landmarks $\{\mathbf{M}_i\}_{i\in\mathbb{N}}$
Outputs: Projection matrices $\widehat{\mathrm{P}_n}$ for each frame I_n

1: **For** $k \in K$ **Do** ▷ Annotation step
2: Localize landmark projections $\{\mathbf{m}_{i,k}\}_{i\in\mathbb{N}}$ in keyframe I_k
3: Estimate the projection matrix $\widehat{\mathrm{P}}_k$ associated to keyframe I_k ▷ [52], Alg 7.1
4: **End For**
5: Joint optimization of projection matrices $\{\widehat{\mathrm{P}}_k\}_{k\in K}$ using bundle adjustment ▷ [52], § 18.1
. . . . . .
6: **For** $t \in \mathbb{N}\backslash K$ **Do** ▷ Interpolation step
7: $k_1 \leftarrow \sup\{k \in K \mid k < t\}$
8: $k_2 \leftarrow \inf\{k \in K \mid k > t\}$
9: Find three-view correspondences between I_n, I_{k_1} and I_{k_2}
10: Estimate the trifocal tensor between $\widehat{\mathcal{T}}_{t,k_1,k_2}$ using the three-view correspondences ▷ [52], Alg 16.4
11: Find landmark projections $\{\mathbf{m}_{i,t}\}_{i\in\mathbb{N}}$, transfered from I_{k_1} and I_{k_2} to I_n ▷ [52], § 15.3.2
12: Estimate the projection matrix $\widehat{\mathrm{P}_n}$ ▷ [52], Alg 7.1
13: **End For**

ALGORITHM B.1: *Algorithm for the offline estimation of the acquisition parameters associated to each frame of a given video.*

algorithm estimating the acquisition parameters using the parameters estimated for the previous images and the registration between the current and previous images. For the second step, we used a correction algorithm following the idea of the visual servoing technique and using a specific Jacobian matrix (see [18] for the derivation), which enables the accurate estimation of acquisition parameters. The implementation of both these steps is based on the analytical expression of the homography $\mathrm{H}^f_{2\leftarrow1}(\mathbf{x}_1, \mathbf{x}, \pi)$ registering image I_2 with respect to image I_1, as a function of the acquisition parameters $\mathbf{x}_1$ and $\mathbf{x}_2$ of these two images and of the parameters π of the dominant plane in the scene. The derivation of this analytical expression is derived in appendix section A.1.1 (you can also refer to [18] for more details).

Narrow and General cases We considered two cases for our visual servoing algorithm: the narrow case, where we estimate only the extrinsic pose parameters (position and orientation), and the general case, where we also estimate the intrinsic pose parameters (focal lengths and principal point coordinates). These two cases each lead to a different expression of the Jacobian matrix, which is used both in the prediction and in the correction steps. The expression of the Jacobian matrix used for the narrow case is provided in table A.1, whereas the expression for the general case is given in table A.2.

Notice that both the prediction and the correction steps are based on the resolution of a system of linear equations. This system is over-constrained in the narrow case (estimation of extrinsic parameters) and under-constrained in the general case (estimation of extrinsic and intrinsic parameters). In both cases, the system is solved using the singular value decomposition. Moreover, in the general case, we also use a damping on the update equation of the intrinsic parameters, in order to control the evolution of these parameters.

Prediction step We denote the acquisition parameters of a given camera $\mathcal{C}_1$ by $\mathbf{x}_1$ and those of camera $\mathcal{C}_2$ by $\mathbf{x}_2$. We assume that $\mathbf{x}_1$ is known and we want to estimate $\mathbf{x}_2$. As the associated images I_1 and I_2 are available, we can estimate, using RANSAC algorithm [45] on SURF point matches [5], the homography $\widehat{\mathrm{H}}_{2\leftarrow1}$ registering I_2 with respect to I_1. We are then interested in finding the parameters $\mathbf{x}_2$ best explaining the observed homography $\widehat{\mathrm{H}}_{2\leftarrow1}$. In other words, we want to estimate the acquisition parameters $\widehat{\mathbf{x}}_2$ for which the function

$\mathbf{x} \mapsto \text{vec}\left(\text{H}^f_{2\leftarrow 1}(\mathbf{x}_1, \mathbf{x}, \pi)\right) - \text{vec}\left(\widehat{\text{H}}_{2\leftarrow 1}\right)$ is equal to zero, which can be done using Newton's method for multivariate non-linear functions, as shown in algorithm B.2.

This prediction step is very fast and, in practice, its execution time is dominated by the time needed to estimate the registering homography $\widehat{\text{H}}_{2\leftarrow 1}$. However, its accuracy is directly dependent on the accuracy with which parameters $\mathbf{x}_1$ are known. In the case of a sequential use of this algorithm, for instance for the incremental geo-localization of a video, this problem may lead to the accumulation of estimation error and to a divergence of the estimation.

Correction step In order to avoid such a divergence, we use the visual servoing approach to correct the estimation using a 3D model of the scene. We consider a coarse estimation $\widetilde{\mathbf{x}}_2$ of the acquisition parameters for camera $\mathcal{C}_2$, which in practice is obtained using the prediction step. We denote by $I_r(\mathbf{x})$ the image generated using a rendering of the 3D model from the variable acquisition parameters $\mathbf{x}$ and by $\widehat{\text{H}}_{r\leftarrow 2}(\mathbf{x})$ the homography registering $I_r(\mathbf{x})$ with respect to I_2. We are then interested in finding the acquisition parameters leading to a perfect alignment between I_2 and $I_r(\mathbf{x})$. In other words, we want to estimate the parameters $\widehat{\mathbf{x}}_2$ for which the function $\mathbf{x} \mapsto \text{vec}\left(\text{H}^f_{r\leftarrow 2}(\mathbf{x}_2, \mathbf{x}, \pi)\right) - \text{vec}\left(\text{I}_\text{D}\right)$ is equal to zero. However, as this function involves the true acquisition parameters $\mathbf{x}_2$, which we want to estimate, we consequently use the empirically measured homography $\widehat{\text{H}}_{r\leftarrow 2}(\mathbf{x})$ instead of the analytical homography $\text{H}^f_{r\leftarrow 2}(\mathbf{x}_2, \mathbf{x}, \pi)$. Again, this problem may then be solved using Newton's method for multivariate non-linear functions, as shown in algorithm B.2.

Note that this approach is slow, since it requires a rendering of the 3D model and the estimation of the registering homography at each iteration. A faster (but less accurate) version of this correction step consists of using the prediction step a second time, this time in order to predict the acquisition parameters of the current image based on the rendering image (see B.2).

Like the pose interpolation algorithm, this visual servoing algorithm also has linear complexity with respect to the number of images to process. In practice however, it is slightly faster, which is interesting for the online processing of the test video.

B.2.1.3 Evaluation

The accuracy of these geo-localization methods can be evaluated using two criteria: the estimation error, measuring the difference of the estimated value and the target value of the acquisition parameters, and the reprojection error, measuring the accuracy of the alignment between 3D points in the scene and the corresponding 2D points in the image. The latter is particularly important in the context of change detection, especially when using appearance models, since an inaccurate alignment between observations and associated appearance models will lead to poor change detection performances.

Estimation error Figure 6.4 compares, on a synthetical aerial video, the trajectories estimated using our various geo-localization algorithms (pose interpolation and visual servoing technique) with respect to the ground-truth (circular trajectory) and to the trajectory estimated using the technique presented by Crispell et al. [33]. Table 6.1 reports the average execution times per image for each algorithm.

Both our narrow-case visual servoing technique (either using the accurate or the fast version, as explained in section B.2.1.2) and the visual servoing technique by Crispell et al. [33] lead to estimated trajectories which are very close to the ground-truth. However, it is interesting to see that our fast-version narrow-case visual servoing technique is five times faster than the other approaches, with little impact on the estimation accuracy.

The trajectory obtained by our general-case visual servoing algorithm is less accurate and contains obvious disparities with respect to the ground-truth trajectory. This may be explained

Inputs: Current and previous images I_n,I_{n-1}, estimation $\hat{\mathbf{x}}_{n-1}$ of previous acquisition parameters, 3D model of the scene, choice between fast or accurate correction step
Outputs: Estimation $\hat{\mathbf{x}}_n$ of the current acquisition parameters

Find normal $\mathbf{n}_\pi$ and distance to origin d_π, parameters of the dominant plane in the 3D model, using least square fitting
. . . . . .
Compute the homography $\widehat{\mathrm{H}}_{n\leftarrow n-1}$ registering I_n w.r.t. I_{n-1} ▷ Prediction step
$\mathbf{x} \leftarrow \hat{\mathbf{x}}_{n-1}$
Do
  Compute the analytical registration $\mathrm{H}^f_{n\leftarrow n-1}(\hat{\mathbf{x}}_{n-1}, \mathbf{x}, \mathbf{n}_\pi, d_\pi)$
  Solve (SVD) for $d\mathbf{x}$: $J_H(\hat{\mathbf{x}}_{n-1}, \mathbf{n}_\pi, d_\pi) \cdot d\mathbf{x} = \mathrm{vec}\left(\widehat{\mathrm{H}}_{n\leftarrow n-1}\right) - \mathrm{vec}\left(\mathrm{H}^f_{n\leftarrow n-1}(\hat{\mathbf{x}}_{n-1}, \mathbf{x}, \mathbf{n}_\pi, d_\pi)\right)$
  Update $\mathbf{x}$ using $d\mathbf{x}$ (accumulate linear parameters, compose rotation matrices)
Until convergence ($d\mathbf{x}$ close to zero)
$\tilde{\mathbf{x}}_n \leftarrow \mathbf{x}$
. . . . . .
$\mathbf{x} \leftarrow \tilde{\mathbf{x}}_n$ ▷ Correction step
If Accurate version **Then**
  Do
    Render image $I_r(\mathbf{x})$ of the 3D model from parameters $\mathbf{x}$
    Compute the homography $\widehat{\mathrm{H}}_{n\leftarrow r}(\mathbf{x})$ registering I_n w.r.t. $I_r(\mathbf{x})$
    Solve (SVD) for $d\mathbf{x}$: $J_H(\mathbf{x}, \mathbf{n}_\pi, d_\pi) \cdot d\mathbf{x} = \mathrm{vec}\left(\mathrm{ID}\right) - \mathrm{vec}\left(\widehat{\mathrm{H}}_{r\leftarrow n}(\mathbf{x})\right)$
    Update $\mathbf{x}$ using $d\mathbf{x}$ (accumulate linear parameters, compose rotation matrices)
  Until convergence ($d\mathbf{x}$ close to zero)
Else
  Render image $I_r(\tilde{\mathbf{x}}_n)$ of the 3D model from parameters $\tilde{\mathbf{x}}_n$
  Compute the homography $\widehat{\mathrm{H}}_{n\leftarrow r}(\tilde{\mathbf{x}}_n)$ registering I_n w.r.t. $I_r(\tilde{\mathbf{x}}_n)$
  Do
    Compute the analytical registration $\mathrm{H}^f_{n\leftarrow r}(\tilde{\mathbf{x}}_n, \mathbf{x}, \mathbf{n}_\pi, d_\pi)$
    Solve (SVD) for $d\mathbf{x}$: $J_H(\tilde{\mathbf{x}}_n, \mathbf{n}_\pi, d_\pi) \cdot d\mathbf{x} = \mathrm{vec}\left(\widehat{\mathrm{H}}_{n\leftarrow r}(\tilde{\mathbf{x}}_n)\right) - \mathrm{vec}\left(\mathrm{H}^f_{n\leftarrow r}(\tilde{\mathbf{x}}_n, \mathbf{x}, \mathbf{n}_\pi, d_\pi)\right)$
    Update $\mathbf{x}$ using $d\mathbf{x}$ (accumulate linear parameters, compose rotation matrices)
  Until convergence ($d\mathbf{x}$ close to zero)
End If
$\hat{\mathbf{x}}_n \leftarrow \mathbf{x}$

ALGORITHM B.2: *Algorithm for the online estimation of the acquisition parameters associated to each frame of a given video.*

by the fact that this technique also estimates intrinsic parameters, which introduces an ambiguity in the acquisition parameters since, for instance, moving the camera farther from the scene can be approximately counterbalanced by a zoom. This ambiguity may lead to a divergence of the estimation, which in practice may be prevented using a damping in the update equation for the intrinsic parameters.

The trajectory obtained using our pose interpolation algorithm follows the general trend of the ground-truth trajectory, however it is much more noisy. This may be explained by the ambiguity introduced by the estimation of intrinsic parameters. Furthermore, this method focuses on the minimization of the reprojection error, which makes it more appropriate in the context of change detection, as will be explained in the following.

Reprojection error The quantitative evaluation, in terms of reprojection error, of our two geo-localization techniques and of the method proposed by Crispell et al. [33] leads to the histogram presented in figure 6.6a. This histogram shows that the approach associated with the smallest reprojection error is our pose interpolation technique, with an average reprojection error of 0.2 pixel. Moreover, our visual servoing technique is associated to an average reprojection error of 0.9 pixel, whereas in the case of the alternative visual servoing technique, the average

reprojection error is 4.6 pixels. This gap between these two techniques may be explained by (i) the use, in the case of the alternative technique, of an approximate expression for the Jacobian matrix, not taking into account changes caused by certain types of camera motion [33], and by (ii) the use, in the case of our technique, of a prediction step allowing to obtain an initial alignment, between the image and the 3D model, very close to the optimal solution.

This prediction step enables our visual servoing technique to be much more robust than the alternative technique to high differences between the initial alignment and the target alignment. This point is clearly demonstrated by the graph shown in figure 6.6b, which compares the point clouds representing the final reprojection error as a function of the initial reprojection error. This graph shows that, even though our visual servoing technique has to deal with much larger initial reprojection errors (up to 1000 pixels), it results in lower final reprojection errors (less than 2 pixels for most frames). In other words, even when subjected to initial alignments which are considerably worse than in the case of the alternative visual servoing technique, our technique is able to recover a very accurate alignment between the considered image and the 3D model.

This point is crucial in the context of change detection based on appearance modeling, because if a bad alignment between the image and the 3D model is used, it will be impossible to correctly compare test observations with the corresponding appearance models. Therefore bad performances in terms of reprojection error will necessarily lead to bad performances in change detection.

Real data In real scenarios, it might be necessary to use a geo-localization technique able to estimate intrinsic parameters in a frame-wise manner, for instance in the case of a stabilized aerial video. Figure 6.7 compares the trajectories estimated on such a video, for which the ground-truth is unknown, using our pose interpolation technique, our general-case visual servoing technique, and the visual servoing technique proposed by Crispell et al. [33]. These results show a good consistency between the trajectories estimated using our techniques, which both enable frame-wise estimation of the intrinsic parameters. As already mentioned, the trajectory estimated using our visual servoing technique has a better continuity than the one estimated using our pose interpolation technique. On the other hand, the trajectory estimated using the alternative visual servoing technique is clearly diverging, showing that in certain cases which nowadays are more and more frequent, frame-wise estimation of the intrinsic parameters is essential.

Figure 6.8 compares the change detection performances obtained on the same stabilized aerial video, using the complete framework developed during this PhD and one of the three geo-localization techniques mentioned above. Two cases were considered: (i) the case where the raw estimated acquisition parameters are used directly, and (ii) the case where a final affine registration was performed between the image and the 3D model [2], in order to further decrease the reprojection error before the actual change detection.

In the first case, where the raw geo-localization is used, the performances are significantly different for each technique. First, the visual servoing technique proposed by Crispell et al. [33] leads to poor change detection performances. Moreover, our visual servoing technique leads to performances clearly inferior to those obtained using our pose interpolation technique. These differences may be explained by the different performances obtained using these techniques in terms of reprojection errors. In the second case, where an affine registration is performed prior to the change detection, the curves all show an improvement over the case where the raw geo-localization is used. In addition, the performance gap between our pose interpolation technique and our visual servoing technique is considerably reduced. This can be explained by the fact that, even though the alignment estimated using our visual servoing technique is

2. In practice, the registration between the image under geo-localization and the 3D model is performed by registering the image with a rendering of the 3D model from the acquisition parameters estimated by the geo-localization technique.

only approximate, the resulting trajectory is accurate enough to obtain a good alignment after a simple affine registration between the image and the 3D model. On the contrary, this is not the case for the alternative visual servoing technique, which, even after the affine registration, still leads to poor change detection performances.

To conclude this section, it is to be noted that the quality of the 3D appearance model, which is used extensively in the visual servoing approach, plays an important role in the final accuracy of the estimated trajectory and acquisition parameters. More particularly, the registration of the rendered image and the real image may be inaccurate if the resolution of the appearance model is not sufficient or if the appearances used for the rendering (here, average observation) are too different from the observations in the real image (e.g. due to illumination effects, extreme appearance variations, etc).

B.2.2 Illumination attenuation

The first mechanism used in order to handle illumination consists of using a representation of observations which is invariant to illumination variations. The choice of this type of techniques was mainly motivated by our goal to perform change detection, which naturally led us towards an approach allowing a stable comparison between observations. Moreover, we also looked for an approach allowing to exploit the multiple observations available in reference videos, in order to determine normalized illumination conditions.

Consequently, we chose to use the notion of chromaticity, which was employed by many approaches [38, 44, 47] to achieve invariance to illumination variations. This notion of chromaticity is also interesting because it makes it possible to process each pixel or cell in an independent manner. This was a requirement due to the structure of our 3D model, where defining a neighborhood relationship between cells was cumbersome.

B.2.2.1 Representations using the classic chromatic coordinates

The representations using the classic chromatic coordinates are the ones leading to the best performances, among the techniques which were evaluated, and are therefore described below.

Raw representation The raw representation using classic chromatic coordinates is based on ratios between the colors of the observations, in order to eliminate the luminosity of these observations. For a given observation, denoted (R,G,B), the classic chromaticities can be obtained as follows:

$$(r,g,b) = \left(\frac{R}{R+G+B}, \frac{G}{R+G+B}, \frac{B}{R+G+B} \right) \tag{B.19}$$

Under the assumption that the light source is white and that the objects in the scene follow the Lambertian diffusion law, Gevers et Smeulders [47] showed that this representation is invariant to variations in the illumination direction or intensity. This representation remain interesting beyond this rather strict assumption, but then false alarms may occur in some cases.

The second column in figure 3.3 shows the raw representations of the two images shown in the first column, using classic chromatic coordinates. These results show that illumination variations are strongly attenuated, however it is clear that this representation leads to a loss of information, since, for instance, the white buildings (in the center) cannot be distinguished from the gray asphalt anymore. This loss of information may have important consequences on the detection of changes, particularly related to non-detections, as the discriminative power of observations is diminished.

Normalized representation In order to minimize the influence of this loss of information, we proposed [18], for any physical point under consideration, to transform the result of this invariant representation back to the traditional color space. For that purpose, we used the average observation luminosity in reference videos, at the considered physical point, denoted $\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}$, in order to scale the chromatic coordinates. This approach is justified by two points. First, using the average of many observations for a given physical point is an effective approach to attenuate the non-stationary effects of illumination [78]. Second, the approach consisting of using the reference illumination conditions in order to represent test observations can be seen as a way to compare the reference and test data under normalized illumination conditions.

The normalized representation using classic chromatic coordinates is expressed as follows:

$$\begin{aligned} R_{norm} &= r \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) = \frac{R}{R+G+B} \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) \\ G_{norm} &= g \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) = \frac{G}{R+G+B} \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) \\ B_{norm} &= b \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) = \frac{B}{R+G+B} \cdot (\overline{R}_{ref} + \overline{G}_{ref} + \overline{B}_{ref}) \end{aligned} \tag{B.20}$$

The image shown on the second line of the first column in figure 3.4 illustrates the result obtained using this normalized representation. When compared with the raw representation, it is clear that the discriminative power of the observations has been restored, allowing to avoid the generation of non-detection without introducing any variability related to illumination. However, the normalized representation is unstable for dark observations and generate over-saturated colors in dark areas (as highlighted by black arrows), which lead to an increase of false-alarms.

Compensated representation In order to avoid this problem, we proposed [18] to progressively compensate dark observations using the average observation in reference videos. More precisely, with the same notations as above, the compensated representation is as follows:

$$\begin{aligned} R_{comp} &= (1-\beta) \cdot R_{norm} + \beta \cdot \overline{R}_{ref} \\ G_{comp} &= (1-\beta) \cdot G_{norm} + \beta \cdot \overline{G}_{ref} \\ B_{comp} &= (1-\beta) \cdot B_{norm} + \beta \cdot \overline{B}_{ref} \\ \text{with} \qquad \beta &= \left(1 - \frac{R+G+B}{3}\right)^5 \end{aligned} \tag{B.21}$$

The image shown on the third line of the first column in figure 3.4 illustrates the result obtained using this compensated representation. This resulting image shows that dark areas are correctly handled using this representation. Still, it is to be noted that this compensated representation has a weakness: it replaces information from the test observations by information from the reference observations. Therefore, using this representation may generate non-detections, for instance if a dark change occurred in the test video. In such a case, the change will be replaced by reference information, which obviously contain no change, hence the initial change would not be detectable anymore.

B.2.2.2 Evaluation

In order to evaluate the performances obtained using these illumination handling techniques, we compared the ROC curves obtained with each representation and using different types of chromatic coordinates: classic chromatic coordinates [38], logarithmic chromatic coordinates [44] and $L1L2L3$ chromatic coordinates [47]. These curves are compared in figure 6.11, using a common color when they correspond to the same type of chromatic coordinates and a common point shape when they correspond to the same type of representation.

In a general manner, the objective of illumination attenuation algorithms is to obtain a representation of the observations which is invariant to illumination variations, while still allowing to detect significant changes. However, in practice, these two constraints are often contradictory and lead to a trade-off between the minimization of false positives and the minimization of false negatives. This trade-off is well illustrated by the ROC curves presented in figure 6.11, which show that, for values of the false positive rate greater than 15%, the best performance is obtained when no illumination-invariant representation is used. In other words, when false alarms are tolerated, the best way to detect most of the changes which occured in the test video is to work directly with the test observations, without any illumination-related preprocessing.

However, minimizing the number of false alarms is generally wanted, especially for a semi-automatic video-analysis system, which is supposed to alleviate the user's work load. In such cases, that is for values of the false positive rate lower than 15%, the graph shows that the technique leading to the best performance is the one based on the classic chromatic coordinates using the compensated representation.

Table 6.2 reports the average execution times per image (each image being of size 1280×720 pixels), expressed in milliseconds, as a function of the representation and the type of chromatic coordinates. Moreover, some examples of the change masks estimated with the various representations and chromatic coordinates are presented in figure 6.12, where the image shown in (d) is obtained without attenuation and the one in (f) is the ground-truth change mask.

B.3 Change detection

As briefly mentioned in the introduction, the problem of detecting significant changes between videos acquired from mobile cameras raises many sub-problems. The approach consisting of doing a three-dimensional appearance modeling addresses a good part of these sub-problems, including in particular the appropriate handling of geometrical effects (e.g. occlusions and parallax effects) and of varying appearances in the scene. The method developed during this PhD has the specificity of separating the geometrical aspect from the appearance modeling aspect, which enables using and comparing various appearance modeling algorithms.

B.3.1 Tri-dimensional database

Our method makes use of a three-dimensional database in order to perform a spatial organization, consistently with the geometry of the scene, of the reference video frames and the appearance models.

For the purpose of organizing the appearance models, some techniques in the literature [33, 91] chose to use a volumetric model of the scene. However, in a scenario where the acquisition platform stays at significant distance from the observed scene, which is frequent in aerial observation, it seems more appropriate to model the scene as a surface than as a volume. Indeed, the accuracy improvement of a volumetric modeling is often anecdotal, but results in a much larger memory occupancy which can be prohibitive for large scenes. We therefore proposed [18] to represent the scene by a surface with variable height, using a hierarchic structure, which will be referred to as an augmented Quad-Tree, combining a height map with a Quad-Tree [43].

In order to allow the user to analyze and browse the reference video frames manually, which are much more informative to him than the appearance models, we also chose to integrate in our database a spatial indexing of the raw video frames, based on a R-Tree structure [50].

B.3.1.1 Indexing video data

This section presents the algorithms related to the step of data indexing from a video. These algorithms assume that the elevations in the scene are available, for instance from a Digital

Terrain Model [3] or a 3D reconstruction, and that the acquisition parameters for every considered image are known (see section B.2.1.1 for an offline estimation method).

Video frame indexing In order to index the video frames, we use two steps: (i) computing the ground viewprint of every video frame, and (ii) generating the R-Tree for the spatial indexing of these frames.

Ground viewprint computation The ground viewprint of each video frame may be computed using the knowledge of the acquisition parameters of this frame and of the elevations in the scene. Given a pixel on the border of the considered image, we use a ray-casting algorithm to determine the point in space representing the intersection between the scene surface and the ray back-projected from the considered pixel. This process leads to the determination of a sequence of 3D points, which represent the ground viewprint of the considered video frame. In practice, this sequence is filtered in order to remove the points aligned with their neighbors, which are numerous and might slow the computations. Finally, the bounding box of this ground viewprint is found using the upper and lower bounds on coordinates of the points in the sequence. Figure 4.5 show the ground viewprint estimated for all video frames of a real video.

R-Tree generation In order to spatially index the ground viewprint of all video frames, we chose to use the STR algorithm [66], for its simplicity and effective indexing performance. Since this algorithm is very fast, it can be run each time a new reference video is ingested in the database. In the case of the figure 4.5, the high overlapping rate makes the use of a R-Tree rather irrelevant, because the majority of the video frames show the same geographical region. However, the use of a R-Tree is crucial when the observed scene is vast and when each video frame only observe a small fraction of it. Notice that, once the R-Tree is generated, its root node contains the bounding box of the ground viewprints of all video frames, which is used to define the geographical extent of the reference scene.

Spatial organization of appearance models In order to organize the appearance models, we use three steps: (i) defining the root of the augmented Quad-Tree, (ii) adapting the Quad-Tree resolution to the ground resolution of the reference data, and (iii) computing the elevation of each Quad-Tree cell.

Definition of the root Given a new video to be ingested in the database, the first step for indexing its observations is to define the root node of the augmented Quad-Tree in order to cover the extent of the observed region. However, in order to avoid another processing of all data already ingested in the database, it is necessary to define a new root node which is compatible with the previously existing structure. This may be done by an upwards development of the Quad-Tree structure, as illustrated in figure 4.6. It may be noted that using a tree structure, whose resolution is adaptive contrarily to a simple grid structure, makes this process very efficient and causing no memory waste nor any copy or re-processing of existing data.

Adaptation of the resolution Once the root node of the Quad-Tree structure is defined to cover the observed region, its resolution can be adapted to match the ground resolution of the reference data. For that purpose, we developed a simple algorithm allowing to efficiently adapt the resolution of the Quad-Tree, or in other words its depth, without analyzing every pixel of every video frame [65]. Our algorithm performs a recursive check that the current tree structure

3. Nowadays, DTM data is freely available for any region in the world. For instance, NASA provides the elevations measured by the Shuttle Radar Topography Missions (SRTM) at the following URL: http://dds.cr.usgs.gov/srtm/version2_1.

Inputs: Projection matrix P_I and ground viewprint E_I of considered image I, Quad-Tree cell $\mathcal{C}$ under consideration by the current recursive call, minimum depth $\text{depth}_{\text{min}}(\mathcal{C})$ among the descendent of $\mathcal{C}$, 3D model of the observed region, minimum cell size l_{min}, maximum number N_{pixels} of frame pixels per projected cell
Outputs: Tree structure matching the ground resolution of the considered image

Compute the projected of cell $\mathcal{C}$ in image I using P_I the 3D model of the observed region
If $\text{Intersection}(\mathcal{C}, E_I) = \varnothing$ **Then**
    Return $\text{depth}_{\text{min}}(\mathcal{C})$
Else If $\text{Resolution}(\mathcal{C})/2^{\text{depth}_{\text{min}}(\mathcal{C})} < l_{min}$ **Then**
    Return $\text{depth}_{\text{min}}(\mathcal{C})$
Else If $\text{Area}(\text{projection of } \mathcal{C} \text{ in } I) < N_{\text{pixels}}$ **Then**
    Return $\text{depth}_{\text{min}}(\mathcal{C})$
End If
$\text{depth}_{\text{min}}(\mathcal{C}) \leftarrow \infty$
For Each child $\mathcal{D}$ of $\mathcal{C}$ **Do**
    If $\mathcal{D}$ is undefined **Then**
        Create $\mathcal{D}$
        $\text{depth}_{\text{min}}(\mathcal{D}) \leftarrow 0$
    End If
    $\text{depth}_{\text{min}}(\mathcal{C}) \leftarrow \text{MIN}\left[\text{depth}_{\text{min}}(\mathcal{C}), 1 + \text{Result of recursive call on } \mathcal{D}\right]$
End For
Return $\text{depth}_{\text{min}}(\mathcal{C})$

ALGORITHM B.3: *Recursive algorithm for the adaptation of the Quad-Tree resolution according to the available observations.*

is sufficient in order to obtain an accurate description of the considered frame. When this is not the case, a downwards development of the tree is performed until the required accuracy is reached. More particularly, we use three different stop cases in order to avoid browsing the whole structure for each video frame. The first one checks that the current Quad-Tree cell does intersect the ground viewprint of the considered frame, in order to avoid browsing cells which are useless for the considered frame. The second stop case compares the maximum resolution of the descendent nodes with the requested minimum resolution, l_{min}, which is a parameter of the algorithm. This test allows the algorithm to avoid browsing a tree branch when it cannot be developed anymore. The third stop case compares the area of the current cell projected in the considered frame with the resolution of the frame, in order to avoid browsing a branch when the description accuracy is sufficient for the considered frame. When these three stop cases allow a branch to be explored, our algorithm performs a downwards development of the tree structure and do the recursive calls on the current cell's child nodes. The pseudo-code of this method is given in algorithm B.3.

Figure 4.7 shows the exemple of a Quad-Tree subdivision obtained using this algorithm, using a minimal cell resolution of $25cm$ and an area of the projected cell matching the size of a group of 3×3 pixels. This visualization shows that the resulting structure, composed of 142 000 leaf nodes, is rather heavy and illustrates the trade-off to be found between memory occupancy and modeling accuracy. Some approaches of the literature [32] suggest to perform a post-processing compression of the model, by merging adjacent cells having similar appearance models and close heights. This idea is relevant when the objective is to generate a model of the observed scene, however it is not appropriate in the context of change detection. Indeed, depending on the chosen appearance model technique, this loss of resolution may cause a degradation of the change detection performance, especially for small changes.

Computation of the elevations Finally, we need to define the height associated with each of the Quad-Tree cells. These heights can be initialized using a priori information, such as DTM

data or a planar ground assumption, and may then be refined using a dense 3D reconstruction algorithm [34, 94]. However, it is to be noted that the heights obtained from DTM data cannot be used directly, as the size of the Quad-Tree cells may differ from the sampling rate of the DTM. Therefore, it is necessary to perform an interpolation or averaging of the DTM elevations in order to estimate the height of a given cell. Figure 4.8 shows the heights of each cell of the previous Quad-Tree structure.

B.3.1.2 Spatial querying

This section presents the algorithms which are used, when processing a given video frame, in order to (i) retrieve the video frames indexed in the database which show the same geographical area and (ii) find the appearance models corresponding to each of the considered observations. These algorithms assume that the acquisition parameters for the considered image are known (see section B.2.1.2 for an online estimation method).

Query for reference video frames The algorithm using a R-Tree structure to find the reference video frames showing the same area than a given test image is a classical one. The idea is explore a given tree branch only when the bounding box of the current node intersects the ground viewprint of the considered test image. When it does, the search is propagated to the child nodes of the current node. When a leaf node is reached, then it contains the ground viewprint of one of the reference video frame, which is added to the resulting list if it intersects the ground viewprint of the test image.

It can be noted that when the overlapping rate of reference video frames is high, the resulting list may contain almost all the reference video frames of the database. In such a case, the algorithm has to choose a single one to present the the image analyst, for instance the one whose intersecting area with the test image is largest (i.e. reference image with most similar content) or the one whose viewpoint is closest to the viewpoint of the test image (i.e. reference image with least perspective distortion). It is also possible to integrate other filtering criteria (such as a particular acquisition date and time, image modality, type of acquisition platform or image quality, etc) in order to allow the user to specify more accurately the reference image he wishes to retrieve.

Query for appearance models Finding the appearance models corresponding to a given set of observations is essential both for modeling appearances from the reference data and for detecting changes in the test data. The technique used in order to efficiently find the appearance model corresponding to a given observation is based on the ray-casting algorithm, but its actual implementation depends on the required processing.

When the required processing must result in an image, such as a rendering of the 3D model or a detection of changes in a test image, we use the classical ray-casting algorithm, which associates one appearance model to each pixel of the resulting image. However, when the required processing involves a modification of the appearance models, for instance when they are subjected to an incremental update step, we use an inverse ray-casting algorithm. Indeed, in general, the classical ray-casting algorithm does not associate one observation to each appearance model visible in the considered input image [4]. This pseudo-code of this inverse ray-vasting algorithm is given in algorithm B.4.

It is to be noted that both the classical and the inverse ray-casting algorithm make an extensive use of the collision algorithm [114] computing whether a given ray intersects a given cell (which also is a axis-aligned bounding-box, well known in computer graphics).

4. This problem is frequent in the context of image resampling after a geometric transformation, which can lead to holes in the final image if the resampling is not carried out correctly.

Inputs: Projection matrix P_I of input image I, Quad-Tree cell $\mathcal{C}$ under consideration by the current recursive call
Outputs: Associates one interpolated observation to each Quad-Tree cell visible in input image I

Using P_I, find the convex hull of cell $\mathcal{C}$ after projection in image I
If $E_{\mathcal{C}}$ is completely outside I **Then**
    Return
End If
For Each child $\mathcal{D}$ of $\mathcal{C}$ **Do**
    If $\mathcal{D}$ is defined **Then**
        Recursive call on $\mathcal{D}$
    End If
End For
If $\mathcal{C}$ contains an appearance model **Then**
    Scan the segment between $\mathcal{C}$ and the viewpoint of I to find a possible occluding cell
    If $\mathcal{C}$ is unoccluded **Then**
        Compute the interpolated observation **o** corresponding to the area $E_{\mathcal{C}}$ in image I
        Apply the required processing on $\mathcal{C}$ using interpolated observation **o**
    End If
End If

ALGORITHM B.4: *Inverse ray-casting algorithm, associating one interpolated observation to each cell of the augmented Quad-Tree visible in the input image.*

The left column in figure 4.9 shows two real images, one providing an overview of a scene and the other providing a more detailed view of a specific area. The red frame in the overview image shows the ground viewprint of the zoomed image. The right column of this figure shows corresponding synthetical images, obtained using a rendering, from the same viewpoint as the real images, of the augmented Quad-Tree generated using a reference video with the same ground resolution as the overview real image. The intensities used to generate the synthetical images are the average observations seen in the reference video at each physical point. These comparisons show that the correspondence is accurate in the case of the overview image, but that artifacts clearly appear in the zoomed synthetical view. This can be explained by the fact that the reference video used to generate the augmented Quad-Tree was not resolute enough to enable the accurate representation in the case of the zoomed image.

B.3.1.3 Evaluation

Elevation accuracy One of the main weakness related to three-dimensional approaches for change detection is that the performance may strongly depend on the quality of the 3D model of the scene, whose accurate estimation from aerial videos is still difficult in practice. Hence, to demonstrate the interest of such an approach, it is necessary to evaluate the influence of inaccuracies in the 3D model on change detection performances. For that purpose and in order to avoid entering the vast problem of 3D reconstruction, we used different a priori assumptions on the elevations in the observed scene. These assumptions are illustrated in figure 6.9 and correspond to (i) a planar ground (figure 6.9a), (ii) data from SRTM digital terrain model (figure 6.9b) and (iii) data from SRTM digital terrain model with manually annotated buildings (figure 6.9c). In each case, the augmented Quad-Tree was generated using a reference video and change detection performances were evaluated, using a test video where changes were located in strategic places (e.g. near buildings, in empty places where the elevation was different from the dominant plane, etc).

Figure 6.10 compares the ROC curves obtained in each case. These results show that the planar ground assumption clearly leads to inferior performances, but that the performances obtained in the two other cases are very close. This last point implies that small errors on the elevations estimated by a 3D reconstruction algorithm would have very small influence on the

change detection performances. This may be explained by our use of a height map structure for the representation of the scene. This strongly constrained form of 3D model is associated with a good robustness to moderate ray-casting errors, which leads to good robustness for the representation of the scene. More generally, this also shows that, in rural or lightly urban regions, the use of DTM data, which nowadays are freely available for any region of the world, is sufficient to obtain satisfactory change detection performance.

However, it is to be noted that using accurate elevation may be important for other algorithms. This is typically the case for the visual servoing geo-localization approach, which requires a good similarity between the 3D appearance model and the real scene in order to work properly.

Ground resolution The ground resolution of the considered data plays an important role in our approach, because, as mentioned in section B.3.1.1, the resolution of the 3D model directly depends on the maximum ground resolution in the reference videos. Hence, as mentioned in the previous section, artifacts may appear if the ground resolution of the reference videos is inferior to the ground resolution of the test video, which would lead to poor change detection performances. This point is illustrated by the ROC curves presented in figure 6.30, which shows that when the reference video has a ground resolution inferior to the test video, change detection performances are reduced.

Three solutions may be considered in future work in order to avoid this problem. The first one consists of maximizing the resolution of the 3D model independently from the ground resolution of the reference videos [34]. This would make the scene approximation invisible for a large range of ground resolution in the test video, but would also considerably increase the size of the final model. A second solution would be to artificially decrease the resolution of the test images to make it match the resolution in the reference videos. Finally, another solution would be to implement a multi-scale hierarchical approach, where a specific appearance model would be computed for each hierarchical level. The detection of changes would then be performed by choosing the most appropriate level, depending on the resolution of the test video.

B.3.2 Appearance modeling

The goal of appearance modeling is to perform an incremental compression of the reference observations, to obtain a model of the normal variations in the scene, and to enable the comparison of a test observation with the corresponding appearance model. Such an approach uses two main mechanisms: (i) the incremental update of a given appearance model based on an input observation and (ii) the comparison between a given appearance model and an input test observation, in order to detect possible changes.

Several modeling techniques have been evaluated and compared during this PhD. Most of them have been proposed for the domain of background subtraction, which is close, but different, to the domain of change detection. The following presents the technique providing the best performance, which is based on the incremental principal component analysis (PCA) approach proposed by Li [67], and provides details on the adaptations made.

B.3.2.1 Incremental PCA

The incremental PCA technique is very close to the *eigenfaces* technique proposed [109] in the domain of face recognition. The idea is to consider the observation of a given image as a huge column vector and to analyze, using a PCA, the variation space of the vectors corresponding to every reference video frame. In practice, the huge dimension of the considered vectors makes the computation of the covariance matrix intractable, which hence forbids the use of a classical PCA technique. Therefore, the incremental PCA approach computes an approximated PCA, updated incrementally for each reference video frame.

As the domain of change detection has specificities with respects to the domain of background subtraction, some modifications have been made to this incremental PCA technique. Firstly, in change detection, we are interested in comparing the test image with the set of all reference video frames, and not only with the most recent ones, which requires the use of a degressive update weight instead of a constant one. Secondly, the problem of missing data occurs very frequently due the geometrical effects (e.g. 3D occlusions) and partial views of the scene, which therefore necessitates the extensive use of the mechanism robust to missing data [67]. Finally, reference observations are modeled in an offline manner and may therefore be used in random order instead of sequentially, which avoids focusing the modeling effort on a non-representative fraction of the observations.

Model update The incremental update mechanism starts by gathering all observations resulting from the inverse ray-casting algorithm(see algorithm B.4) into the observation vector, denoted by $\mathbf{x}_t \in \mathbb{R}^{DN_{\text{cell}}}$ where D is the dimension of an observation (e.g. 3 for RGB observations) and N_{cell} is the number of cells in the Quad-Tree (e.g. $N_{\text{cell}} = 142\,000$ for the Quad-Tree in figure 4.7). We also use a missing coefficient vector, denoted by $\boldsymbol{\delta}_t \in \{0,1\}^{DN_{\text{cell}}}$, where a coefficient of 0 means that the corresponding observation is missing and a coefficient of 1 means that it is present. This missing coefficient vector also results from the inverse ray-casting algorithm. Moreover, in order to assign an identical weight to all observations in the final model, we use a degressive update weight vector, denoted by $\boldsymbol{\alpha}_t$, where $\forall i \in [\![0, DN_{\text{cell}-1}]\!], \alpha_t^i = \frac{\delta_t^i}{\sum_{n=0}^{t} \delta_n^i}$. The observation empirical mean and variance are then computed as follows:

$$\begin{array}{ll} \text{For } t = 0: & \text{For } t \geqslant 1: \\ \left\{ \begin{array}{l} \boldsymbol{\mu}_0 = \mathbf{x}_0 \\ \mathbf{v}_0 = \mathbf{0}_{DN \times 1} \end{array} \right. \quad \text{and} & \left\{ \begin{array}{l} \boldsymbol{\mu}_t = \boldsymbol{\mu}_{t-1} + \boldsymbol{\alpha}_t \circ (\mathbf{x}_t - \boldsymbol{\mu}_{t-1}) \\ \mathbf{v}_t = (1 - \boldsymbol{\alpha}_t) \circ [\mathbf{v}_{t-1} + \boldsymbol{\alpha}_t \circ (\mathbf{x}_t - \boldsymbol{\mu}_{t-1})^{②}] \end{array} \right. \end{array} \tag{B.22}$$

where $\circ$ and ^{②} respectively denote the element-wise multiplication and squaring. The empirical covariance matrix Σ_t of the observations up to step t can be approximated using the previous principal components and the current observation. Denoting the N_{CP} previous eigenvectors by $\{\mathbf{u}_{t-1}^k\}_{k \in [\![1, N_{\text{CP}}]\!]}$, and the corresponding previous eigenvalues, denoted by $\{\lambda_{t-1}^k\}_{k \in [\![1, N_{\text{CP}}]\!]}$, the new covariance matrix can be approximated as follows:

$$\begin{aligned} \Sigma_t &= (1 - \frac{1}{t+1}) \cdot \Sigma_{t-1} + \frac{\frac{1}{t+1}}{1 - \frac{1}{t+1}} \cdot \check{\mathbf{x}}_t \check{\mathbf{x}}_t^T \\ &\approx \frac{t}{t+1} \cdot \sum_{k=1}^{N_{\text{CP}}} (\lambda_{t-1}^k \cdot \mathbf{u}_{t-1}^k {\mathbf{u}_{t-1}^k}^T) + \frac{1}{t} \cdot \check{\mathbf{x}}_t \check{\mathbf{x}}_t^T \\ &= \mathrm{A}_t {\mathrm{A}_t}^T \end{aligned} \tag{B.23}$$

$$\begin{aligned} \text{where} \quad \check{\mathbf{x}}_t &= \boldsymbol{\delta}_t \circ (\mathbf{x}_t - \boldsymbol{\mu}_t) \\ \mathrm{A}_t &= \left[\begin{array}{cccc} \mathbf{y}_1 & \dots & \mathbf{y}_{N_{\text{CP}}} & \sqrt{\frac{1}{t}} \cdot \check{\mathbf{x}}_t \end{array} \right] \\ \mathbf{y}_k &= \sqrt{\frac{t}{t+1} \lambda_{t-1}^k} \cdot \mathbf{u}_{t-1}^k \end{aligned} \tag{B.24}$$

The incremental update requires the computation of the new principal components, which cannot be inferred from the diagonalization of $\Sigma_t = \mathrm{A}_t {\mathrm{A}_t}^T$ due to its huge dimensions. However, the new principal components can be inferred from the diagonalization of the Gram matrix $\mathrm{A}_t^T \mathrm{A}_t$, whose dimension is much smaller (here, $(N_{\text{CP}} + 1) \times (N_{\text{CP}} + 1)$). Indeed, the first N_{CP} eigenvalues of the Gram matrix are the same as those of the covariance matrix, and its eigenvectors, denoted by $\{\mathbf{u'}_t^k\}_{k \in [\![1, N_{\text{CP}}]\!]}$, are related to those we want to compute as follows:

$$\forall k \in [\![1, N_{\text{CP}}]\!], \mathbf{u}_t^k = \mathrm{A}_t \mathbf{u'}_t^k \tag{B.25}$$

This update mechanism can be used for iterations $t \geqslant N_{\text{CP}}$, and initialized at iteration $t_0 = N_{\text{CP}} - 1$ using the diagonalization of matrix A^TA, where :

$$\text{A} = \left[\begin{array}{ccc} \boldsymbol{\delta}_0 \circ (\mathbf{x}_0 - \boldsymbol{\mu}_{t_0}) & \dots & \boldsymbol{\delta}_{t_0} \circ (\mathbf{x}_{t_0} - \boldsymbol{\mu}_{t_0}) \end{array} \right] \tag{B.26}$$

It is to be noted that at each step, the eigenvector corresponding to the smallest eigenvalue is discarded, in order to keep a constant number of principal components equal to N_{CP}. This introduces an approximation with respect to the result of the classical PCA, which is impossible to obtain in practice (due to the huge dimension of the problem). It is however possible to measure this approximation by comparing the cumulative energy $\sum_{k=1}^{N_{\text{CP}}} \lambda_k$ of the final principal components (i.e. the sum of the final eigenvalues) to the sum of the elements in the final empirical variance vector $\sum_{i=0}^{DN_{cells}} v^i$. Figure 4.11 shows the empirical mean and the 7 final principal components obtained on a given video using the incremental PCA technique and provides the cumulative energies in parenthesis. Using $N_{\text{CP}} = 20$, we reach a total cumulative energy of 48.0% of the total variance in the observations.

Change detection Once the incremental PCA modeling has been computed on the reference videos, it may be used in order to detect deviations from this model. We denote by $\text{U} = \left[\begin{array}{ccc} \mathbf{u}^1 & \dots & \mathbf{u}^{N_{\text{CP}}} \end{array} \right]$ the $(DN_{\text{cell}}) \times N_{\text{CP}}$ matrix of final eigenvectors, and respectively by $\boldsymbol{\mu}$ and $\mathbf{v}$ the final empirical mean and variance vectors. Finally, we denote by $\mathbf{x}$ a test observation vector and by $\boldsymbol{\delta}$ the corresponding missing coefficient vector, and we define $\check{\mathbf{x}} = \boldsymbol{\delta} \circ (\mathbf{x} - \boldsymbol{\mu})$. The residual error vector is defined as the difference between $\check{\mathbf{x}}$ and its projection on the PCA sub-space:

$$\mathbf{r}(\mathbf{x}) = \check{\mathbf{x}} - \text{U}\text{U}^T\check{\mathbf{x}} \tag{B.27}$$

The elements of this residual error vector may then be compared to the empirical variance vector, defining a score vector $\mathbf{s}$ as follows:

$$\mathbf{s}(\mathbf{x}) = \frac{1}{\gamma^2} \cdot \left[\mathbf{r}(\mathbf{x}) \circ \mathbf{r}(\mathbf{x})\right] \oslash \mathbf{v} \tag{B.28}$$

where $\oslash$ represents the element-wise division and γ is a scaling constant related to the occurrence probability of a change. Finally, we assign, to each cell of index $i \in [\![0, N_{\text{cell}} - 1]\!]$ of the Quad-Tree, a detection score $\varepsilon_{\text{ipca}}$ computed using the D elements of $\mathbf{s}$ corresponding to cell i (e.g. for RGB observations, $D = 3$):

$$\varepsilon_{\text{ipca}}(\mathbf{x}) = 1 - \frac{1}{1 + \sum_{k=0}^{D-1} s^{D.i+k}(\mathbf{x})^2} \tag{B.29}$$

B.3.2.2 Evaluation

During this PhD, we compared several appearance modeling techniques using two criteria: (i) exploitation of the spatial context of the observations, (ii) number of modes in the modeling. The techniques which have been evaluated are the following: the single gaussian model (1G), the gaussian mixture model (GMM, [91, 102]), a block PCA technique (bPCA), an incremental singular value decomposition technique (iSVD, [21]) and the incremental PCA technique (iPCA) detailed in the previous section.

Exploitation of the spatial context The different method considered in the comparison make different uses of the spatial context of the observations. The single gaussian and gaussian mixture models work in a pixel-wise manner and do not use the context of the observations. The block PCA technique performs a classical PCA on small sets of Quad-Tree cells and hence uses the local context of the observations. Finally, the incremental SVD and incremental PCA

techniques both work on huge observation vectors and hence use the global context of the observations.

Figure 6.14 compares the ROC curves obtained using each of these method, whereas figure 6.15 compares sample change masks estimated using each technique for three different cases. These results show that the two techniques which do not use the context of the observations lead to the worst performances and that the associated change masks contain a tremendous amount of false alarms. This may be explained by the fact that, since the appearance modeling is done in a pixel-wise manner, an object composed of already observed appearance but corresponding to a change due to its structure (for instance, constructed as illustrated in figure 6.13c) would never be detected. Therefore, such pixel-wise approaches, which only base the change detection decision on the similarity between the appearances of reference and test data, cannot lead to satisfactory performances on realistic scenarios where strong appearance variations are frequent. Moreover, the ROC curves show that both techniques using the global context of observations are clearly superior to the technique using only the local context. In practice, when using the block PCA technique, we observed that most of the changes in the test data where correctly detected at their border, but their interior, which is more homogeneous and less textured, was often not detected. This may be explained by the fact that the modeling is done independently for each block and that a change is often more easily detectable by analyzing its border, where the new gradients are more intense. A possible solution would be to use wider blocks, but as the modeling uses a classic PCA scheme, we are quickly limited by the amount of memory required to compute the covariance matrices. More generally, this superiority of using the global context over using the local context can be interpreted by recognizing that the notion of change is vague and covers a wide range of possibilities. Consequently, the a priori determination of a relevant size for the local context to be used in the modeling is impossible, and hence, using the global context is preferable a priori [5].

Number of modes in the appearance model The accuracy of the appearance modeling also depends on the number of modes in the model. Indeed, the higher the number of modes in the appearance model, the closer this model match to the set of observations. In the case of the incremental PCA, the number of modes depend on the number of principal components kept in the model. Figure 6.16 compares the ROC curves obtained for various number of modes in the incremental PCA technique and in the Gaussian mixture model. These results show that the performance increase is rather small beyond 20 principal components. Moreover, table 6.3 compares the execution times and memory occupancies for various number of modes. These measures show that increasing the number of modes has little influence on the average detection time, but significantly increase the memory occupancy and the modeling time.

Diversity in reference data An interesting point when using the approach of appearance modeling is that the change detection performance is is linked to the diversity of the observations used as reference. In order to illustrate the influence of this diversity, we used two reference videos acquired under different ranges of viewpoints and different illumination conditions. We also used a test video acquired under the same illumination conditions as the first reference video, but under the same range of viewpoint as the second reference video. Figure 6.26 shows the camera trajectories for each video, and compares the image contents of the second reference video and of the test video, where changes are highlighted by black arrows.

Figure 6.27 compares the ROC curves corresponding the the cases where the change detection is performed using both reference videos or each reference video independently. These

5. Obviously, this does not entail that detecting a change in a particular area of a given image necessitates the use of every pixel of that image, only that we cannot determine a priori which pixels should be used. This point can be discussed further by considering the spatio-temporal optimization technique, presented in section B.4.1.1, which provides a way to limit the size of the context used for the change detection decision at a given pixel.

results show that, when both reference videos are used jointly, the change detection performances are superior to both other cases. This demonstrates that using the maximum available amount of information enables the most accurate modeling of the observed scene, and therefore the highest change detection performances.

B.4 Consolidation

The algorithms presented in the previous sections enable the detection of changes as test appearances deviating from reference appearance models. However, in general, all such deviations are not of interest to the image analyst, for instance those corresponding to moving cars on a road or to changing cast shadows. Consequently, in order to refine the change detection results, using prior information about the distinction between target changes and irrelevant variations might be useful.

As mentioned in introduction (section B.1), during this PhD, we focused on the detection of stationary changes corresponding to people and/or artificial objects. We therefore proposed two techniques in order to refine the detections: exploiting the spatio-temporal redundancy in the test video (section B.4.1) and performing a fine analysis of the score map (section B.4.2). Finally, in order to generalize this approach of using prior information, we also explored a third approach using relevance feedback to learn the appropriate distinction between irrelevant and possibly interesting changes in an interactive manner (section B.4.3).

B.4.1 Temporal consolidation

Computing a 3D appearance model of the observed scene enables the exploitation of the redundancy in the reference videos, in order to robustly detect changes under realistic conditions. This idea is well known and used by most background subtraction techniques. In the context of change detection in videos, the same principle may be applied in order to exploit the redundancy in the test video. However the requirement of online processing of the test video constitutes an important limitation, both in terms of available frames and of computation time. Assuming that the changes to be detected are stationary, we therefore proposed a method for the incremental exploitation of the spatio-temporal redundancy [16].

B.4.1.1 Spatio-temporal optimization

In order to improve the spatio-temporal consistency of the detection results in an incremental manner, we formulated a belief propagation optimization problem [10, 117, 118]. More precisely, we are interested in determining the more likely values of a set of structured non-observable states (*change* or *unchanged* labels), based on observable evidence (continuous detection scores). For that purpose, we chose to solve this problem using belief propagation rather than other possible approaches (e.g. graph cuts [20]), because its implementation is simpler and well suited to hardware acceleration since it does not require the explicit construction of the underlying graph.

Our method models the successive detection score maps, obtained using the appearance modeling algorithm presented previously, as a Markov random field (MRF). In this graph, nodes represent groups of 4×4 pixels in the detection score map corresponding to a given test frame and edges define the neighborhood system linking one given non-border node to four spatial neighbors and two temporal neighbors. Figure 5.1 illustrates the MRF graph corresponding to a few successive video frames. The following details the technical implementation of our method.

For each incremental step t, we denote by $\Omega_t = [\![t-T+1,t]\!]$ the sliding window of length T used to optimize the spatio-temporal consistency (in practice, we use $T=8$). We denote respectively by $\{I_k^{\text{test}}\}_{k\in\Omega_t}$, $\{S_k\}_{k\in\Omega_t}$ and $\{I_k^{\text{ref}}\}_{k\in\Omega_t}$ the T most recent test video frames, the corresponding detection score maps and the corresponding reference images (in practice obtained using a rendering of the 3D appearance model, from the same viewpoint as the associated test video frame). We assume that all of these images are registered with respect to the current test video frame, I_t^{test}, for which we want to improve the change detection results. In order to minimize execution times, we also assume that these images have been re-sized by a factor of $\frac{1}{4}$. We then denote by $w \times h$ their common dimensions and we define the following intervals: $\mathcal{U} = [\![0,w-1]\!]$ and $\mathcal{V} = [\![0,h-1]\!]$. In order to simplify notations, the following uses a spatio-temporal index denoted by $l = (u,v,k) \in \mathcal{L}_t = \mathcal{U} \times \mathcal{V} \times \Omega_t$.

We denote the Markov random field used at step t by $\mathcal{G}_t = \{\mathcal{N}_t, \mathcal{A}_t\}$, where $\mathcal{N}_t = \{n_l\}_{l\in\mathcal{L}_t}$ represents the node set and $\mathcal{A}_t = \{a_{l\leftrightarrow l'} = (n_l, n_{l'}) \in {\mathcal{N}_t}^2, \|l-l'\|_1 = 1\}$ represents the edge set and $\|\cdot\|_1$ is the L_1 norm. Each node $n_l, l \in \mathcal{L}$ of this graph is associated to a hidden state, denoted by $y_l \in \{0,1\}$, and to an observable evidence, denoted by $x_l = x_{u,v,k} = S_k(u,v) \in \mathbb{R}$.

The objective of the generalized belief propagation algorithm is to find the values of the hidden states $\{y_l\}_{l\in\mathcal{L}_t}$ corresponding in an optimal manner to the observed evidences $\{x_l\}_{l\in\mathcal{L}_t}$. This is done by maximizing the joint probability distribution, which can be expressed as follows using an energy:

$$\begin{aligned} p\big(\{x_l,y_l\}_{l\in\mathcal{L}_t}\big) &= \frac{1}{Z} \prod_{a_{l\leftrightarrow l'}\in\mathcal{A}_t} \Psi_{a_{l\leftrightarrow l'}}(y_l,y_{l'}) \prod_{n_l\in\mathcal{N}_t} \Phi_{n_l}(y_l,x_l) \\ &= \frac{1}{Z} \exp\Big[-E\big(\{x_l,y_l\}_{l\in\mathcal{L}_t}\big)\Big] \end{aligned} \tag{B.30}$$

$$\text{avec} \quad E\big(\{x_l,y_l\}_{l\in\mathcal{L}_t}\big) = -\sum_{a_{l\leftrightarrow l'}\in\mathcal{A}_t} \ln\, \Psi_{a_{l\leftrightarrow l'}}(y_l,y_{l'}) - \sum_{n_l\in\mathcal{N}_t} \ln\, \Phi_{n_l}(y_l,x_l) \tag{B.31}$$

where $\frac{1}{Z}$ is a normalizing factor and the terms $\Phi_{n_l}(y_l,x_l), n_l \in \mathcal{N}_t$ and $\Psi_{a_{l\leftrightarrow l'}}(y_l,y_{l'}), a_{l\leftrightarrow l'} \in \mathcal{A}_t$ represent respectively the evidence function and the interaction function. On the first hand, the evidence function Φ_{n_l} is used to link the *changed* value of a given hidden state to high values of the corresponding detection score. More precisely, the evidence function is expressed as follows:

$$\Phi_{n_l}(y_l,x_l) = \begin{cases} f_{n_l}(x_l) & \text{if } y_l = 0 \text{ (unchanged)} \\ 1 - f_{n_l}(x_l) & \text{if } y_l = 1 \text{ (changed)} \end{cases} \tag{B.32}$$

$$\text{where} \quad f_{n_l}(x_l) = \Big[c_0 \exp\big[-c_1\Delta(n_l)\big] - c_0 + 1\Big] \exp\Big(-\frac{x_l}{\tau}\Big) \tag{B.33}$$

$$\Delta(n_l) = \Delta(n_{u,v,k}) = \Big|I_k^{\text{test}}(u,v) - I_k^{\text{ref}}(u,v)\Big|$$

where τ is a detection threshold used to generate the ROC curves, and c_0 and c_1 are parameters (in practice, we use $c_0 = \frac{1}{3}$ and $c_1 = \frac{\log(2)}{30}$). On the other hand, the interaction function $\Psi_{a_{l\leftrightarrow l'}}$ is used to enforce consistency between the values of neighbor hidden states. In our method, we used this interaction function to link the relationship between two neighbor hidden states to the gradient in the difference image, which enables to modulate their estimated values whether the corresponding nodes belong to the same object or to different objects in the scene. More precisely, the interaction function is expressed as follows:

$$\Psi_{a_{l\leftrightarrow l'}}(y_l,y_{l'}) = \begin{cases} 0.95\eta\big[|\Delta(n_l) - \Delta(n_{l'})|\big] + 0.5\Big[1 - \eta\big[|\Delta(n_l) - \Delta(n_{l'})|\big]\Big] & \text{si } y_l = y_{l'} \\ 0.05\eta\big[|\Delta(n_l) - \Delta(n_{l'})|\big] + 0.5\Big[1 - \eta\big[|\Delta(n_l) - \Delta(n_{l'})|\big]\Big] & \text{si } y_l \neq y_{l'} \end{cases} \tag{B.34}$$

$$\text{where} \quad \eta(z) = \frac{\frac{\pi}{2} - \text{atan}\big[c_2(z-c_3)\big]}{\frac{\pi}{2} - \text{atan}(-c_2c_3)} \tag{B.35}$$

where $\eta(\cdot)$ is a family of scalar radial-basis function controlled by parameters c_2 and c_3 (in practice, we use $c_2 = 0.175$ and $c_3 = 25$). Figure 5.2 presents a few examples of functions from this family along with the function we used in practice (highlighted in red).

Using these two functions, the iterative algorithm of belief propagation estimates, for each node, the probability that it corresponds to a change. For that purpose, messages containing information about the relative likelihood of each state are passed from each node to its neighbors. Convergence of this message passing algorithm is only guaranteed when the graph is cycle-free, which is not the case with a Markov random field. However, many authors [117] have reported that, even in the presence of cycles, this generalized belief propagation algorithm leads to relevant solutions in practice.

B.4.1.2 Evaluation

Figure 6.17 compares the change detection performances obtained using the spatio-temporal consolidation approach presented in the previous section (in blue), with those obtained using a simple temporal smoothing of the detection score (in red) and when no temporal consolidation is done (in black). These results show that using a temporal consolidation approach significantly improves the results with respect to the raw change detection results. Moreover, they show that our approach based on spatio-temporal optimization leads to a considerable improvement when compared to the baseline version using a temporal smoothing of the scores.

However, this superior effectiveness is obtained at the expense of a lower efficiency, as reported by table 6.5. The algorithmic complexity of the spatio-temporal consolidation algorithm is in $O(N_{\text{iter}} \cdot N_{\text{voisins}} \cdot w \cdot h \cdot T)$, where N_{iter} is the number of message passing iterations and N_{voisins} is the number of neighbors per node. In practice, we have $N_{\text{iter}} = 4$ and $N_{\text{voisins}} = 6$. Consequently, since the dimensions of the considered images, w and h, are divided by a factor of 4 with respect to those used by the temporal smoothing method, we obtain an execution time approximately 12 times higher. Even though our effort has not focused on this point, it may be noted that the belief propagation algorithm can be significantly accelerated using an efficient implementation [42] or using hardware acceleration [23].

B.4.2 Binarization

Most approaches in the literature use a simple thresholding operation for the *binarization* task, which consists of converting the continuous detection scores into binary, *changed* or *unchanged* labels. However, the thresholding operation is a naive way to address the binarization problem and causes a loss of information, as demonstrated by the two example in figure 5.3: thresholding the two score maps 1a) and 2a), which are clearly different, leads to the same binary change masks. These two toy cases have been designed to behave similarly to the different behavior observed in the score map respectively in the presence of significant changes or illumination effects. Indeed, the former lead to very strong variations of the detection scores in the score map, whereas the latter lead to slower and smoother score variations.

We therefore proposed to exploit this different behavior in order to further refine the change detection results through the binarization task. For that purpose, we performed the continuous-to-binary conversion using the maximally stable extremal region (MSER) extraction algorithm [37]. Indeed, the notion of region stability as defined by the MSER [75] is very appropriate to distinguish the two types of behavior observed respectively in the presence of significant changes or of illumination effects. In addition, using the MSER algorithm in order to detect changed regions leads to very stable detections in the test video, which will help the interactive learning mechanism presented in the next section.

Figure 6.19 compares the change detection performances obtained when performing the binarization task using the thresholding operation or using the MSER extraction algorithm. Note that the curve corresponding to the MSER extraction algorithm is truncated, which is due

to the fact that the limited value range of the parameter (Δ, see [75]) used to vary the false positive and true positive rates did not enabled reaching false positive rates greater than 13%. However, we are mainly interested in the low false positive rate values, hence the truncation of this curve is not important. Figure 6.20 also compares change masks estimated with these two methods, thresholding (a) with two different threshold values and MSER (b), with respect to the ground truth (c). Again, this improved effectiveness is obtained at the expense of a lower efficiency, as reported by table 6.6.

B.4.3 Relevance feedback

The change detection approach composed of all the algorithm presented in the previous section provides results which are satisfactory but still containing false positives. An example of irrelevant variation causing challenging false positives are hard shadows, which are often completely black and associated to strong gradients at their borders. All these residual false positives could be removed one by one using appropriate ad-hoc mechanisms, however, this approach lead to inefficient and hardly adaptable processing. However, this problem could be addressed using a more flexible approach, by learning directly form the image analyst, in an interactive manner, the distinction between significant changes and irrelevant variations. Therefore, we developed a lightweight and intuitive relevance feedback mechanism able to learn from interactions with the image analyst.

B.4.3.1 Principle

The main motivation for introducing this relevance feedback mechanism is to reduce the false positive rate. Therefore, we proposed to use the set of all algorithm presented in the previous sections as the process providing initial detections. These initial detections are then examined by the user, who provides a feedback about their relevance, by giving binary annotations. These annotations are then utilized to train an automatic classifier, which is able to learn the underlying distinction between the true positives and the false positives, as defined by the user himself. Hence, the objective is to reduce the false positive rate and to maintain the true positive rate. Note however that this relevance feedback approach will not be able to retrieve target changes missed by the initial change detection process. Indeed, such a feature would require doing a brand new detection of changes each time the user provides annotations, which would have led to an inefficient and annoying system. The pseudo-code of our relevance feedback mechanism is given by algorithm 5.1.

In order to characterize each detected region, we proposed a 22-dimensional descriptor designed to distinguish significant changes from irrelevant variations. This descriptor enables analyzing the evolution of a given region between the reference and test observations, for instance by using measures on the score map or computed jointly on the reference and test images. For a given region, the associated descriptor is based on measures related to the intensity (average intensity difference on the region between reference and test images in each color channel, dominant value of the grayscale intensity ratio [112]), the color (average hue shift on the region [47], dominant value of the fraction of the observations independent from the illumination color [44]), the gradient (average gradient at the border of the region in the score map and in each color channel of the difference image), the MSER stability (for several values of the Δ parameter) and the shape (elongation, non-compacity, smoothness of the region boundary). Considered individually with respect to a given region, each of these measures are weak indicators of the belonging of this region to one of the two considered classes. However, when used jointly, they lead to a good prediction accuracy. Figure 5.5 presents the result of a 2-dimensional discriminant analysis computed on an annotated set of such descriptors, where blue points correspond to *unchanged* regions and red points to *changed* regions. This graph shows that the two modes

Inputs: Regions $\{Q_j^t\}_{j\in[\![0,N_t-1]\!]}$ detected as possible changes on image I_t
Outputs: Classification of the detected regions in two classes, Classifier $\mathcal{C}_t$ trained on the user annotation provided for image I_t

For $j \in [\![0, N_t - 1]\!]$ **Do**
  Extract descriptor $\mathbf{d}_j^t$ corresponding to region Q_j^t
  If $t > 0$ **Then**
    Predict class $\hat{y}_j^t \in \{0,1\}$ of region Q_j^t using descriptor $\mathbf{d}_j^t$ and classifier $\mathcal{C}_{t-1}$
  Else
    Assign Q_j^t to the clas of significant changes ($\hat{y}_j^t \leftarrow 1$)
  End If
End For
Display detections and their classes on image I_t
For $J \subset [\![0, N_t - 1]\!]$ chosen by the user, receive true classes $\{y_j^t\}_{j\in J}$ associated to regions $\{Q_j^t\}_{j\in J}$
If $J \neq \varnothing$ **Then**
  Train classifier $\mathcal{C}_t$ using current and previous annotations $\left\{ \left(y_j^k, \mathbf{d}_j^k \right) \right\}_{j\in J, k\in[\![0,t]\!]}$
End If

ALGORITHM B.5: *Relevance feedback algorithm used in the context of online change detection in a video.*

are well distinguishable and that, even though a linear separation of the two classes is likely to be impossible [6], a linear classifier will lead to satisfactory error rates.

Since the amount of annotated data will likely be limited, due to the annotation effort required from the user, we chose to use a descriptor with relatively few dimensions in order to minimize the impact on the quality of the classification. Moreover, we used a classification based on linear support vector machines (SVM), whose margin maximization principle guaranties a good generalization capacity even in cases where only few training samples are available. As the test video must be processed online, we train a new classifier each time new annotations are provided by the user, which in practice is performed very quickly, especially since the amount of training data will likely remain small.

B.4.3.2 Evaluation

Offline evaluation We started by evaluating the ideal performances obtained in an offline manner. Figure 6.21 presents the evolution of the classifier's error rate (i.e. average of the classifier's false positive and false negative rates) as a function of the annotation iterations. Here, we assumed that the user annotated 0.16% of the regions at each iteration, leading to a total of 5% of annotated regions after 30 iterations. Before the first annotation iteration, all detected regions are classified as changes, hence the classifier's false positive rate is 1 and its false negative rate is 0. Therefore, at iteration 0, the classifier's error rate is 0.5. The curves show that the error rate drop significantly with the very first annotation iteration and keeps decreasing with the following iterations. Logically, these good performances on the classification problem lead to good performances on the change detection problem, as demonstrated by figure 6.22. This graph shows that the first annotation iteration leads to a significant reduction of the false positive rate, at the expense of a moderate reduction of the true positive rate, and that subsequent iterations further reduce the false positive rate while increasing the true positive rate.

Online evaluation In practice, our change detection approach aims at the online processing of the test video, which hence requires that the relevance feedback is applied in an online man-

6. Strictly speaking, the non linear-separability cannot be deduced from a 2-dimensional projection of the point clouds, as the separability could arise from additional dimensions.

ner. This means that the annotation provided by the user on a given image cannot modify the results obtained on previous images, which therefore has an influence on the ideal performances presented above.

In particular, figure 6.23 compares the performances obtained using different annotation policies: annotation on the first 33 images (denoted 0:1:33), annotation on one out of 3 images among the 99 first images (denoted 0:3:99), etc. This graph shows that the best performance is obtained when annotations are provided regularly during the video, which guarantees that the training samples are diverse and up-to-date enough to help analyzing the detection at each incremental processing step. Finally, table 6.7 presents the average execution times associated to each task of the relevance feedback mechanism: descriptor extraction, prediction and training of the classifier. These low times enable a fluid and transparent execution of the interactive mechanism.

B.5 Future work

To conclude, as mentioned in the introduction section, the problem of detecting changes in aerial videos with arbitrary trajectories is challenging and involves solving many different sub-problems in order to obtain a usable processing chain. During this PhD, we chose to orient our studies towards a semi-automatic approach aiming to assist an image analyst in his analyzis task. Therefore, our approach occasionally request input from the user, in particular for geo-loccalizing the reference videos (section B.2.1.1) and for the interactive consolidation of detection results (section B.4.3). However, in order to minimize the effort required from the user, we dedicated ourselves in maximizing the amount of information extracted from the available data and we proposed innovative algorithms such as visual servoing geo-localization (section B.2.1.2), spatio-temporal consolidation of the results (section B.4.1) or MSER-based binarization (section B.4.2). Figure 7.1 presents the diagram summarizing the global change detection approach developed during this PhD.

This global approach leads to very promising performance and was for instance able to generate less than 1 false positive every 3 minutes, on a 5 fps video (see precision/recall analysis using image evaluation criterion, reported in figure 6.29, on data presented in figure 6.28). Such a false positive rate might seem high, but from an operational point of view, this is very interesting since it enables filtering a considerable amount of irrelevant data with few user effort.

The work presented here also enabled the identification of many promising leads for improvement or for extension of our approach.

Extension of the geo-localization algorithms Geo-localization of the considered images is essential for detecting changes using our approach, but can also be applied to many other problems. The proposed algorithms may therefore be extended in many ways.

First, related to the pose interpolation algorithm, it would be interesting to minimize the effort required from the user. For that purpose, a first step would be to automatically determine the optimal number and position of key-frames, for the geo-localization of a given video. This could for instance be done by analyzing the similarity between the video frames and to find the key-frames minimizing the number of key-frames (in order to reduce user annotation effort) and maximizing the similarity between successive key-frames (in order to ensure good accuracy for the estimation of the trifocal tensor). Another lead may be to use videos already geo-localized on the same area to reduce the number of images requiring manual calibration. Finally, various ways could be considered to improve the accuracy of the geo-localization, for instance by using meta-data from the aircraft to initialize the estimation, or by enforcing a better continuity for

the final trajectory. It may be noted that, operationally, designing an automatic approach for these tasks would be of considerable interest in the context of geo-localized data mining.

Related to the visual servoing algorithm, it would be interesting to find a registration algorithm both efficient and able to robustly estimate the transformation between a real image and a synthetical image obtained by rendering a 3D model. In particular, when using a 3D appearance model, it might be possible to exploit the appearance models rather than mere average observations. Another possibility would be to improve the estimation accuracy, in the general case, where the intrinsic parameters are estimated at each frame.

Improved appearance modeling Appearance modeling is the center technique allowing our approach to detect significant changes and further work on this topic would certainly lead to considerable improvements.

Firstly, it might be interesting to utilize the hierarchical structure of the 3D appearance model, in order to compute appearance models matching a wide range of ground resolution in the observations. Such a hierarchical approach would enable the efficient processing of large homogeneous areas in the scene model and also the appropriate handling of test videos acquired with a ground resolution superior to the ground resolution in the reference videos.

Secondly, we showed that the incremental PCA technique was the one providing the best performance, among the techniques we compared. We mentioned that this technique was also very well suited to the change detection problem in the context of videos, due its incremental compression capability and to its robustness to missing data and aberrant observations. However, PCA remain very simple and, in the context of satellite imagery, many authors have obtained better results with other techniques, such as the canonical correlation analysis (CCA) [85] or independent component analysis (ICA) [72]. Incremental versions of these techniques may exist (for instance for CCA [110]), but to our knowledge, no application to appearance modeling have been published.

Finally, it might be considered to use other change detection criteria, in order to achieve better robustness to irrelevant variations (e.g. illumination effects), such as edges, gradients or mutual information.

Offline change detection The evaluation results presented here show that performances could be considerably improved if the detection of changes was performed in an offline manner, that is, once all test video frames are available.

First, we saw that the reprojection error obtained using the offline geo-localization was better than the one obtained using the online geo-localization, which led to differences between the change detection performances. Secondly, an offline change detection could enable a sound exploitation of the redundancy present in the test video, for instance allowing the comparison between the set of all reference images and the set of all test images. Finally, without the online constraint, the relevance feedback mechanism could be extended, for instance to integrate active learning techniques, and could therefore lead to significantly better performances.

Acceleration of the algorithms The last point, less interesting from a fundamental point of view but operationally much more important, would be to focus on the acceleration of the proposed algorithms (in particular via GPU) in order to reach real-time execution. More precisely, the classical and inverse ray-casting algorithm, which are used extensively in our approach, are known to be well suited to hardware acceleration. Similarly, accelerating the spatio-temporal consolidation algorithm based on belief propagation would make it much more interesting, in light of the excellent change detection performances it provides.

Bibliographie

[1] Aach, T. et A. Kaup. 1995, «Bayesian algorithms for adaptive change detection in image sequences using markov random fields», *Signal Processing-Image Communication*, vol. 7, n° 2, p. 147–160. (Cité pages 37, 38).

[2] Aach, T., A. Kaup et R. Mester. 1993, «A statistical framework for change detection in image sequences», *Signal Processing*, vol. 31, n° 2, p. 165–180. (Cité pages 15, 31, 32, 154).

[3] Bajcsy, R. et S. Kovacic. 1989, «Multiresolution elastic matching», *Computer vision, graphics, and image processing*, vol. 46, n° 1, p. 1–21. (Cité page 22).

[4] Basseville, M. et I. Nikiforov. 1993, *Detection of abrupt changes : theory and application*, vol. 104, Prentice Hall Englewood Cliffs, NJ. (Cité pages 4, 153).

[5] Bay, H., T. Tuytelaars et L. Van Gool. 2006, «Surf : Speeded up robust features», *Proceedings of the European Conference on Computer Vision*, p. 404–417. (Cité pages 51, 52, 161).

[6] Beauchemin, S. et J. Barron. 1995, «The computation of optical flow», *ACM Computing Surveys (CSUR)*, vol. 27, n° 3, p. 433–466. (Cité pages 22, 66).

[7] Benedek, C. et T. Sziranyi. 2009, «Change detection in optical aerial images by a multilayer conditional mixed markov model», *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, n° 10, p. 3416–3430. (Cité pages 15, 31, 32).

[8] Benedek, C., T. Sziranyi, Z. Kato et J. Zerubia. 2009, «Detection of object motion regions in aerial image pairs with a multilayer markovian model», *IEEE Transactions on Image Processing*, vol. 18, n° 10, p. 2303–2315. (Cité pages 15, 31, 32).

[9] Bhaskaran, V. et K. Konstantinides. 1997, *Image and video compression standards : algorithms and architectures*, Springer. (Cité pages 4, 153).

[10] Bishop, C. 2006, «Graphical models», dans *Pattern Recognition and Machine Learning*, vol. 4, chap. 8, Springer, p. 359–422. (Cité pages 86, 87, 176).

[11] Black, M., D. Fleet et Y. Yacoob. 2000, «Robustly estimating changes in image appearance», *Computer Vision and Image Understanding*, vol. 78, p. 8–31. (Cité pages 15, 25, 26, 27, 29, 31, 32, 156).

[12] Bourdis, N. 2011, «Aicd change detection dataset», Base de données hébergée par Computer Vision Online. URL `http://www.computervisiononline.com/dataset/change-detection-dataset`. (Cité pages 11, 158).

[13] Bourdis, N. et D. Marraud. 2013, «Procédé d'analyse de régions géographiques et de détection de zones d'intérêt», Brevet. (Cité pages 11, 158).

[14] Bourdis, N., D. Marraud et H. Sahbi. 2011, «Constrained optical flow for aerial image change detection», dans *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium*, p. 4176–4179. (Cité pages 11, 21, 22, 23, 62, 64, 65, 156, 158).

[15] Bourdis, N., D. Marraud et H. Sahbi. 2012, «Camera pose estimation using visual servoing for aerial video change detection», dans *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium*, p. 3459–3462. (Cité pages 11, 16, 17, 50, 158).

[16] Bourdis, N., D. Marraud et H. Sahbi. 2012, «Spatio temporal interaction for aerial video change detection», dans *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium*, p. 2253–2256. (Cité pages 11, 37, 38, 86, 87, 89, 90, 156, 158, 176).

[17] Bourdis, N., H. Sahbi et D. Marraud. 2011, «Exploitation de vidéos aériennes multiples pour la détection de changements», cahier de recherche, Telecom ParisTech. (Cité pages 11, 158).

[18] Bourdis, N., H. Sahbi et D. Marraud. 2013, «Reliable detection of significant changes in aerial videos with arbitrary trajectories», *IEEE Transactions on Geoscience and Remote Sensing*. (Cité pages 11, 17, 23, 28, 41, 46, 55, 66, 104, 154, 158, 160, 161, 166, 167).

[19] Bouwmans, T. 2009, «Subspace learning for background modeling : A survey», *Recent Patent On Computer Science*, vol. 2, n° 3, p. 223–234. (Cité page 34).

[20] Boykov, Y. et O. Veksler. 2006, «Graph cuts in vision and graphics : Theories and applications», *The Handbook of Mathematical Models in Computer Vision. Springer*. (Cité pages 33, 88, 176).

[21] Brand, M. 2002, «Incremental singular value decomposition of uncertain data with missing values», dans *Proceedings of the European Conference on Computer Vision*, p. 707–720. (Cité pages 31, 35, 36, 118, 174).

[22] Breunig, M., H. Kriegel, R. Ng et J. Sander. 2000, «LOF : identifying density-based local outliers», *Sigmod Record*, vol. 29, n° 2, p. 93–104. (Cité pages 31, 33, 34).

[23] Brunton, A., C. Shu et G. Roth. 2006, «Belief propagation on the gpu for stereo vision», dans *Proceedings of the 3rd Canadian Conference on Computer and Robot Vision*, p. 76–76. (Cité pages 91, 178).

[24] Buchanan, A. 2009, «Novel view synthesis for change detection», dans *Proceedings of the 2009 Conference of Electro Magnetic Remote Sensing Defence Technology Centre*. (Cité pages 14, 16, 18, 21, 23, 42, 156).

[25] Carlotto, M. 2007, «Detecting change in images with parallax», dans *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, vol. 6567, p. 42. (Cité pages 15, 21, 22, 23, 156).

[26] Chandola, V., A. Banerjee et V. Kumar. 2009, «Anomaly detection : A survey», *ACM Computing Surveys (CSUR)*, vol. 41, n° 3, p. 15. (Cité pages 31, 33).

[27] Chen, Y., C. Chen, C. Huang et Y. Hung. 2007, «Efficient hierarchical method for background subtraction», *Pattern Recognition*, vol. 40, n° 10, p. 2706–2715. (Cité pages 31, 35, 36).

[28] Cho, J. et S. Kim. 2005, «Object detection using multi-resolution mosaic in image sequences», *Signal Processing : Image Communication*, vol. 20, n° 3, p. 233–253. (Cité pages 16, 18, 19).

[29] Clifton, C. 2003, «Change detection in overhead imagery using neural networks», *Applied Intelligence*, vol. 18, n° 2, p. 215–234. (Cité pages 15, 21, 31, 36).

[30] Cluff, S. 2009, *A Unified Approach to GPU-Accelerated Aerial Video Enhancement Techniques*, Phd, Brigham Young University. (Cité page 22).

[31] Cortes, C. et V. Vapnik. 1995, «Support-vector networks», *Machine Learning*, vol. 20, p. 273–297. (Cité page 100).

[32] Crispell, D. 2010, *A Continuous Probabilistic Scene Model for Aerial Imagery*, Phd, Brown University. (Cité pages 16, 18, 19, 21, 23, 40, 42, 66, 71, 156, 169).

[33] Crispell, D., J. Mundy et G. Taubin. 2008, «Parallax-free registration of aerial video», dans *Proceedings of the British Machine Vision Conference*, p. 73.1–73.10. (Cité pages 16, 17, 23, 41, 108, 109, 110, 111, 112, 113, 158, 160, 162, 163, 164, 167).

[34] Crispell, D., J. Mundy et G. Taubin. 2012, «A variable-resolution probabilistic three-dimensional model for change detection», *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, n° 2, p. 489–500. (Cité pages 16, 18, 19, 21, 23, 40, 42, 71, 83, 118, 137, 155, 156, 157, 170, 172).

[35] Dame, A. et E. Marchand. 2012, «Second-order optimization of mutual information for real-time image registration», *IEEE Transactions on Image Processing*, vol. 21, n° 9, p. 4190–4203. (Cité page 142).

[36] Dempster, A., N. Laird et D. Rubin. 1977, «Maximum likelihood from incomplete data via the EM algorithm», *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, n° 1, p. 1–38. (Cité pages 29, 32).

[37] Donoser, M. et H. Bischof. 2006, «Efficient maximally stable extremal region (mser) tracking», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, p. 553–560. (Cité pages 92, 93, 178).

[38] Elgammal, A., R. Duraiswami, D. Harwood et L. Davis. 2002, «Background and foreground modeling using nonparametric kernel density estimation for visual surveillance», *Proceedings of the IEEE*, vol. 90, n° 7, p. 1151–1163. (Cité pages 25, 27, 28, 31, 34, 35, 54, 57, 157, 165, 166).

[39] Farneback, G. 2003, «Two-frame motion estimation based on polynomial expansion», *Image Analysis*, p. 363–370. (Cité pages 22, 63, 64).

[40] Fattal, R. 2008, «Single image dehazing», dans *ACM Transactions on Graphics*, vol. 27, p. 72. (Cité pages 29, 30).

[41] Faugeras, O. et Q. Luong. 2004, *The geometry of multiple images : the laws that govern the formation of multiple images of a scene and some of their applications*, MIT press. (Cité page 22).

[42] Felzenszwalb, P. et D. Huttenlocher. 2004, «Efficient belief propagation for early vision», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, p. 261–268. (Cité pages 91, 178).

[43] Finkel, R. et J. Bentley. 1974, «Quad trees a data structure for retrieval on composite keys», *Acta informatica*, vol. 4, n° 1, p. 1–9. (Cité pages 17, 66, 167).

[44] Finlayson, G., S. Hordley, C. Lu et M. Drew. 2006, «On the removal of shadows from images», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, n° 1, p. 59–68. (Cité pages 25, 27, 28, 54, 58, 150, 157, 165, 166, 179).

[45] Fischler, M. et R. Bolles. 1981, «Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography», *Communications of the ACM*, vol. 24, n° 6, p. 381–395. (Cité pages 16, 51, 52, 161).

[46] Gao, J. 2009, *Digital Analysis of Remotely Sensed Imagery*, McGraw Hill Professional. (Cité pages 1, 152).

[47] Gevers, T. et W. M. Smeulders. 1999, «Color based object recognition», *Pattern recognition*, vol. 32, n° 3, p. 453–464. (Cité pages 25, 27, 28, 54, 57, 59, 97, 157, 165, 166, 179).

[48] Gueguen, L., S. Cui, G. Schwarz et M. Datcu. 2010, «Multitemporal analysis of multisensor data : Information theoretical approaches», dans *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium*, p. 2559–2562. (Cité pages 15, 21, 31, 32).

[49] Gueguen, L., M. Pesaresi, D. Ehrlich et L. Lu. 2011, «Urbanization analysis by mutual information based change detection between spot 5 panchromatic images», dans *International Workshop on the Analysis of Multi-temporal Remote Sensing Images*, p. 157–160. (Cité page 138).

[50] Guttman, A. 1984, *R-trees : a dynamic index structure for spatial searching*, ACM. (Cité pages 67, 167).

[51] Haralick, R. et L. Shapiro. 1993, *Computer and robot vision*, Addison-Wesley Pub. Co. (Cité page 37).

[52] Hartley, R. et A. Zisserman. 2000, *Multiple view geometry*, 2e éd., Cambridge university press. (Cité pages 22, 23, 46, 47, 48, 50, 146, 160, 161).

[53] Hayman, E. et J. Eklundh. 2003, «Statistical background subtraction for a mobile observer», dans *Proceedings of the IEEE International Conference on Computer Vision.*, p. 67–74. (Cité pages 16, 18, 19).

[54] He, K., J. Sun et X. Tang. 2009, «Single image haze removal using dark channel prior», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, p. 1956–1963. (Cité pages 29, 30).

[55] He, Z., X. Xu et S. Deng. 2003, «Discovering cluster-based local outliers», *Pattern Recognition Letters*, vol. 24, n° 9-10, p. 1641–1650. (Cité pages 31, 33).

[56] Heas, P. et M. Datcu. 2005, «Modeling trajectory of dynamic clusters in image time-series for spatio-temporal reasoning», *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, n° 7, p. 1635–1647. (Cité pages 15, 21, 37).

[57] Hsu, Y., H. Nagel et G. Rekers. 1984, «New likelihood test methods for change detection in image sequences», *Computer vision, graphics, and image processing*, vol. 26, n° 1, p. 73–106. (Cité pages 31, 32).

[58] Irani, M., P. Anandan, J. Bergen, R. Kumar et S. Hsu. 1996, «Efficient representations of video sequences and their applications», *Signal Processing*, vol. 8, n° 4, p. 327–351. (Cité pages 16, 18).

[59] Kaewtrakulpong, P. et R. Bowden. 2001, «An improved adaptive background mixture model for realtime tracking with shadow detection», *Proceedings of the European Workshop on Advanced Video Based Surveillance Systems*, vol. 1, n° 3. (Cité pages 25, 26, 31, 34, 35, 156).

[60] Kay, S. 1993, *Fundamentals of Statistical Signal Processing*, Prentice-Hall PTR (Englewood Cliffs, NJ). (Cité page 31).

[61] Kim, K., T. Chalidabhongse, D. Harwood et L. Davis. 2005, «Real-time foreground-background segmentation using codebook model», *Real-Time Imaging*, vol. 11, n° 3, p. 172–185. (Cité pages 31, 34, 35, 155, 156).

[62] Kumar, R., P. Anandan et K. Hanna. 1994, «Shape recovery from multiple views : A parallax based approach», *Proceedings of the IEEE Int. Conf. Pattern Recognition.* (Cité page 62).

[63] Kumar, R., P. Anandan, M. Irani, J. Bergen et K. Hanna. 1995, «Representation of scenes from collections of images», dans *Proceedings of ICCV'95 IEEE Workshop on Representation of Visual Scenes*, p. 10–17. (Cité pages 21, 22, 42, 156).

[64] Kumar, R., H. Sawhney, S. Samarasekera, S. Hsu, H. Tao, Y. Guo, K. Hanna, A. Pope, R. Wildes, D. Hirvonen et al.. 2001, «Aerial video surveillance and exploitation», *Proceedings of the IEEE*, vol. 89, n° 10, p. 1518–1539. (Cité pages 1, 3, 152).

[65] Lee, C. et S. Kim. 2003, «Multi-resolution mosaic construction using resolution maps», *Visual Content Processing and Representation*, vol. 2849, p. 180–187. (Cité pages 16, 18, 19, 70, 168).

[66] Leutenegger, S., M. Lopez et J. Edgington. 1997, «Str : A simple and efficient algorithm for r-tree packing», dans *Proceedings of the 13th International Conference on Data Engineering (1997)*, p. 497–506. (Cité pages 69, 168).

[67] Li, Y. 2004, «On incremental and robust subspace learning», *Pattern recognition*, vol. 37, n° 7, p. 1509–1518. (Cité pages 31, 35, 36, 42, 79, 80, 156, 157, 172, 173).

[68] Lillesand, T., R. Kiefer et J. Chipman. 2004, *Remote sensing and image interpretation*, Ed. 5, John Wiley & Sons Ltd. (Cité pages 1, 152).

[69] Lowe, D. G. 1999, «Object recognition from local scale-invariant features», dans *Proceedings of the IEEE International Conference on Computer Vision.*, p. 1150. (Cité pages 16, 48).

[70] Luo, X., D. Wang, P. Wang et F. Liu. 2008, «A review on blind detection for image steganography», *IEEE Transactions on Signal Processing*, vol. 88, n° 9, p. 2138–2157. (Cité pages 4, 153).

[71] Mallick, S., T. Zickler, P. Belhumeur et D. Kriegman. 2006, «Specularity removal in images and videos : A PDE approach», *Proceedings of the European Conference on Computer Vision*, p. 550–563. (Cité pages 25, 26, 29, 156).

[72] Marchesi, S. et L. Bruzzone. 2009, «ICA and kernel ICA for change detection in multispectral remote sensing images», dans *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium*, vol. 2, p. II.980–II.983. (Cité pages 142, 182).

[73] Mas, J. 1999, «Monitoring land-cover changes : a comparison of change detection techniques», *International Journal on Remote Sensing*, vol. 20, n° 1, p. 139–152. (Cité pages 4, 153).

[74] Masood, A. et W. Kanwal. 2010, «Efficient representation of zooming information in videos using multi resolution mosaics», *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, p. 294–300. (Cité pages 16, 18).

[75] Matas, J., O. Chum, M. Urban et T. Pajdla. 2004, «Robust wide-baseline stereo from maximally stable extremal regions», *Image and Vision Computing*, vol. 22, n° 10, p. 761–767. (Cité pages 92, 157, 178, 179).

[76] Matlin, E. et P. Milanfar. 2012, «Removal of haze and noise from a single image», dans *IS&T/SPIE Electronic Imaging*, p. 82960T. (Cité pages 29, 30).

[77] Mattyus, G., C. Benedek et T. Sziranyi. 2010, «Multi-target tracking on aerial videos», dans *ISPRS Istanbul Workshop 2010 on Modeling of optical airborne and spaceborne Sensors*. (Cité pages 16, 31, 34).

[78] Miike, H., L. Zhang, T. Sakurai et H. Yamada. 1999, «Motion enhancement for preprocessing of optical flow detection and scientific visualization», *Pattern Recognition Letters*, vol. 20, n° 5, p. 451–461. (Cité pages 55, 166).

[79] Mittal, A. et D. Huttenlocher. 2000, «Scene modeling for wide area surveillance and image synthesis», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, p. 160–167. (Cité pages 16, 18, 19, 21, 22, 42, 156).

[80] Mittal, A. et N. Paragios. 2004, «Motion-based background subtraction using adaptive kernel density estimation», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, p. II.302–II.309. (Cité pages 31, 34, 35).

[81] Monnin, D. et E. Bieber. 2009, «Change detection for securing routes», *Revue de l'électricité et de l'électronique (REE)*, vol. 9, p. 49–54. (Cité pages 15, 16, 25, 27).

[82] Nadimi, S. et B. Bhanu. 2004, «Physical models for moving shadow and object detection in video», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, n° 8, p. 1079–1087. (Cité pages 25, 26, 156).

[83] Narasimhan, S. et S. Nayar. 2001, «Removing weather effects from monochrome images», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, p. II.186–II.193. (Cité page 29).

[84] Nava, F., A. Nava, J. Lamolda, M. Redondo et L. Bruzzone. 2005, «Change detection for remote sensing images with graph cuts», dans *Conference on Image and Signal Processing for Remote Sensing*, vol. 5982, SPIE, p. 59 820Q.1–59 820Q.14. (Cité pages 15, 31, 32, 154).

[85] Nielsen, A., K. Conradsen et J. Simpson. 1998, «Multivariate alteration detection (MAD) and MAF postprocessing in multispectral, bitemporal image data : New approaches to change detection studies», *Remote Sensing of Environment*, vol. 64, n° 1, p. 1–19. (Cité pages 31, 32, 142, 182).

[86] Papadimitriou, S., H. Kitagawa, P. Gibbons et C. Faloutsos. 2003, «LOCI : fast outlier detection using the local correlation integral», dans *Proceedings of the 19th International Conference on Data Engineering*, p. 315–326. (Cité pages 31, 33, 34).

[87] Piccardi, M. 2004, «Background subtraction techniques : a review», dans *IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, p. 3099–3104. (Cité page 34).

[88] Ping, T., S. Lin, L. Quan et H. Y. Shum. 2003, «Highlight removal by illumination-constrained inpainting», dans *Proceedings of the IEEE International Conference on Computer Vision.*, p. 164–169. (Cité page 29).

[89] Pokrajac, D., A. Lazarevic et L. Latecki. 2007, «Incremental local outlier detection for data streams», dans *IEEE Symposium on Computational Intelligence and Data Mining*, p. 504–515. (Cité pages 31, 33, 34).

[90] Pollard, T. 2009, *Comprehensive 3-d Change Detection Using Volumetric Appearance Modeling*, Phd, Brown University. (Cité pages 25, 29, 30, 40, 41, 66, 83, 156, 158).

[91] Pollard, T. et J. Mundy. 2007, «Change detection in a 3-d world», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, p. 1–6. (Cité pages 14, 16, 18, 19, 21, 23, 25, 41, 42, 118, 155, 156, 158, 167, 174).

[92] Primdahl, K., I. Katz, O. Feinstein, Y. Mok, H. Dahlkamp, D. Stavens, M. Montemerlo et S. Thrun. 2005, «Change detection from multiple camera images extended to non-stationary cameras», *Proceedings of Field and Service Robotics*. (Cité pages 14, 15, 16, 21, 25, 27, 41, 158).

[93] Pritt, M. et K. LaTourette. 2011, «Stabilization and georegistration of aerial video over mountain terrain by means of lidar», dans *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium*, p. 4046–4049. (Cité pages 16, 17, 25, 156).

[94] Pritt, M. et K. LaTourette. 2012, «Dense 3d reconstruction for video stabilization and georegistration», dans *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium*, p. 6737–6740. (Cité pages 71, 170).

[95] Radke, R., S. Andra, O. Al-Kofahi et B. Roysam. 2005, «Image change detection algorithms : a systematic survey», *IEEE Transactions on Image Processing*, vol. 14, n° 3, p. 294–307. (Cité pages 5, 14, 15, 24, 31, 32, 37, 40, 153, 158, 159).

[96] Ratsch, G., S. Mika, B. Scholkopf et K. R. Muller. 2002, «Constructing boosting algorithms from SVMs : an application to one-class classification», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, n° 9, p. 1184–1199. (Cité pages 31, 33).

[97] Rocchio, J. 1971, «Relevance feedback in information retrieval», dans *The SMART Retrieval System : Experiments in Automatic Document Processing*, édité par G. Salton, chap. 14, Prentice-Hall Series in Automatic Computation, Prentice-Hall, Englewood Cliffs NJ, p. 313–323. (Cité page 94).

[98] Rowe, N. et L. Grewe. 2001, «Change detection for linear features in aerial photographs using edge-finding», *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, n° 7, p. 1608–1612. (Cité pages 25, 27).

[99] Salvador, E., A. Cavallaro et T. Ebrahimi. 2001, «Shadow identification and classification using invariant color models», dans *Proceedings of the IEEE Int. Conf. Acoustics Speech Signal Processing.*, vol. 3, p. 1545–1548. (Cité pages 25, 27).

[100] Sarkar, S. et K. Boyer. 1996, «Quantitative measures of change based on feature organization : Eigenvalues and eigenvectors», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, p. 478–483. (Cité pages 31, 32).

[101] Sjahputera, O., C. Davis, B. Claywell, N. Hudson, J. Keller, M. Vincent, Y. Li, M. Klaric et C. Shyu. 2008, «Geocdx : An automated change detection & exploitation system for high resolution satelite imagery», dans *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium*, vol. 5, p. V.467–V.470. (Cité pages 37, 38, 43, 157).

[102] Stauffer, C. et W. Grimson. 2000, «Learning patterns of activity using real-time tracking», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, n° 8, p. 747–757. (Cité pages 4, 31, 34, 35, 36, 37, 42, 77, 83, 153, 155, 156, 174).

[103] Stennett, C. et R. Evans. 2009, «Visual change detection for route monitoring», dans *Proceedings of the 2009 Conference of Electro Magnetic Remote Sensing Defence Technology Centre*. (Cité pages 15, 16, 41, 158).

[104] Stringa, E. 2000, «Morphological change detection algorithms for surveillance applications», dans *Proceedings of the British Machine Vision Conference*, p. 402–412. (Cité page 37).

[105] Tax, D. 2001, *One-class classification : concept-learning in the absence of counter-examples*, thèse de doctorat, Delft University of Technology. (Cité pages 31, 33).

[106] Toth, D., T. Aach et V. Metzler. 2000, «Illumination-invariant change detection», dans *4th IEEE Southwest Symposium on Image Analysis and Interpretation*, p. 3. (Cité pages 15, 25, 27).

[107] Toyama, K., J. Krumm, B. Brumitt et B. Meyers. 1999, «Wallflower : Principles and practice of background maintenance», dans *Proceedings of the IEEE International Conference on Computer Vision.*, vol. 1, p. 255–261. (Cité pages 31, 35).

[108] Tsai, D. et S. Lai. 2009, «Independent component analysis-based background subtraction for indoor surveillance», *IEEE Transactions on Image Processing*, vol. 18, n° 1, p. 158–167. (Cité pages 31, 32).

[109] Turk, M. et A. Pentland. 1991, «Eigenfaces for recognition», *Cognitive Neuroscience*, vol. 3, n° 1, p. 71–86. (Cité pages 79, 172).

[110] Via, J., I. Santamaria et J. Perez. 2005, «A robust RLS algorithm for adaptive canonical correlation analysis», dans *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, p. iv/365–iv/368. (Cité pages 142, 182).

[111] Watanabe, S. et K. Miyajima. 2001, «Detecting building changes using the epipolar constraint from aerial images taken at different positions», dans *Proceedings of the IEEE Int. Conf. Image Processing.*, vol. 2, p. 201–204. (Cité pages 15, 21, 22, 156).

[112] Watanabe, S., K. Miyajima et N. Mukawa. 1998, «Detecting changes of buildings from aerial images using shadow and shading model», dans *Proceedings of the Fourteenth International Conference on Pattern Recognition*, vol. 2, p. 1408–1412. (Cité pages 15, 25, 26, 96, 156, 179).

[113] Wiles, R., D. Hirvonen, S. Hsu, R. Kumar, W. Lehman, B. Matei et W.-Y. Zhao. 2001, «Video georegistration : algorithm and quantitative evaluation», dans *Proceedings of the IEEE International Conference on Computer Vision.*, vol. 2, p. 343–350. (Cité pages 16, 17, 25, 27).

[114] Woo, A. 1990, «Fast ray-box intersection», dans *Graphic Gems*, édité par A. S. Glassner, Academic Press Professional, Inc., p. 395–396. (Cité pages 75, 76, 170).

[115] Yao, A., J. Gall, C. Leistner et L. Van Gool. 2012, «Interactive object detection», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, p. 3242–3249. (Cité pages 37, 38, 40, 43, 157).

[116] Yao, J. et Z. Zhang. 2006, «Hierarchical shadow detection for color aerial images», *Computer Vision and Image Understanding*, vol. 102, n° 1, p. 60–69. (Cité pages 25, 156).

[117] Yedidia, J., W. Freeman et Y. Weiss. 2000, «Generalized belief propagation», dans *Advances in Neural Information Processing Systems*, MIT Press, p. 689–695. (Cité pages 86, 87, 91, 176, 178).

[118] Yedidia, J., W. Freeman et Y. Weiss. 2003, «Understanding belief propagation and its generalizations», dans *Exploring artificial intelligence in the new millennium*, édité par G. Lakemeyer et B. Nebel, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, p. 239–269. (Cité pages 87, 176).

[119] Yilmaz, A., O. Javed et M. Shah. 2006, «Object tracking : A survey», *ACM Computing Surveys (CSUR)*, vol. 38, n° 4, p. 13. (Cité pages 4, 153).

[120] Yin, Z. et R. Collins. 2007, «Belief propagation in a 3D spatio-temporal MRF for moving object detection», dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, p. 1–8. (Cité pages 37, 38, 42, 86, 156).

[121] Yu, W., X. Yu, P. Zhang et J. Zhou. 2008, «A new framework of moving target detection and tracking for UAV video application», dans *Proc. ISPRS Congress Beijing, Comm. III, Work. Gr. III/5*, vol. 37, p. 609–614. (Cité page 16).

[122] Zhou, X. et T. Huang. 2003, «Relevance feedback in image retrieval : A comprehensive review», *Multimedia Systems*, vol. 8, p. 536–544. (Cité page 94).

[123] Zitova, B. et J. Flusser. 2003, «Image registration methods : a survey», *Image Vis. Comput.*, vol. 21, n° 11, p. 977–1000. (Cité pages 21, 22, 51, 142).