Système multi-caméras pour l'analyse de la posture humaine

Laetitia Gond

N° d'ordre : D.U. 1922
EDSPIC : 433

Université Blaise Pascal - Clermont-Ferrand II

École Doctorale
Sciences Pour l'Ingénieur de Clermont-Ferrand

Thèse présentée par :
Laetitia Gond

pour obtenir le grade de

Docteur d'Université

spécialité : Vision pour la robotique

Système multi-caméras pour l'analyse de la posture humaine

Soutenue publiquement le 5 mai 2009 devant le jury :

M. Jean-Marc Lavest	Président
Mme. Catherine Achard	Rapporteur
M. Michel Devy	Rapporteur
M. Patrick Sayd	Encadrant
M. Thierry Chateau	Encadrant
M. Michel Dhome	Directeur de thèse

Remerciements

Je remercie tout d'abord Catherine Achard, Michel Devy et Jean-Marc Lavest d'avoir accepté de faire partie de mon jury de thèse. Je remercie également mes encadrants Michel Dhome, Thierry Chateau et Patrick Sayd de m'avoir proposé ce sujet de thèse, et pour leurs conseils tout au long de ces trois années.

Je remercie aussi le LASMEA de m'avoir accueillie et de m'avoir permis de prolonger ma thèse de bonnes conditions. Je remercie également les personnes qui ont suivi ma thèse à différentes périodes de ces trois années, Nicolas Allezard, Quo-Cuong Pham, Yoann Dhome, et Steve Bourgeois, pour ses conseils sur mes publications, sa patience et son aide sur toutes les questions informatiques. Merci à Yoann et Patrick pour les acquisitions réalisées au laboratoire qui m'ont permis de conclure ma thèse avec de jolis résultats.

Un grand merci à toutes les personnes qui m'ont écoutée et soutenue pendant les moments difficiles, à commencer par mon encadrant Patrick, Fred, toute la petite bande du LCE (Maria, Mathieu, Suresh, Karim, Maroun, Selim...), Laurent, Fabien, François... Je pense aussi à tous les collègues et amis du CEA avec qui j'ai passé de bons moments pendant ma thèse : le petit clan des filles du labo (girl power ! !) Hai, Agnes, Hanna et Julie, Caro, et les garçons aussi, Edouard, Etienne, Pierre, les Julien, Vincent, Stevens, Romain, et tous les amis pongistes de L'ASCEA. Merci aux thésards du LASMEA pour leur accueil et le courage dont ils ont fait preuve pour supporter l'"arme de démoralisation massive" venue de Paris, en particulier Laetitia, Bertrand, Manu, Alex et François.

Je remercie enfin chaleureusement Steve, Michael et mes parents pour le soutien (moral et matériel !) qu'ils m'ont apporté, en particulier à la fin de ma thèse.

Résumé

L'analyse de la posture d'un humain à partir d'images est un problème difficile en raison à la fois de la complexité de l'objet étudié (causée entre autres par le nombre de degrés de liberté et la forte variabilité d'apparences entre les personnes) et des ambiguïtés visuelles introduites par le système d'observation (liées aux phénomènes d'auto-occultation et à la perte d'information sur la profondeur). La diversité de ses applications potentielles - comme la réalité virtuelle, l'interface homme machine, l'analyse du geste sportif...- en fait toutefois un sujet de recherche très actif.

Cette thèse présente un système d'estimation de la configuration d'un modèle articulé du corps à partir des images acquises par un système de caméras fixes et calibrées, observant une personne évoluant dans une pièce. La méthode proposée ne suppose pas de connaissance sur les estimations précédentes dans la vidéo, et s'affranchit donc des éventuels problèmes d'initialisation ou de perte de suivi. L'objectif de ce travail est d'ouvrir la voie vers une analyse robuste et temps-réel de la posture pour l'interprétation de scènes et la vidéo surveillance.

L'analyse s'appuie tout d'abord sur une extraction de la silhouette pour chacune des caméras par une méthode de soustraction de fond. Une reconstruction en voxels de l'enveloppe visuelle du corps est ensuite obtenue grâce à un algorithme de *Shape from Silhouettes*. Cette enveloppe 3D fusionne les primitives extraites des images et les informations sur la géométrie du système d'acquisition, et représente un moyen de rendre l'estimation plus indépendante du placement des caméras.

L'estimation est ensuite basée sur une régression : l'application permettant de passer de la forme 3D reconstruite à la configuration du corps correspondante est modélisée durant une phase d'apprentissage. Les informations a priori intégrées dans le modèle appris permettent une prédiction directe de la pose à partir des données images (représentées par l'enveloppe visuelle). Le temps de calcul associé à l'estimation est réduit car le travail de modélisation est reporté sur la phase d'entrainement effectuée hors-ligne. Des bases

d'apprentissage synthétiques ont été créées grâce à des logiciels d'animation d'avatars et de rendu 3D. Pour encoder de manière concise la géométrie de l'enveloppe visuelle, un nouveau descripteur 3D a été proposé.

Différentes possibilités sur la paramétrisation du mouvement du corps, la complexité du descripteur, la méthode de régression, la configuration des caméras...ont été envisagées et testées. Toutes les méthodes proposées sont évaluées quantitativement sur des données synthétiques, qui permettent une comparaison à la vérité terrain. La robustesse du système est éprouvée qualitativement grâce à des tests sur des séquences réelles, portant sur l'analyse des mouvements de marche et de bras.

Mots-clef : posture, enveloppe visuelle, descripteur 3D, régression.

Abstract

Human pose analysis from images is a challenging task due to both the complexity of human body (as a result of the high number of degrees of freedom of the body and the variability of human appearance) and the visual ambiguities inherent to the use of image projection (lack of depth information, self occlusions...). However, the number of potential applications, such as virtual reality, human-computer interaction or athletes' gesture analysis, has intensified the interest for this topic within the computer vision community.

This thesis presents a procedure to recover the pose of an articulated body model from the images acquired by several static and calibrated cameras, observing a person moving inside a room. The proposed method does not make any assumption on the knowledge of the previous state estimates in the sequence, and so avoids the problem of initialization and lost tracks. Our goal is to provide a first step to a robust and real-time posture analysis for applications such as scene interpretation and visual surveillance.

A background subtraction algorithm is first used to compute binary silhouettes of the body from each camera. A 3D voxel reconstruction of the visual hull is then obtained through a *Shape from Silhouettes* algorithm. This 3D shape merges all the information about image data and camera calibration, and can make the estimation independent of camera setup given enough viewpoints.

We then propose a regression-based estimation : the mapping between the 3D silhouette and the configuration of the human body is learnt during an off-line training phase. The learnt model contains a priori information and allows a direct prediction of the pose from the low-level image features (encoded by the visual hull). The cost of the estimation is reduced because the main part of the modelling computation is done during the off-line training stage. Training examples have been synthesized with animation and rendering software. A new 3D shape descriptor has been proposed to encode the 3D shape in a low-dimensional vector and give it as input to the regression process.

Various possibilities have been tested concerning body parameterization, configuration of the shape descriptor, regression, camera setup...Throughout this thesis, all the proposed methods are quantitatively evaluated on synthetic data for a ground truth comparison, and qualitatively demonstrated on real sequences of walking and gesture movements.

Key-words : pose estimation, visual hull, 3D shape descriptor, regression.

Table des matières

Table des figures

Liste des tableaux

Introduction

Ce travail de thèse aborde le problème de la reconnaissance de la posture d'une personne à partir des images acquises par un système de caméras. Il s'agit d'un problème extrêmement ardu, d'abord en raison de la complexité de l'objet qui est observé. Le corps humain est un objet hautement articulé : un modèle cinématique du corps, même simplifié, contient au minimum une vingtaine de degrés de liberté. Les configurations possibles du corps sont donc extrêmement nombreuses. La variabilité d'apparence entre différentes personnes est aussi très forte. La même posture prise par différentes individus génère des observations très différentes en raison des différences de taille, de morphologie, ou de la déformation des vêtements. A cela s'ajoutent les ambiguïtés visuelles introduites par le mode d'observation. On cherche à estimer une information 3D sur la position des membres à partir de projections 2D, et même si l'on dispose de plusieurs points de vue, une partie de l'information sur la profondeur est nécessairement perdue. L'apparence du corps dans une image varie aussi beaucoup en fonction du point de vue ou de l'éclairage. Il faut enfin gérer les phénomènes d'auto-occultation : tous les membres du corps ne peuvent pas être observés à partir d'une image car ils sont souvent cachées par d'autres parties.

Les applications de ce sujet sont extrêmement nombreuses et ont justifié un intérêt grandissant dans la communauté scientifique. L'estimation précise de la posture par vision pourrait tout d'abord offrir une méthode non-invasive de reconstruction du mouvement humain pour l'animation de personnages virtuels. Les systèmes de capture de mouvement actuellement employés (voir paragraphe 1.2.4) imposent l'utilisation de matériel couteux et peu flexible, et sont surtout bien souvent contraignants pour la personne dont on mesure le mouvement, car ils l'obligent à porter un système de capteurs et une tenue spécifique. De tels systèmes ont des applications dans l'industrie du film ou du jeu. L'analyse hors-ligne des mouvements trouve également des applications en sport et en médecine : on peut imaginer l'utiliser par exemple pour le

diagnostic automatique en orthopédie, l'optimisation des performances des sportifs... D'autres applications sont l'indexation des vidéos par le contenu, la compression de l'information. La reconnaissance automatique des gestes et des postures pourrait enfin permettre un mode d'interaction naturel entre un humain et un ordinateur (e.g. reconnaissance automatique du langage des signes).

L'application principalement visée par ce travail de thèse est la vidéo surveillance. L'analyse de la posture et du comportement des personnes présentes dans une scène fait partie des problématiques liées à l'émergence des méthodes de surveillance intelligente, au même titre que la détection et le suivi de personnes, la reconnaissance des visages... L'objectif est de concevoir des systèmes de surveillance qui, au lieu de se contenter d'un enregistrement passif des séquences, nécessitant l'interprétation d'un opérateur humain, sont capables d'analyser automatiquement et en temps réel le contenu des vidéos pour attirer l'attention sur des événements anormaux (chutes, gesticulations, mouvements inhabituels...). Ces techniques trouvent bien sûr des applications dans le domaine de la sécurité (détection de comportements suspects dans des lieux publics, surveillance des piscines...), mais on peut aussi imaginer s'en servir pour des études statistiques sur le comportement des consommateurs dans un magasin, ou des individus dans une foule...

Notre étude s'est concentrée sur l'analyse des mouvements d'une personne seule dans une pièce couverte par plusieurs caméras. Elle représente une étape clé pour la mise en place de systèmes de vidéo assistance intelligente. L'idée est de proposer des systèmes autonomes et non-invasifs de surveillance de personnes en perte d'autonomie (malades ou personnes âgées), qui détectent automatiquement à partir des images acquises par les caméras des événements anormaux comme des chutes, une immobilité prolongée ou des gestes inhabituels, et déclenchent une alarme pour prévenir des secours. L'analyse que nous proposons est toutefois un peu plus poussée qu'une simple reconnaissance de postures "critiques" : à partir des images, nous avons cherché à reconstruire l'évolution des paramètres d'un modèle articulé du corps. Notre système estime à chaque instant d'une séquence vidéo la position dans l'espace des différentes parties du corps. Ce travail peut être vu comme la première étape d'un système plus global d'analyse du comportement : une phase complémentaire d'interprétation de l'évolution des paramètres de la posture au cours du temps pourrait permettre de reconnaitre des actions particulières, comme des phases de la marche, des gestes...

Choix techniques

Concernant le nombre de caméras utilisées pour l'estimation, un compromis est à trouver entre la précision du système et sa flexibilité (coût et simplicité de mise en oeuvre). Nous avons opté pour un système multi-caméras qui permet d'une part d'améliorer la robustesse de l'estimation de la pose en réduisant les ambiguïtés visuelles (voir paragraphe 1.2.1), et d'autre part laisse à plus long terme espérer une certaine robustesse vis-à-vis des occultations dans des scènes encombrées (ces problèmes n'ont pour l'instant pas été pris en compte). Nous avons tout de même cherché à limiter le nombre de caméras employées (comparativement aux systèmes de capture de mouvement) : nos expériences ont été réalisées avec des systèmes de 4 ou 5 caméras calibrées.

Nous faisons l'hypothèse que la personne dont on observe les mouvements est debout dans l'intersection des champs de vision des caméras. Notre objectif est d'estimer sa pose de manière robuste, de façon à pouvoir animer un avatar reproduisant ses mouvements au cours du temps. Les informations provenant des différents points de vue sont combinées via une reconstruction 3D de l'enveloppe visuelle : nous verrons que cette technique rend l'estimation plus indépendante du placement des caméras.

Même si les temps de calcul n'ont pas été une priorité dans nos développements, nous avons cherché à limiter la complexité calculatoire des outils employés pour permettre d'envisager à terme un fonctionnement en temps réel du système. C'est la raison pour laquelle notre choix s'est porté sur l'utilisation d'algorithmes d'apprentissage. Ce type d'approche permet de limiter le coût de l'estimation en reportant la plus grande partie des calculs dans une phase d'entrainement hors-ligne. Il évite aussi les étapes couteuses de prédiction sur la configuration d'un modèle du corps et la génération de rendus de ce modèle. De plus, les méthodes basées sur un apprentissage ne souffrent pas des problèmes d'initialisation ou de ré-initialisation auxquelles sont confrontées les méthodes de suivi : la prédiction est effectuée à partir d'une seule image et ne nécessite pas de connaissance sur les états précédents de la séquence.

Contributions

Nous avons proposé un système complet d'estimation de la pose d'un modèle articulé du corps à partir des images fournies par les caméras. Parmi les contributions de ce travail :

- un nouveau **descripteur 3D** a été proposé pour encoder la géométrie de l'enveloppe visuelle ; les performances de ce descripteur ont été comparées expérimentalement au Shape Context 3D proposé dans [106].

- **un processus d'estimation de la pose en deux temps** : l'orientation du corps dans l'espace est estimée dans un premier temps, puis les angles des différentes articulations sont évalués en supposant connue cette orientation.

- de nombreuses **expérimentations sur des données de synthèse** ont été menées pour évaluer l'importance du choix du descripteur, de la paramétrisation des mouvements, du nombre et du positionnement des caméras...

- nous avons aussi envisagé la possibilité de compléter l'estimation par le **raffinement d'un modèle du corps** se basant sur la comparaison de la position d'un modèle simplifié, initialisé avec le résultat de la régression, avec celle de l'enveloppe visuelle reconstruite. Une méthode simple a été proposée, et les résultats obtenus ouvrent des perspectives pour la mise en place d'une méthode plus complète combinant les avantages des deux types d'estimation (modèle/apprentissage).

- de bons **résultats sur séquences réelles** ont été obtenus sur des mouvements assez variés (notamment des mouvements de bras) grâce à des bases d'apprentissage appropriées. Les méthodes d'estimation de pose basées sur un apprentissage présentées dans la littérature se contentent souvent de présenter des résultats sur des séquences de marche.

Les travaux réalisés au cours de cette thèse ont donnée lieu à plusieurs publications [39, 40, 85, 84].

Structure du document

Le manuscrit est composé de cinq chapitres. Le premier chapitre est un état de l'art des méthodes d'estimation de la pose à partir d'images. Le chapitre 2 présente différents outils mis en jeu dans notre système d'estimation : il décrit d'une part les différentes représentations des mouvements du corps qui ont été envisagées, et d'autre part de quelle manière sont extraites et combinées les primitives visuelles provenant des différentes caméras. La mé-

thode de construction des bases d'apprentissage est aussi exposée. Le chapitre 3 détaille le descripteur 3D qui a été proposé dans cette thèse. Le chapitre 4 propose un état de l'art et une comparaison expérimentale de plusieurs méthodes de régression. Le chapitre 5 présente d'une part une évaluation expérimentale sur données synthétiques de plusieurs facteurs influant sur la qualité de l'estimation, et d'autre part nos résultats sur données réelles.

Chapitre 1

Etat de l'art

1.1 Introduction

L'estimation du mouvement humain à partir d'images est un problème complexe qui a fait l'objet de très nombreux travaux ces dernières années. Plusieurs états de l'art sur ce sujet ont été proposés dans la littérature [36, 73, 74, 86, 120]. Le niveau de détail et la précision à atteindre dépendent à la fois des contraintes d'acquisition du système (la qualité des images, le positionnement des caméras par rapport à la personne observée, leur nombre...) et de l'application visée. Par exemple, pour des applications de vidéo surveillance, on ne dispose souvent que d'une seule caméra, et l'estimation de la position dans l'image de blobs 2D correspondant à certaines parties du corps (jambes, torse...) représente déjà une information satisfaisante. A l'inverse, si l'objectif est la capture de mouvement pour de l'animation, une estimation très précise de la configuration 3D des différentes articulations du corps sera exigée. Pour l'interface homme-machine, il faut généralement parvenir à estimer de manière assez précise les positions de certains membres, mais l'évaluation porte parfois seulement sur une partie du corps (le haut du corps, les bras), et s'applique à une personne proche de la caméra et lui faisant face. L'interprétation de scènes requiert quant à elle une analyse encore plus haut niveau de la pose et de son évolution au cours du temps.

Dans le paragraphe préliminaire 1.2, nous aborderons différentes questions générales liées à ce problème. La deuxième partie 1.3 présente les principales approches proposées dans la littérature pour estimer la pose à partir d'images. Nous montrerons enfin dans le paragraphe 1.4 comment les travaux présentés dans cette thèse se positionnent par rapport aux approches existantes.

1.2 Généralités

1.2.1 Monoculaire vs. multi-cameras

On peut faire une première distinction entre les systèmes d'estimation mono-caméra et les systèmes multi-caméras. Les ambiguïtés visuelles ne sont pas les mêmes suivant le nombre de caméras utilisées, et la méthode d'estimation doit tenir compte du degré d'ambiguïté du système. Les approches monoculaires ont un champ d'application extrêmement large puisqu'elles sont applicables dans tous les cas où l'on ne dispose que d'une seule caméra, et ne nécessitent en général aucun calibrage. Elles permettent par exemple de traiter a posteriori des séquences provenant de scènes de films, de vidéos recueillies sur internet, ou d'interpréter des images issues de caméras de surveillance (dans un lieu couvert par une seule caméra)...
Estimer une pose 3D à partir d'une seule caméra est un problème particulièrement complexe en raison des ambiguïtés liées à la projection $3D \rightarrow 2D$. Les ambiguïtés sont principalement de deux types :

- l'absence d'information sur la profondeur ne permet pas de savoir dans quel sens sont orientées certaines parties du corps par rapport au plan de l'image. Par exemple, pour une personne marchant face à la caméra, il est difficile de savoir si ses jambes sont orientées vers l'avant ou vers l'arrière (voir figure 1.1(a)).
- les problèmes d'auto-occultations (se produisant lorsqu'une partie du corps en cache un autre dans l'image) ne peuvent pas être gérés correctement.

(a) (b) (c) (d) (e)

FIG. 1.1 – Exemples d'ambiguïtés visuelles.
a : ambiguïté sur la position des jambes. **b et c :** perte d'information sur la profondeur. **d et e :** exemples d'auto-occultations.

L'estimation de la pose du corps à partir d'une seule image est un problème sous-déterminé : dans [105], Sminchisescu et Triggs estiment ainsi

qu'environ un tiers des degrés de liberté d'un modèle humain est inobservable à partir d'une séquence monoculaire. Il est à la fois difficile d'identifier dans l'image les différentes parties du corps (déterminer quelle partie du corps occulte quelle autre, distinguer le bras gauche du bras droit etc.), et même lorsque les positions en 2D des points du corps sont connues, les ambiguïtés sur la profondeur font qu'il existe encore de nombreuses configurations 3D susceptibles de leur correspondre (*kinematic flipping ambiguities*).

L'estimation monoculaire impose de recourir à des méthodes de résolution particulières en raison de toutes ces ambiguïtés. En particulier, la méthode doit tenir compte du fait qu'on ne peut généralement pas donner une estimation unique de la pose à partir d'une seule image, et qu'il faut souvent formuler plusieurs hypothèses. Différentes techniques ont donc été proposées pour gérer ce problème : dans [5, 104], une mixture de régresseurs est employée pour estimer plusieurs solutions possibles. Dans [7], les auteurs proposent aussi d'utiliser la cohérence temporelle pour lever des ambiguïtés qui apparaissent lorsque l'estimation est faite à partir d'une seule image. Si l'estimation est basée sur un suivi temporel, des algorithmes de suivi tels que le filtre à particules permettent de propager dans le temps plusieurs hypothèses sur la pose (voir paragraphe 1.3.6). Une autre possibilité est de se limiter à l'estimation d'une pose 2D, soit en estimant la configuration d'un modèle 2D (voir 1.3.3), soit en identifiant simplement la position des parties des parties du corps dans l'image (voir 1.3.5).

Certaines limitations de ces approches peuvent être levées par l'utilisation de systèmes multi-caméras. L'utilisation de tels systèmes devient aujourd'hui envisageable d'une part grâce à la diminution des coûts du matériel et d'autre part grâce l'augmentation de la puissance de calcul des ordinateurs. Le nombre de caméras utilisées peut varier énormément suivant les contraintes matérielles et l'application visée. On trouve par exemple des systèmes stéréoscopiques de reconnaissance de gestes, dans lesquels la deuxième caméra permet simplement d'avoir une information supplémentaire sur la profondeur, ou à l'autre extrême, des systèmes tels que la Plateforme GrImage de l'INRIA Rhone Alpes (8 caméras, 16 projecteurs et un cluster de 11 PCs), dont l'objectif est de reconstruire en temps réel un modèle 3D de la scène observée.

Avec un système multi-caméras, les ambiguïtés sur la profondeur et les problèmes d'occultations sont réduits, et l'estimation peut être à la fois plus précise et plus robuste. Les méthodes d'estimation multi-caméras varient suivant le nombre de caméras, la façon dont elles sont positionnées, leur calibrage éventuel... Plusieurs manières de combiner les données provenant des

différentes caméras ont été proposées (la fusion des informations peut être réalisée plus ou moins tôt dans l'estimation) :

- une première solution est d'estimer la pose indépendamment pour chacune des caméras, puis de fusionner les hypothèses pour retrouver la pose la plus cohérente. Dans [95], les auteurs proposent ainsi de formuler pour chaque caméra une hypothèse sur la pose grâce l'estimation monoculaire proposée dans [94], et de combiner ces hypothèses pour retrouver à la fois la pose la plus cohérente avec l'ensemble des images et la position des caméras.

- une deuxième possibilité, proposée dans [78], est de rechercher, pour une posture donnée, la caméra pour laquelle les ambiguïtés visuelles sont les plus faibles, puis d'effectuer l'estimation en se basant uniquement sur cette caméra. Dans ces travaux, les auteurs définissent un critère permettant de mesurer les ambiguïtés d'une image en fonction de la pose, et sélectionnent la vue la moins ambigüe pour effectuer l'estimation.

- dans le cas d'une méthode basée sur l'apprentissage d'une application $image \rightarrow pose$, la fusion des images peut être réalisée en concaténant les descripteurs des différentes vues pour former un vecteur caractéristique de plus grande dimension (voir paragraphe 3.2.2).

- lorsque le calibrage des caméras est connu, une possibilité est d'utiliser les techniques de *Shape from Silhouette* pour calculer une reconstruction 3D, et d'estimer la pose à partir de cette forme 3D. Nous reviendrons sur ces techniques dans les chapitres suivants.

1.2.2 Estimation de la pose vs. reconnaissance de postures

Dans la suite de cet état de l'art, nous nous intéresserons principalement à des techniques visant à estimer de façon très précise la pose d'une personne, c'est-à-dire à évaluer la localisation (dans l'espace ou dans l'image) des différentes parties de son corps. Un problème voisin est celui de la reconnaissance de postures, dans lequel on cherche à déterminer par exemple si la personne observée est debout, assise, couchée... Dans le premier cas, la sortie du système est une donnée multivariée continue, comme la configuration an-

gulaire des articulations d'un modèle du corps, tandis que dans l'autre cas, l'estimation se contente d'attribuer un label (l'appartenance à une classe) correspondant aux données observées. Ces méthodes de classification de postures peuvent paraître moins ambitieuses, puisque l'information à laquelle elles cherchent à accéder est beaucoup moins riche. Mais elles ont aussi des chances d'être plus robustes et moins sensibles aux conditions d'acquisition et aux erreurs de segmentation, et représentent donc souvent une alternative plus réaliste pour certaines applications comme la vidéosurveillance. La suite de ce paragraphe est consacrée à quelques unes des méthodes de reconnaissance de poses qui ont été proposées.

Dans ces travaux, un ensemble de postures de référence est défini, et l'estimation reste ensuite limitée à ces postures, c'est-à-dire qu'elle ne peut produire en sortie que l'appartenance à l'une des ces classes (ou à aucune des classes) ; la gestion des cas ambigus représente donc souvent un point délicat. La difficulté principale de ce type d'évaluation est que, pour une posture donnée, l'apparence du corps dans l'image peut considérablement varier suivant le point de vue. Il faut aussi prendre en compte la variabilité d'apparence entre les personnes et dans la façon de réaliser une posture donnée.

Les méthodes de reconnaissance ayant comme objectif des applications de vidéosurveillance reconnaissent classiquement 4 grandes classes de postures : assis, debout, couché, et penché. Dans [48], les auteurs définissent ainsi 4 classes de postures et 3 principaux points de vue : vue de face, du côté gauche ou du côté droit. L'appartenance à l'une des 4 classes est estimée à partir de la silhouette, qui est caractérisée par les projections normalisées de ses pixels sur les 2 axes (vertical et horizontal) de l'image. Des modèles d'histogrammes moyens sont construits pour chaque classe de postures, et les histogrammes de la silhouette sont comparés à ces histogrammes pour déterminer à quelle classe elle appartient. Le point de vue est ensuite évalué selon la même méthode. D'une manière similaire, les auteurs de [81] classifient les silhouettes en se basant sur des cartes probabilistes, construites sur des données d'apprentissage, représentant les projections sur les 2 axes de l'image des pixels de la silhouette. Dans [28], cette méthode est intégrée dans un module de suivi qui exploite la cohérence temporelle des poses estimées au cours du temps grâce à un graphe de transition entre états. Dans [15] et [16], Boulay et al. utilisent un modèle 3D pour gérer la dépendance des silhouettes 2D par rapport au point de vue. Le modèle 3D est positionné dans l'espace au même endroit que la personne observée, animé avec l'ensemble des postures prédéfinies, et les silhouettes du modèle sont rendues avec le même point de vue que la caméra réelle. Toutes les orientations possibles (avec un certain pas)

sont testées. Les silhouettes ainsi générées sont comparées avec la silhouette extraite de l'image en utilisant différentes primitives 2D : les moments de Hu, une squelettisation et des projections verticale et horizontale des pixels de la silhouette. Dans [84], un système de reconnaissance multi-caméras permet d'identifier 3 types de postures : debout, assis par terre, et couché. Un filtre à particules, basé sur les silhouettes extraites pour chaque caméra, est utilisé pour suivre la personne en 3D et ajuster un modèle simplifié du corps (un rectangle 3D). Les dimensions du modèle (le ratio hauteur/largeur du rectangle) permettent ensuite d'identifier la posture.

On trouve aussi des méthodes de reconnaissance portant sur des mouvements plus variés, par exemple dans [26], où les autres cherchent à reconnaitre des classes de gestes, pour des applications d'interface homme-machine. Dans ces travaux, un ensemble de 12 postures est défini selon la position des bras, et la reconnaissance de ces poses est basée sur une reconstruction en 3D de l'enveloppe visuelle et un classifieur SVM. Dans [45], un classifieur RVM est utilisé pour reconnaitre des postures de danse.

1.2.3 Reconnaissance de la posture dans des foules

Dans un groupe de personnes ou dans une foule, l'analyse est rendue encore plus complexe par le fait que les différents individus s'occultent les uns les autres dans les images. Elle est le plus souvent basée sur une extraction des régions d'intérêt de l'image par soustraction de fond. Si les personnes sont isolées dans la scène, chaque blob obtenu a des chances de correspondre approximativement à un individu, mais lorsque la densité est plus importante, plusieurs objets d'intérêt sont souvent regroupés en une seule masse. Une étape de détection doit donc être utilisée pour localiser les composantes de ces blobs correspondant aux différentes personnes ; celle-ci s'appuie généralement sur une détection de la tête, ou de l'ensemble tête-épaules.

La plupart des études sur les groupes de personnes se sont concentrées sur la détection et la segmentation les différents individus dans une foule, ou bien sur des techniques de suivi multi-cibles capables de gérer les occultations. Certains travaux s'intéressent à l'analyse des postures des individus, soit en cherchant à segmenter grossièrement certaines parties du corps, soit en faisant correspondre aux données image un modèle simplifié du corps. Une information, même simplifiée, sur la posture des personnes présentes dans une foule, représente une donnée importante pour la surveillance et la détection d'événements anormaux dans des lieux publics. Les systèmes W^4 et W^4S (système stéréo) présentés dans [47] et [49] traitent le cas de groupes

peu denses, et dans les lesquels les différentes personnes sont distribuées horizontalement dans la scène. Ces systèmes fonctionnent en temps réel à partir d'images en niveau de gris (visibles ou infrarouges). A partir des blobs extraits par soustraction de fond, les différentes personnes sont segmentées grâce à des histogrammes de projections verticales (similaires à [48]). Des modèles d'apparence permettent de suivre les différents individus, même après des occultations. L'analyse des silhouettes permet également de localiser différentes parties du corps (tête, torse, mains et pieds), et de déterminer si les personnes portent des objets.

Dans [124], Zhao et al. proposent de segmenter les personnes debout en utilisant un modèle 3D simplifié du corps composé d'ellipsoïdes correspondant à la tête, au torse et aux deux jambes. Le modèle peut prendre plusieurs formes en fonction de la posture (statique, jambes gauche devant ou jambe droite devant). Le problème est posé dans un cadre bayésien, et résolu en calculant le maximum d'une densité a posteriori prenant en compte la vraisemblance par rapport aux données image et un a priori sur le nombre et la taille des objets présents dans la scène. La solution optimale, estimant à la fois le nombre d'individus et la configuration des modèles, est recherchée par un échantillonnage MCMC, guidé par des observations image (détection des têtes et analyse des blobs extraits). Les auteurs proposent aussi dans [125] une analyse en deux étapes : le déplacement des différentes personnes dans la scène est suivi en modélisant le corps par un ellipsoïde 3D, puis un modèle articulé du corps est ajusté pour reconnaitre des phases de la marche et déterminer si la personne est statique, en train de marcher ou de courir.

Dans [85], nous avons proposé une méthode pour segmenter les individus dans un groupe dense et détecter automatiquement des chutes à partir des images acquises par une caméra infrarouge statique. Elle consiste à segmenter dans un premier temps les personnes debout, identifiées par une détection de la tête, en ajustant un modèle simplifié du corps à la carte extraite par soustraction de fond. Les parties de l'image correspondant aux personnes debout sont éliminées de la carte, puis une analyse des blobs restants permet de distinguer les personnes à terre. Le fonctionnement de ce système est détaillé dans l'annexe B.

1.2.4 Systèmes industriels de capture de mouvements

Dans ce paragraphe, nous détaillons le fonctionnement de quelques uns des systèmes commerciaux de capture de mouvement les plus utilisés. Comme nous le verrons, ces systèmes sont souvent assez invasifs, car ils obligent l'uti-

lisateur à porter une tenue spécifique, comprenant parfois des équipements électroniques lourds et encombrants. De tels systèmes sont utilisés dans l'industrie du film ou du jeu pour l'animation de personnages de synthèse, ou bien encore en médecine clinique (analyse de la démarche d'une personne) ou en bio-mécanique.

La présence de marqueurs sur l'acteur fait la force de ces systèmes en leur permettant d'atteindre une très bonne précision sur la position des membres du corps, mais constitue aussi leur faiblesse car elle rend leur utilisation peu adaptée pour des applications comme la vidéosurveillance. Au contraire, les méthodes d'estimation basées sur la vision n'exploitent que les données contenues dans les images (contours, couleur, texture...) en n'imposant aucune contrainte particulière à la personne observée.

Systèmes de capture mécanique

Ces systèmes fonctionnent grâce à un exosquelette placé directement sur les membres du corps dont on souhaite analyser le mouvement. L'exosquelette est constitué de segments rigides reliés par des articulations munies de codeurs angulaires. Ces systèmes permettent de connaitre la configuration relative des membres du corps avec une bonne précision et en temps réel, mais ne fournissent pas d'information sur la position du corps dans l'espace. L'avantage de ces systèmes est qu'ils sont généralement peu coûteux (de l'ordre de 25 000 €), mais ils sont encombrants et l'exosquelette limite la liberté de mouvement de l'acteur.

Il existe également des systèmes de capture basés sur des centrales inertielles, qui permettent de mesurer les mouvements de façon précise et en temps réel, et avec une certaine souplesse d'utilisation : ils ne nécessitent pas d'effectuer les mesures dans un studio spécifique, l'espace de capture est assez étendu, et l'information est transmise des capteurs vers l'ordinateur par des connexions sans fils. Le prix de ce type de système va de 25 000 à 80 000 €.

Systèmes de capture magnétique

Ces systèmes reposent sur l'utilisation d'une source émettant un champ électromagnétique et de capteurs fixés sur les membres du corps. Ils sont moins encombrants pour l'acteur que les systèmes mécaniques, mais l'environnement dans lequel se fait l'acquisition est relativement contraint, car le champ magnétique risque d'être perturbé par la présence d'objets métalliques qui peuvent fausser les résultats, et le volume de capture (lié à la portée de la source du champ) est beaucoup plus limité qu'avec les méthodes optiques.

Systèmes de capture optique avec marqueurs

On distingue principalement deux types d'équipement : les systèmes à marqueurs actifs et à marqueurs passifs. Dans le premier cas, l'acteur est équipé de marqueurs actifs (micro-LED) émettant un signal infrarouge capté par des cellules photosensibles.

Les systèmes optiques à marqueurs passifs sont les systèmes les plus utilisés et sans doute les moins contraignants pour la personne dont on capture les mouvements. L'un des plus connus est le système développé par la société VICON. Les caméras sont équipées de petites diodes lumineuses émettant un signal rouge et/ou infrarouge (voir figure 1.2(c)). Des marqueurs passifs portés par l'acteur (généralement des petites sphères dont la surface est ultra-réfléchissante, voir figure 1.2(a)) réfléchissent le rayonnement vers les caméras. L'utilisation de caméras infrarouges permet d'obtenir un très fort contraste entre les marqueurs et l'image de fond, de sorte que les marqueurs apparaissent dans les images comme des points très lumineux. A partir des points détectés, les positions des marqueurs sont reconstruites dans l'espace 3D. L'acteur porte généralement une tenue proche du corps et de couleur sombre car il faut s'assurer que la scène contienne le moins possible d'objets clairs susceptibles de réfléchir la lumière.

On retrouve dans la mise en oeuvre de ces techniques de nombreuses problématiques rencontrées en vision : les cibles doivent être détectées de manière très précise (détection sous-pixellique par ajustement d'une gaussienne), les caméras doivent être parfaitement calibrées et synchronisées, les marqueurs mis en correspondance dans les différentes images, puis leurs positions reconstruites en 3D. Il faut également gérer les problèmes d'occultations des marqueurs dans les images par des parties du corps. Au moins 6 caméras sont nécessaires pour couvrir l'ensemble des marqueurs, mais en pratique, pour obtenir un suivi plus robuste, un studio de capture de mouvement comprend généralement entre 7 et 24 caméras (voir figure 1.2(b)). Le prix de ce type de système varie en fonction du nombre de caméras, et peut atteindre 400 000 € pour un système de 12 caméras.

1.3 Méthodes d'estimation

Une distinction peut être faite entre les méthodes d'estimation de pose image par image, et les méthodes de suivi. Dans le premier cas, le système doit être capable d'estimer la pose d'un humain à partir d'une seule image, ou dans le cas d'un système multi-caméras, d'un ensemble d'images acquises

(a) (b) (c)

FIG. 1.2 – Systèmes optiques de capture de mouvements.
a : acteur portant les marqueurs réfléchissants utilisés pour la capture de mouvements (source : http ://en.wikipedia.org/wiki/Motion_capture). **b :** studio de capture de mouvement (source : www.creativereview.co.uk). **c :** caméra commercialisée par la société VICON (www.vicon.com).

simultanément. La méthode ne nécessite aucune connaissance sur les états estimés dans les vues précédentes. Les méthodes de suivi cherchent quant à elles à estimer la pose sur un ensemble d'images consécutives d'une séquence vidéo. Elles supposent généralement qu'on dispose d'une initialisation de la pose sur la première vue de la séquence, et se concentrent sur l'évolution de cette pose au cours du temps. Elles supposent également que l'écart de temps entre deux images successives est suffisamment faible pour se permettre de rechercher la pose de la vue courante dans le voisinage de l'estimation précédente. Un système de suivi complet et robuste doit cependant être capable d'initialiser la pose, et de se réinitialiser lorsque le suivi est perdu.

Il existe principalement deux grandes familles d'approches pour estimer la posture à partir d'images : les méthodes basées sur un modèle et les méthodes basées sur des exemples. Ces méthodes seront présentées respectivement dans les paragraphes 1.3.3 et 1.3.4. Un troisième type de méthode, un peu à part, consiste à détecter dans un premier temps des hypothèses sur les positions des différentes parties du corps dans les images (les bras, la tête, le torse...), puis à assembler ces parties pour remonter à une information sur la pose, en s'appuyant sur un critère de cohérence sur la proximité des membres du corps dans l'image. Ces méthodes sont présentées dans le paragraphe 1.3.5. Le paragraphe 1.3.6 aborde les différentes techniques d'utilisation des aspects temporels dans une séquence vidéo.

1.3.1 Primitives visuelles utilisées

Toutes les méthodes d'estimation de la pose, qu'elles se basent sur l'optimisation d'un modèle ou sur un apprentissage, ont en commun le fait de devoir tirer profit au mieux des informations bas niveau présentes dans l'image pour déduire une estimation haut niveau sur la configuration du corps humain qui a généré ces observations. Elles peuvent pour cela s'appuyer sur différents types de primitives extraites de l'image.

Les primitives sur lesquelles s'appuie l'estimation doivent être à la fois simples et rapides à extraire, robustes aux conditions d'acquisition (par exemple aux variations d'éclairage), et en même temps contenir suffisamment d'information pour permettre d'analyser la posture de la personne observée. Selon les travaux, les données extraites de l'image peuvent être des informations bas-niveau, comme les contours, la silhouette..., ou des données plus élaborées, contenant déjà une certaine quantité d'information sur la position du corps dans l'image (détection des mains, du visage, de la peau...).

Parmi les primitives les plus couramment utilisées, on trouve :

- **la silhouette et les contours externes de la silhouette :** la silhouette et les contours externes (les contours de la silhouette) peuvent généralement être extraits de manière robuste à partir du moment où la caméra est statique et le fond stable. Comme souligné dans [7], la silhouette contient déjà une grande quantité d'information intéressante pour estimer la pose, tout en étant invariante à la plupart des caractéristiques inutiles comme la couleur des vêtements ou leur texture. Son principal défaut est qu'elle rend invisible certains degrés de liberté du corps, et que des poses assez différentes peuvent avoir des silhouettes similaires. Si par exemple les bras sont le long du corps, il est très difficile d'analyser leur mouvement simplement à partir des contours de la silhouette. Elle introduit aussi des ambiguïtés de symétrie. Par exemple, si une personne se déplace en marchent parallèlement au plan de la caméra, la silhouette ne donne aucun moyen d'identifier la jambe gauche ou la jambe droite. Avec une seule image, elle ne permet pas non plus de savoir si une personne est de face ou de dos. Toutes ambiguïtés sont évidemment réduites si plusieurs silhouettes extraites de différents points de vue sont utilisées. Les performances de l'estimation peuvent aussi être limitées par la présence sur la silhouette de bruits ou d'artéfacts (trous, ombres...).
  Il est également possible de segmenter la silhouette d'une personne en

mouvement grâce à des méthodes basées sur les contours actifs [30]. Dans le cas où les caméras sont calibrées, une reconstruction 3D peut être construite à partir des silhouettes extraites des différentes images : l'enveloppe visuelle peut donc être vue comme une utilisation particulière des silhouettes. Plusieurs approches s'appuyant sur une reconstruction 3D de l'enveloppe ont été proposées [14, 71, 72, 106].

- **les contours de l'image :** les contours d'une image peuvent être extraits de manière robuste à un faible coût. Les contours internes permettent d'accéder à des informations sur des parties du corps situées à l'intérieur de la silhouette, par exemple en cas d'auto-occultation. Pour ne considérer que les contours utiles de l'image, l'analyse ne porte souvent que sur les contours situés à l'intérieur de la silhouette [98] ou dans la boite englobante de la silhouette [87]. Dans les méthodes basées sur des exemples ou un apprentissage, ces contours sont souvent décrits par des histogrammes d'orientation de gradient [98, 87, 6], ou par le Shape Context [75]. L'inconvénient principal est qu'il est difficile de différencier les contours utiles (ceux qui délimitent une partie du corps) des autres contours (par exemple un pli ou un motif sur les vêtements), qui constituent alors un bruit dans le descripteur.

- **la couleur ou la texture :** leur utilisation se base sur le fait que la couleur ou la texture des membres du corps restent inchangées le long d'une séquence, même lorsque leur pose varie. Elle nécessite en général de connaitre a priori la couleur ou la texture de l'objet d'intérêt, par exemple en réalisant un apprentissage hors-ligne. Dans [65], une phase préliminaire d'apprentissage permet de modéliser la texture des vêtements (volontairement très texturés) pour recaler un modèle de la jambe sur les images. Dans [90], l'apparence des membres du corps est apprise à partir de la vidéo, puis utilisée pour améliorer la détection dans les images suivantes. Dans [76], la reconnaissance des parties du corps est guidée en faisant l'hypothèse que des parties symétriques (par exemple le bras gauche et le bras droit) doivent avoir la même couleur dans l'image. Ce type de primitives est toutefois assez sensible, et les performances de la méthode risquent d'être dégradées par des variations d'éclairage, la déformation des vêtements, ou le manque de texture.

- **le mouvement :** le flot optique mesuré dans l'image peut fournir une

indication intéressante sur le mouvement du modèle qui l'a généré. Dans [55], le mouvement de patchs modélisant en 2D les différentes parties du corps est estimé en appliquant des contraintes sur le flot optique des pixels de l'image situés dans la zone délimitée par un patch. Dans [22], Bregler et Malik introduisent des outils (*Twist Motion Model*) permettant de simplifier la relation entre le mouvement d'un modèle 3D et le mouvement mesuré dans l'image.

– **détection de la peau ou de parties du corps :** nous verrons en 1.3.5 que certaines méthodes sont entièrement basées sur une détection préalable des différentes parties du corps dans l'image. Dans d'autres approches, la détection de parties du corps peut être utilisée en complément d'autres observations pour guider l'estimation de la pose. Certains travaux tirent notamment profit du fait que la peau possède une couleur bien spécifique et peut se distinguer facilement des autres éléments de l'image. Dans [17, 62], des blobs correspondant au visage ou aux mains sont ainsi extraits de l'image grâce à une analyse de la couleur. Dans [62], le visage également est détecté par AdaBoost.

– **une combinaison de plusieurs primitives** : pour tirer un maximum de bénéfice des informations contenues dans l'image, il est bien sûr possible de combiner plusieurs de ces primitives. Dans [62], les auteurs combinent par exemple une détection du visage et de la peau et une analyse des contours. Dans [105], la fonction de vraisemblance est construite à partir d'informations sur l'intensité, les contours et le flot optique.

1.3.2 Critères d'évaluation des méthodes

Comparer les différentes méthodes d'estimation de la pose n'est pas évident. Pour évaluer quantitativement la précision d'une méthode, il faut disposer d'une base de données avec d'un coté de la vérité terrain sur la configuration 3D du corps et de l'autre les images associées. De telles données sont relativement faciles à générer avec des logiciels de synthèse comme Poser, Character Studio, Maya... En revanche, la construction d'une base avec des images réelles est plus délicate, car elle nécessite de faire les acquisitions à la fois avec les caméras et un système de capture de mouvement (voir 1.2.4). Il faut aussi construire un critère d'erreur suffisamment général pour pouvoir être utilisé quelle que soit la méthode d'estimation. Des bases de données

ont ainsi été proposées pour tenter de comparer de manière objective différents systèmes d'estimation : on peut citer par exemple la base MoBo du CMU (Carnegie Mellon University) [43] ou plus récemment la base HumanEva [101]. Cette dernière met à disposition un ensemble de données vidéos acquises simultanément avec 7 caméras calibrées (3 caméras couleur et 4 noir et blanc) et le système VICON, sur une gamme de mouvements variés (marche, gestes...), réalisés par différentes personnes.

Il reste néanmoins très difficile de comparer des méthodes qui n'ont pas forcément le même objectif et ne produisent pas le même type de résultats en sortie : certaines méthodes se contentent d'estimer la position 2D des parties du corps dans l'image, alors que d'autres recherchent une configuration 3D du squelette. Et même pour les méthodes qui estiment une pose 3D , on peut définir autant de mesures de l'erreur que de paramétrisations possibles des mouvements du corps (positions de points du corps, angles...), et l'erreur dépend aussi du niveau de détail dans la modélisation du squelette (nombre d'articulations...).

Il existe en outre bien d'autres critères que la précision pour apprécier la qualité d'une méthode d'estimation : d'autres éléments importants sont la rapidité (système temps réel), la robustesse au bruit, aux occultations, le nombre de caméras nécessaires, la variété des poses reconnues, la flexibilité (par exemple s'il est nécessaire de réapprendre un modèle du corps pour chaque personne observée), la nécessité ou non de fournir une initialisation, la généralité de la méthode par rapport à la position des caméras...

1.3.3 Méthodes basées sur un modèle

Ces méthodes reposent sur l'utilisation d'un modèle explicite du corps humain, défini a priori, pour représenter la personne observée dans les images. La pose du corps est estimée par une approche de type "analyse-synthèse" : des prédictions sont effectuées sur la configuration du modèle, et sont ensuite mises à jour grâce aux informations contenues dans l'image. Plusieurs étapes de modélisation sont nécessaires dans la mise en oeuvre de ces techniques : il faut d'abord définir le modèle du corps humain (1.3.3), et la façon dont il se projette dans l'image pour prédire une apparence. Une fonction de vraisemblance, s'appuyant sur différentes primitives extraites de l'image doit également être construite pour mesurer la concordance entre les données visuelles et l'apparence générée du modèle dans l'image (1.3.3). La configuration optimale du modèle, c'est-à-dire qui maximise cette fonction de vraisemblance, est ensuite estimée (1.3.3).

Les méthodes d'estimation basées sur un modèle sont souvent assez précises mais aussi coûteuses en temps de calcul : elles requièrent l'optimisation d'une fonction de coût très complexe, et le modèle du corps humain doit être rendu dans les images et la fonction de coût évaluée pour chaque hypothèse sur la pose du corps.

Les différents modèles du corps humain

Pour définir un modèle du corps humain, il faut préciser d'une part la structure cinématique du squelette et d'autre part la forme et l'apparence des différentes parties du corps (la chair qui entoure le squelette). Le corps est généralement modélisé par un arbre cinématique, constitué de segments reliés entre eux par des articulations. Chaque articulation a un certain nombre de degrés de liberté (DDL), et l'ensemble des DDL forme la représentation d'une pose du corps et définit l'ensemble des paramètres à estimer lors de l'optimisation de la fonction de coût. Par ailleurs, comme la modélisation du corps est explicite, il est possible d'imposer a priori certaines contraintes au modèle, par exemple en limitant l'amplitude des angles de certaines articulations.

De nombreux modèles, 2D ou 3D, ont été proposés dans la littérature. Le nombre de DDL, ainsi que la complexité de l'apparence des parties du corps humain varient énormément suivant les travaux. D'une façon générale, plus le modèle est complexe, plus le résultat a de chance d'être précis, mais plus l'estimation risque d'être coûteuse en temps de calcul. Dans certaines approches utilisant un modèle assez détaillé, une phase préliminaire de réglage du modèle est nécessaire, pour adapter ses paramètres à l'acteur dont on estime les mouvements. Utiliser un modèle 3D présente l'avantage de rendre l'approche plus indépendante de la position des caméras, car un modèle 2D doit être adapté à l'angle de vue : le modèle 2D d'une personne vue de dessus est nécessairement différent de celui qui est utilisé si la personne est observée de face. Cependant, l'information recherchée est plus faible avec un modèle 2D car ces méthodes permettent simplement de localiser les différentes parties du corps dans l'image sans inférer de positions 3D.

Modèles 2D. L'un des modèles les plus simples a été introduit par [55] sous le nom de *Cardboard People* ("personnes en carton") : il s'agit d'une structure chainée dans laquelle chaque partie du corps est représentée par un morceau de plan, connecté à un prédécesseur et un successeur (figure 1.3(a)). Chaque segment est paramétré par un ensemble de 8 DDL, représentant

des paramètres de rotation et de déformation affine permettant d'adapter le modèle à l'apparence des segments dans l'image. Ce type de modèle est repris dans [49] et [51]. Un modèle similaire, appelé *Scaled Prismatic Model*, est proposé dans [77] et [24] (figure 1.3(b)). Ce modèle décrit l'apparence dans le plan de l'image des segments de la chaîne cinématique 3D. Chaque articulation possède un degré de liberté θ définissant l'angle de rotation du segment autour d'un axe perpendiculaire au plan de l'image, et un autre degré de liberté d représentant la distance entre deux articulations dans la chaîne, et qui modélise l'allongement ou le raccourcissement apparent d'un membre lorsque l'articulation effectue une rotation perpendiculaire au plan de l'image. Ce modèle contient au total 19 DDL. Chaque segment est couplé avec un modèle d'apparence du membre du corps qu'il représente. Ce type de modèle permet selon les auteurs de contourner certaines singularités inhérentes à la mise en correspondance d'un modèle 3D avec des images [77].

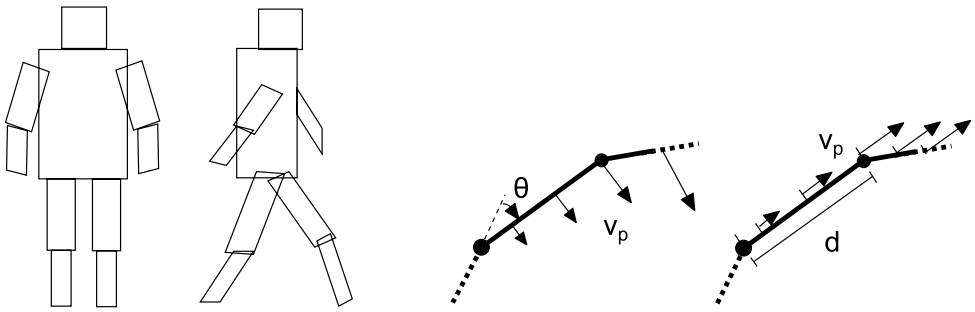


FIG. 1.3 – Exemples de modèles 2D.
a : Modèle *Cardboard people* de [55]. **b :** *Scaled Prismatic Model* de [77].

Modèles 3D. Les modèles 3D représentent le plus souvent les parties du corps par des segments rigides, avec au maximum 3 degrés de liberté de rotation par articulation. Le nombre total de DDL du squelette est très variable : cela peut aller d'une quinzaine de DDL dans le cas où seule la configuration du haut du corps est estimée [37], à plus de 50 DDL [10]. Concernant la modélisation de l'apparence des membres du corps, la gamme des modèles proposés dans la littérature est extrêmement variée. On peut citer par exemple (en allant des modèles les plus simples vers les plus complexes) :

- des modèles simplement composés de bâtons [72]
- des modèles composés de différentes primitives géométriques simples : cylindres ou cylindres généralisés, cônes [30], cônes elliptiques [119], sphères, ellipsoïdes...

- des modèles un peu plus élaborés, par exemple avec des superquadriques [38, 105]
- des modèles qui s'adaptent aux dimensions de l'acteur : avec des modèles déformables [56], ou un maillage triangulaire pour modéliser la surface [9, 14],
- des modèles contenant plusieurs niveaux de description, comme dans [96] (voir figure 1.4(d)), où le modèle est composé d'une structure cinématique modélisant le squelette, d'un ensemble d'ellipsoïdes (des "métaballes") pour modéliser les muscles et la chair, et d'un modèle polygonal de la surface pour représenter la peau.

Quelques exemples de ces modèles sont donnés sur la figure 1.4.

Fonction de vraisemblance

La fonction de vraisemblance est une mesure de similarité, qui évalue à quel point l'image et le modèle synthétisé se correspondent. Cette fonction est évaluée pour toutes les poses prédites du modèle jusqu'à ce que la meilleure configuration du modèle (celle qui explique le mieux les données image) soit trouvée. Elle dépend des primitives visuelles (paragraphe 1.3.1) sur lesquelles s'appuie la méthode. Si par exemple l'étude de la pose est basée sur les contours, la correspondance peut être faite en évaluant les forces qu'il faudrait appliquer au modèle pour ajuster ses contours à ceux de l'image [30] ou bien en calculant la distance de Chamfer [38]. Si la méthode est basée sur la silhouette, une similarité peut être définie en mesurant l'aire chevauchement entre les silhouettes. Dans [65], Lerasle et al. comparent la texture d'un modèle de la jambe (appris hors-ligne) à celle des images en calculant un coefficient de corrélation.

Estimation

Une fois définis le modèle du corps humain et la fonction de vraisemblance, la méthode d'estimation de la pose se ramène à la maximisation de cette fonction de vraisemblance. L'optimisation doit permettre de trouver la configuration du modèle qui explique le mieux les données image. Pour cela, les outils classiques d'optimisation non-linéaire peuvent être employés. Les auteurs de [22] utilisent une méthode de type Newton-Raphson pour minimiser la fonction de coût. Dans [55] et [14], une optimisation par descente de gradient est utilisée. Dans [72], la pose du modèle est ajustée par un algorithme EM utilisant la méthode de Levenberg-Marquardt. Dans [30], Delamarre et Faugeras cherchent à minimiser des forces entre les contours de la

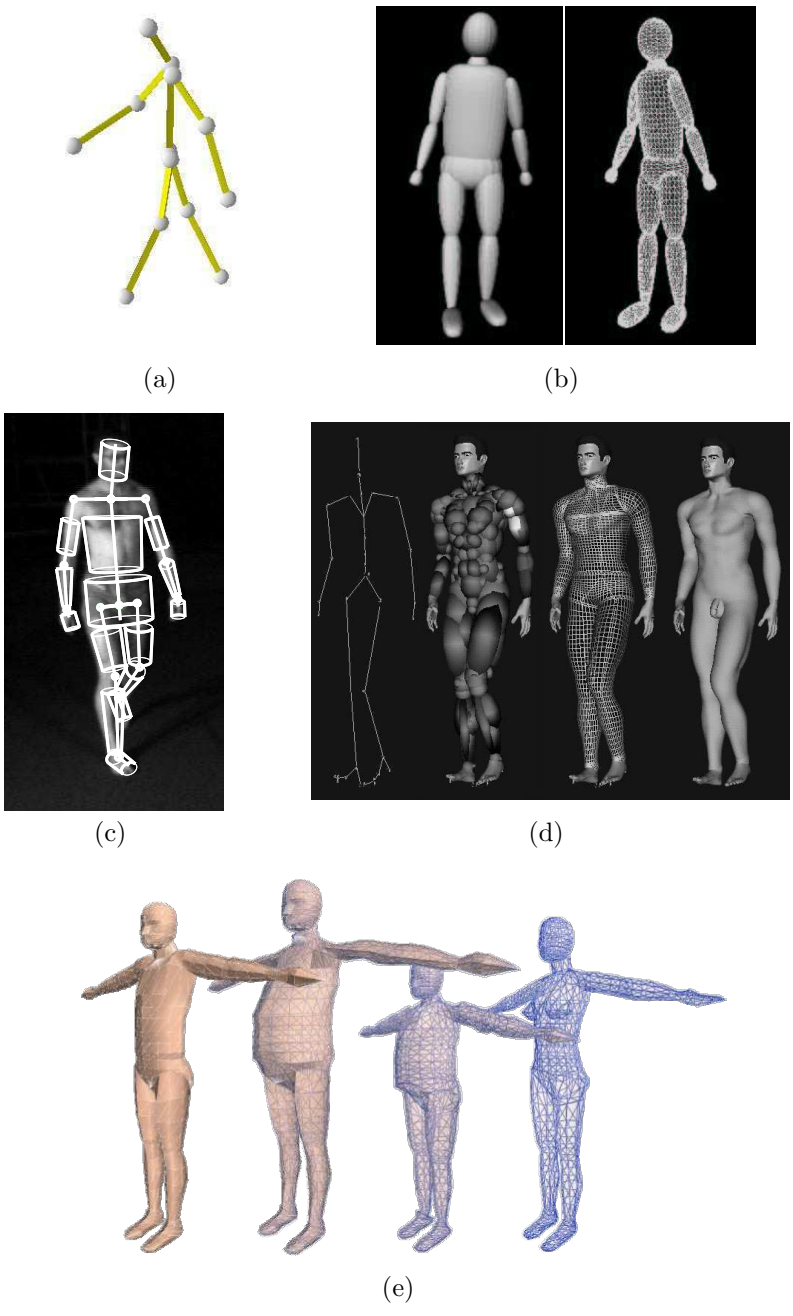


FIG. 1.4 – Exemples de modèles 3D du corps humain.
a : modèle bâtons de [72]. **b :** modèle avec des superquadriques de [105]. **c :** modèle composé de cônes à section elliptique de [31]. **d :** modèle à plusieurs niveaux proposé dans [96]. **e :** modèle anthropométrique RAMSIS utilisé dans [9].

silhouette et la projection du modèle en résolvant les équations dynamiques du mouvement. Dans le cas de l'estimation de la pose, la fonction que l'on cherche à maximiser est généralement très complexe : l'espace dans lequel se fait la résolution est de dimension élevée, et la fonction de coût présente souvent de nombreux optima locaux. Il n'y a donc souvent pas d'autre possibilité que de chercher l'optimum en partant de l'estimation de l'image précédente dans la vidéo ; le gros inconvénient de ces méthodes est donc qu'elles nécessitent une initialisation pour la première vue de la séquence. Dans [38], Gavrila et Davis proposent une méthode générative pour estimer la pose du corps : un jeu de paramètres est utilisé pour générer une hypothèse sur la pose, et la concordance du modèle avec les observations est calculée. Pour limiter l'espace de recherche, la pose optimale est recherchée pas à pas dans le voisinage (discrétisé) d'une prédiction effectuée à partir des vues précédentes, et les différents paramètres sont estimés de manière hiérarchique : les positions du torse et de la tête sont estimées en premier en supposant que les membres sont dans la configuration prédite, puis la configuration des autres parties du corps est estimée, etc. Ce type d'approche reste toutefois extrêmement coûteux.

Une autre grande famille d'approches consiste à effectuer l'estimation dans le cadre d'un suivi bayésien (méthodes type filtre de Kalman ou filtre à particules) : nous reviendrons sur ces méthodes dans le paragraphe 1.3.6. Concernant les méthodes de résolution par échantillonnage, on peut citer les travaux de [62, 63], dans lequel une échantillonnage MCMC (Monte Carlo par chaînes de Markov) permet de rechercher la solution optimale dans l'espace des poses. Le point fort de la méthode est qu'elle fonctionne sur des images statiques, mais cela implique aussi des temps de calcul déraisonnables pour des applications temps-réel.

1.3.4 Méthodes basées sur des exemples

Contrairement aux approches décrites dans le paragraphe précédent, ces méthodes fournissent directement une estimation de la pose 3D à partir des données bas niveau de l'image, sans passer par des prédictions sur un modèle. Elles reposent sur l'utilisation d'une base d'exemples créée à l'avance, qui contient un ensemble de paires image-pose. Les données d'entrainement sont généralement obtenues à partir de logiciels d'animation d'avatars et de rendu 3D (Poser, Maya...). Aucune modélisation explicite du corps humain (sur l'apparence des membres ou les contraintes de la chaîne cinéma-

tique) n'est utilisée, toutes ces données sont implicitement modélisées dans la construction de la base d'exemples.

Les principaux avantages de ces techniques sont d'une part qu'elles évitent lors de l'estimation de passer par l'optimisation d'une fonction de vraisemblance complexe et d'effectuer des rendus d'un modèle du corps, et d'autre part qu'elles fonctionnent généralement à partir d'une seule image et qu'aucune initialisation n'est nécessaire. L'inconvénient est qu'elles sont limitées à la reconnaissance de postures voisines de celles contenues dans la base d'exemples (même dans le cas de méthodes possédant de bonnes propriétés de généralisation).

On distingue deux grandes classes de méthodes : les méthodes non paramétriques basées sur une comparaison aux exemples de la base (paragraphe 1.3.4), dans lesquelles la base d'exemples est explicitement stockée en mémoire et sert ensuite de référence pour comparer les nouveaux exemples, et les méthodes basées sur un apprentissage (paragraphe 1.3.4), dans lesquelles un entrainement, effectué hors ligne, permet de générer un modèle paramétrique qui généralise les propriétés de la base d'exemples.

Méthodes basées sur une comparaison aux exemples

Ces travaux représentent l'espace des poses par un nombre fini d'exemples qui couvrent l'ensemble des états possibles. Chaque exemple de pose est associé à un descripteur dans l'espace image. Pour retrouver la pose étant donnée une nouvelle image, son descripteur est comparé aux exemples de la base. La pose estimée est celle correspondant à l'exemple de la base qui a le descripteur le plus proche. On peut aussi choisir de garder les n exemples les plus proches, puis interpoler une nouvelle pose. Ainsi, dans [88], une pose est interpolée à partir des 25 exemples les plus proches. Dans cet article, ses auteurs comparent l'utilisation de plusieurs descripteurs de la silhouette extraite (coefficients de Fourier du contour de la silhouette, Shape Context et moments de Hu). Plus récemment, les auteurs proposent dans [87] une approche similaire en se basant sur des histogrammes d'orientation de gradient calculés sur la zone de l'image délimitée par la boîte englobante de la silhouette extraite.

Dans [75], Mori et Malik utilisent une base d'images dans lesquelles les positions 2D de 14 points de référence du corps (mains, épaules, coudes, hanches, genoux, pieds et taille) ont été annotées manuellement. La description des images est basée sur des histogrammes de Shape Context calculés sur des points échantillonnés le long des contours internes et externes de la sil-

houette. Etant donnée une nouvelle image, l'exemple le plus proche de la base est recherché pour une distance définie à parti du Shape Context, et les positions 2D des points de référence de l'image test sont obtenues en déformant l'exemple de référence suivant une chaine cinématique 2D. La configuration 3D des points du corps est ensuite évaluée à partir des positions 2D grâce la méthode exposée dans [109].

Les méthodes basées sur la recherche des plus proches voisins ont deux inconvénients principaux : la quantité d'espace nécessaire pour stocker la base d'exemples, et le temps de calcul associé à la recherche des meilleurs exemples dans la base. Plus la dimension de l'espace à couvrir est grande, plus le nombre d'exemples nécessaires pour bien décrire l'espace est important (le lien entre les deux est exponentiel). Or, le temps de calcul associé à la recherche (naïve) des plus proches voisins croit linéairement en fonction du nombre d'exemples. Dans [98], Shakhnarovich propose une méthode pour accélérer significativement la recherche des exemples les plus proches dans la base. La pose du haut du corps est estimée à partir d'un descripteur du type HOG. La recherche des plus proches voisins est basée sur le *Parameter Sensitive Hashing*. Il s'agit d'une technique similaire au boosting qui vise à réduire le temps de recherche des plus proches voisins : plusieurs classificateurs binaires sont construits et les éléments les plus proches d'un nouvel exemple sont recherchés dans l'intersection des parties de l'espace situées du même côté que l'exemple pour tous les classificateurs. Une contribution de ce travail est la construction d'un espace de description, appris à partir des exemples, dont la structure reflète la proximité des exemples dans l'espace des paramètres de la pose. Lorsque les plus proches voisins ont été trouvés, la pose est estimée par une régression locale pondérée. Dans ces travaux, une base de 150 000 exemples est construite avec POSER.

Une autre façon de gérer la base d'exemples est le modèle de distribution de points (*Point Distribution Model*, PDM), proposé dans [17] et [80]. Dans ces travaux, la pose du corps est estimée pour une personne faisant face à la caméra. Chaque exemple est représenté par un vecteur de grande dimension qui regroupe des données image (les positions 2D des blobs correspondant au visage et aux mains et les coordonnées de points échantillonnés le long du contour) et la configuration du squelette 3D correspondant. Une première ACP est utilisée pour localiser le sous-espace dans lequel les exemples sont répartis, puis l'espace est partitionné en plusieurs sous-espaces afin de modéliser les non-linéarités de l'espace global. Dans chaque groupe, une nouvelle ACP détermine l'espace des combinaisons linéaires possibles. Pour chaque nouvel exemple, les données image sont utilisées pour rechercher le groupe le plus

proche, puis projeter l'exemple dans ce groupe pour retrouver la configuration la plus plausible. Dans [78], la méthode est étendue au cas multi-caméras. Dans [80], la même approche est utilisée pour faire un suivi temporel de la pose, en fusionnant l'algorithme avec un modèle de Markov modélisant les transitions possibles entre les groupes. L'algorithme de CONDENSATION permet d'estimer à l'intérieur d'un groupe les coefficients de la combinaison linéaire.

Méthodes basées sur un apprentissage

Dans ces approches, la base d'exemples est utilisée dans une phase d'apprentissage pour construire un modèle paramétrique capable de prédire une configuration du corps à partir de l'image. Plusieurs façons de construire ce modèle peuvent être envisagées : dans certains travaux, l'apprentissage permet de construire une application reliant un descripteur de l'image à la pose correspondante. Dans d'autres, l'espace des configurations possibles du corps est modélisé, soit par une densité de probabilité a priori, soit par une sous-variété, et la pose la plus plausible est ensuite recherchée dans cet espace.

Dans le premier cas, une régression génère une application compacte qui permet de passer directement des données image à la pose 3D. L'image est représentée par un descripteur qui condense dans un vecteur de petite dimension l'information utile pour estimer la pose. Les différentes approches diffèrent sur le choix du descripteur utilisé (moments de Hu, histogrammes de Shape Context, HOG...) et dans celui de l'algorithme d'apprentissage (réseaux de neurones, SVM, RVM...).

Dans [93] et [94], Rosales et Sclaroff utilisent des applications spécialisées pour estimer la position dans l'image des articulations principales du corps. Dans ces travaux, les silhouettes sont caractérisées par les 7 moments de Hu. Un algorithme de partitionnement permet de diviser l'espace des observations en un ensemble de groupes. Pour chaque groupe, une application (modélisée par un réseau de neurones) permet de relier une observation à une pose 2D. Lorsqu'une nouvelle observation est présentée en entrée du modèle, son image par chacune de ces applications est calculée. Chaque solution possible est utilisée pour animer un modèle simple du corps et générer un rendu. La solution retenue est celle qui a généré le descripteur le plus proche de la donnée d'entrée. Les auteurs proposent aussi d'intégrer un critère de cohérence temporelle pour améliorer la robustesse. Dans [95] une généralisation de la méthode avec un système de caméras non-calibrées est présentée : pour chaque caméra, la méthode précédente donne des hypothèses sur les positions

2D des points du corps. Celles-ci sont ensuite combinées pour remonter à une estimation de la position des points du corps en 3D.

Dans [7], des méthodes de régression non-linéaire permettent de faire la relation entre les histogrammes de Shape Context calculés sur les contours externes de la silhouette et la pose 3D. Différentes méthodes de régression sont comparées : moindres carrées pénalisés, SVM et RVM, avec des noyaux linéaires ou gaussiens. Dans [29], la même approche est utilisée pour retrouver la pose de la main à partir d'un système de 3 caméras. Le descripteur est constitué de la concaténation des histogrammes des 3 images. Dans [106], ce type de méthode est également employé pour estimer la pose 3D à partir d'une reconstruction 3D en voxels obtenue à partir d'un système de plusieurs caméras.

Dans [6], Agarwal et Triggs proposent une méthode adaptée au cas où la silhouette ne peut pas être extraite proprement du fond. La description est basée sur des histogrammes d'orientation du gradient calculée sur une grille dense de l'image. Un ensemble d'exemples est constitué avec des images d'une personne réalisant des gestes devant un fond vide. Une factorisation en matrices non-négatives permet de construire une base de vecteurs sélectionnant les caractéristiques pertinentes dans la description d'une image (celles qui correspondent à des parties importantes du corps comme les épaules ou les coudes). L'utilisation de cette base avec des images encombrées permet ensuite de capturer automatiquement les contours de la personne. La pose du haut du corps est estimée par une méthode de régression analogue à [7]

Dans [103, 104], Sminchisescu et al. modélisent par un apprentissage sur des données de synthèse (Maya) la distribution conditionnelle de la pose 3D du corps étant donné le vecteur descripteur de la silhouette (des histogrammes de Shape Context). Pour tenir compte des ambiguïtés liées à l'estimation monoculaire, la distribution apprise est multi-modale (*Bayesian Mixture of Experts*).

De nombreux travaux modélisent l'espace des configurations possibles du corps par une variété. L'un des premiers à proposer ce type de méthode a été Brand dans [20]. La variété encodant des configurations de la pose et de la vitesse est modélisée par une chaine de Markov cachée, apprise par une minimisation d'entropie. Pour une nouvelle séquence, le système calcule la trajectoire dans l'espace des configurations 3D la plus compatible à la fois avec la dynamique apprise et l'ensemble des silhouettes observées : l'algorithme de Viterbi permet de retrouver la succession des états cachés la plus cohérente avec l'ensemble des observations. Toutes les images de la séquence

sont traitées a posteriori.

De manière similaire, dans [32], Elgammal et Lee modélisent par une variété l'ensemble des configurations du corps au cours du temps pour une activité particulière. Plusieurs sous-variétés de l'espace des observations, correspondant à des mouvements de marche sous différents points de vue, sont apprises à partir de données d'entrainement. Les applications permettant de passer de ces variétés aux observations de l'image et aux poses 3D sont ensuite apprises. Leur méthode passe par l'intermédiaire de cette sous-variété à la fois pour estimer la pose, déterminer le point de vue, et reconstruire les données observées et ainsi repérer d'éventuelles incohérences temporelles dans ces observations. Comme ces variétés sont apprises pour des mouvements bien spécifiques (séquences de marche) et pour un nombre de points de vue limités, la méthode est difficilement généralisable à des mouvements moins contraints.

Les auteurs de [108] proposent d'apprendre la variété des poses via une KPCA (*Kernel Principal Component Analysis*), et une autre variété correspondant aux apparences des silhouettes dans les images. L'application allant de la variété image à la variété des poses est également apprise. Le vecteur utilisé pour décrire les images est un descripteur pyramidal, qui représente, à plusieurs résolutions, la proportion de l'aire occupée par la silhouette dans une sous-image de sa boîte englobante. Un avantage de cette approche est que les silhouettes sont automatiquement débruitées lorsque le descripteur est projeté sur un espace de dimension plus faible par KPCA.

Une autre façon de prendre en compte les données apprises est de s'en servir pour modéliser une distribution de probabilité a priori qui vient compléter les observations de l'image et guider l'estimation. Dans [51], les auteurs utilisent ainsi un apprentissage pour compenser la perte d'information liée à la projection monoculaire. Un suivi 2D permet d'identifier la position de différents points du corps dans l'image, et la probabilité apprise permet ensuite d'estimer une configuration 3D plausible.

Grauman et al. [42] modélisent aussi les données d'apprentissage par une distribution a priori pour estimer les positions en 3D des points du corps à partir d'un ensemble de silhouettes provenant de différentes caméras. La distribution a priori est construite sur un vecteur de grande dimension comprenant à la fois des points échantillonnés sur les contours externes des silhouettes et les positions 3D de points de référence du corps (de manière similaire à [17] et [80]). La distribution est modélisée par des MPPCA (*Mixtures of probabilistic principal component analysers* [115]). Comme dans [108], l'uti-

lisation d'ACP et de projections sur des espaces de dimension réduite permet de reconstruire et débruiter les données d'entrée. Etant donnée une nouvelle observation, la pose est estimée en calculant le maximum de la densité a posteriori, qui se base à la fois sur la vraisemblance des observations et la densité a priori apprise. Comme le vecteur entier {observations + pose} est reconstruit lors de l'estimation, la méthode peut fonctionner selon les auteurs sur des données bruitées, et même avec des silhouettes manquantes.

1.3.5 Méthodes basées sur la détection et l'assemblage des différentes parties du corps

Dans ces approches, la pose du corps est estimée en détectant dans un premier temps des positions vraisemblables des parties du corps dans l'image, puis en cherchant à assembler ces hypothèses de façon cohérente pour obtenir la configuration du corps qui correspond le mieux aux observations. Contrairement aux méthodes décrites dans le paragraphe 1.3.3, ces méthodes ne reposent pas sur l'utilisation d'un modèle explicite du corps humain, décrivant la forme et l'apparence des différentes parties du corps, mais sur un modèle faible, c'est-à-dire un simple critère de cohérence spatiale entre les positions des différentes parties du corps dans l'image. Au lieu de traiter le corps comme un arbre cinématique contenant un grand nombre de degrés de liberté et des parties rigidement connectées, chaque membre du corps est d'abord traité indépendamment des autres, et des contraintes souples lient les positions des parties adjacentes. On trouve un exemple de ce type d'approche dans [76], dans lequel les auteurs utilisent dans un premier temps une segmentation bas-niveau de l'image pour guider la détection des parties du corps : les pixels semblables de l'image sont d'abord regroupés en régions susceptibles de contenir un segment du corps, puis la détection est affinée, et des combinaisons plausibles de ces différentes parties sont recherchées selon des critères de proximité des membres du corps, de cohérence de leurs tailles relatives, de couleur, d'adjacence, de symétrie sur la couleur.

De nombreuses méthodes reposent sur le concept de *Pictorial Structure* introduit dans [34], dans lequel le corps est modélisé par un graphe dont les noeuds sont les différentes parties du corps, et dont les arêtes représentent les interactions entre les parties du corps adjacentes (des liaisons "ressort"). L'un des premiers travaux utilisant cette technique est [33], dans lequel la configuration optimale des parties du corps est estimée en minimisant une fonction de coût mesurant d'une part la cohérence entre la position des parties du corps et l'image (critère sur la couleur des parties du corps qui a

été apprise sur un exemple), et d'autre part à quel point les positions des membres s'accordent avec le modèle du corps. La configuration optimale est recherchée grâce à des outils de programmation dynamique. En dehors des liaisons entre parties adjacentes, d'autres contraintes peuvent être ajoutées à cette structure de graphe, comme par exemple dans [59] où les auteurs introduisent dans le graphe des variables latentes pour tenir compte de la coordination entre les différentes parties du corps.

Concernant l'étape de détection des membres du corps dans l'image, différentes stratégies de reconnaissance ont été employées. Dans [90], les auteurs construisent une modèle d'apparence (basé sur la couleur) des membres du corps à partir de la vidéo. Dans d'autres travaux, un apprentissage est réalisé à partir d'une base d'exemples labellisés : dans [70] le détecteur est basé sur un apprentissage par AdaBoost, et dans [92], un classifieur SVM est utilisé. Dans [91], les membres sont détectés en calculant la réponse d'un filtre construit à partir d'un modèle probabiliste de forme pour les parties du corps.

L'inconvénient principal de ce type de méthode est la complexité calculatoire de la recherche de la solution optimale parmi toutes les configurations possibles. Un autre handicap est que, comme elles s'appuient sur un analyse de l'apparence des membres du corps dans l'image, la gestion des auto-occultations est généralement difficile. Dans [52], Ioffe et Forsyth proposent d'utiliser des mixtures d'arbres pour mieux gérer les parties absentes ou occultées de l'image et accélérer la recherche de la configuration optimale. Dans [58], les auteurs introduisent des modèles à couches dans lesquels les phénomènes d'occultations entre les membres du corps sont explicitement modélisés.

La plupart de ces méthodes se limitent à une analyse en 2D de la position des membres du corps dans l'image. Dans [102], les auteurs étendent ce type d'approche pour évaluer une configuration du corps en 3D à partir d'un système multi-caméras. Ils introduisent pour cela un modèle 3D du corps (*loose-limbed body model*) dans lequel les parties du corps sont paramétrées par des coefficients de position et de rotation, et ne sont pas rigidement connectées mais "attirées" les unes vers les autres par des contraintes souples. L'estimation en 3D impose de recourir à d'autres méthodes de résolution que dans le cas 2D (*non-parametric belief propagation*).

1.3.6 Utilisation du temps dans une séquence vidéo

Lorsque l'estimation est réalisée à partir des images d'une séquence vidéo, il est possible d'utiliser l'information de cohérence temporelle entre les images

consécutives de la séquence. Il existe plusieurs manières de la prendre en compte. Dans certain cas (paragraphe 1.3.6), l'estimation est entièrement basée sur l'exploitation d'une continuité du mouvement du corps dans le temps, alors que dans d'autres (paragraphe 1.3.6), la cohérence temporelle n'est pas indispensable mais vient simplement compléter et améliorer une estimation effectuée à partir d'une seule image.

Méthodes de suivi

Elles consistent à maintenir la cohérence temporelle de la pose estimée le long d'une séquence vidéo, en s'appuyant sur l'hypothèse que les configurations du corps qui correspondent à deux images successives sont voisines l'une de l'autre. Un problème inhérent à ce type de méthode est celui de l'initialisation (ou de la ré-initialisation en cas de perte) : celle-ci peut être manuelle ou faire l'objet d'un traitement spécial à la première vue de la séquence. L'approche la plus communément employée est de placer l'estimation dans le cadre d'un suivi bayésien. Une prédiction de la pose est effectuée à partir de l'estimation de l'image précédente en se basant sur une loi d'évolution du mouvement, et le vecteur d'état est ensuite mis à jour en fonction des observations de l'image courante. Ces techniques permettent d'obtenir une mesure de l'incertitude sur la prédiction. Un certain nombre d'approches sont basées sur l'utilisation d'un filtre de Kalman ou d'un filtre de Kalman étendu pour suivre la pose au cours du temps [30, 71, 119, 123].

L'emploi du filtre de Kalman reste toutefois limité aux situations où la distribution de probabilité du vecteur d'état du modèle est unimodale, c'est-à-dire qu'il ne permet de propager au cours du temps qu'une seule hypothèse sur la configuration du corps. Or, l'estimation de la pose est un problème intrinsèquement mal posé, qui est sujet à de nombreuses ambiguïtés visuelles dues aux phénomènes d'auto-occultations, à la présence éventuelle dans l'image d'objets ressemblant à l'objet suivi etc., en particulier dans le cas monoculaire. En cas d'ambiguïté, l'estimation risque de sélectionner la mauvaise pose, occasionnant une perte du suivi difficilement récupérable. Une alternative est donc d'utiliser des algorithmes de suivi à hypothèses multiples, c'est-à-dire permettant de modéliser des distributions multi-modales. Une des méthodes les plus reconnues est l'algorithme de CONDENSATION [53], ou filtre à particules. L'algorithme est basé sur un échantillonnage de la densité a posteriori estimée à la vue précédente. Les échantillons sont ensuite propagés suivant un modèle de mouvement (prédiction), et le poids des particules est évalué grâce à la fonction de vraisemblance de la vue courante

(mise à jour). Le filtre à particules permet de modéliser les sources d'incertitude du problème, car les configurations du modèle les moins vraisemblables ne sont pas immédiatement éliminées, elles sont gardées et propagées, ce qui leur laisse une chance de s'exprimer plus tard dans la séquence. Le suivi qui en résulte est plus robuste en cas de mouvements rapides, de disparition momentanée de l'objet à suivre, de bruit dans les images... Le principal inconvénient de ce type de suivi est sa complexité calculatoire. Le nombre de particules nécessaires pour suivre l'évolution d'un vecteur d'état croit en effet exponentiellement avec sa dimension. Pour chaque particule, le modèle doit être rendu dans l'image et la fonction de vraisemblance évaluée. Même pour les modélisations les moins complexes, le nombre de degrés de liberté est trop élevé pour envisager d'appliquer l'algorithme tel quel. Cham et Rehg [24] proposent comme alternative une technique d'estimation hybride : la distribution a posteriori est représenté par des morceaux de gaussiennes, et les modes de cette distribution sont propagés suivant un procédé similaire à la prédiction d'un filtre de Kalman. Cette prédiction est ensuite échantillonnée et la vraisemblance est évaluée en chaque échantillon.

Une solution pour alléger le coût de ces méthodes est de réduire le nombre de particules nécessaires en cherchant à les répartir plus efficacement dans les régions intéressantes de la fonction à évaluer (les maxima locaux). Ainsi, Deutscher et al. [31] proposent une méthode inspirée du recuit simulé pour guider les particules sur le maximum global de la densité a posteriori (on peut reprocher à la méthode de diminuer l'aspect multi-modal du filtre à particules). Dans [105], la covariance a posteriori de la vue précédente est utilisée pour attirer les particules dans les régions où l'incertitude est plus grande, qui correspondent par exemple aux zones d'ambiguïtés visuelles.

Une autre solution pour diminuer le nombre de particules est la technique de *Partitioned Sampling*, introduite dans [67], qui consiste à diviser l'espace en plusieurs partitions indépendantes de dimensions plus petites : dans le cas d'un objet articulé comme le corps humain, il s'agit de décomposer l'estimation de manière hiérarchique en estimant d'abord le torse, puis les bras, les jambes... en descendant le long de la chaîne articulée.

Couplage du suivi et de l'estimation image par image.

Dans certains cas, une méthode d'estimation fonctionnant à partir d'une seule image peut être avantageusement complétée en mettant à profit la cohérence temporelle entre les images successives d'une séquence vidéo. Dans [7], Agarwal et Triggs proposent d'améliorer leur système d'estimation en in-

tégrant dans la régression une prédiction sur l'état courant calculée à partir des deux estimations précédentes dans la vidéo. Deux phases d'apprentissage sont nécessaires : une première régression permet de calculer les coefficients du modèle dynamique utilisé pour les prédictions (processus linéaire autoregressif du second ordre), et un second apprentissage permet de générer un modèle capable de produire une estimation de la pose en tenant compte à la fois des observations de l'image et de l'état prédit. Dans ces travaux, l'utilisation de la cohérence temporelle permet d'une part de lisser les estimations au cours du temps et d'autre part de supprimer des erreurs provenant d'ambiguïtés visuelles (estimation monoculaire).

Dans [104], les auteurs modélisent pour un instant t donné la relation entre une observation $\mathbf{r_t}$ et la pose 3D correspondante $\mathbf{x_t}$ par une distribution de probabilité multi-modale, apprise à partir de données d'entrainement. La méthode qu'ils proposent est valable soit dans le cas où la pose est estimée uniquement à partir de l'observation courante (en apprenant $p(\mathbf{x_t}|\mathbf{r_t})$), soit en prenant en compte l'estimation faite à l'instant précédent (en apprenant $p(\mathbf{x_t}|\mathbf{x_{t-1}}, \mathbf{r_t})$).

Dans [100], la méthode d'estimation proposée dans [102] est étendue en ajoutant des contraintes temporelles dans le modèle graphique décrivant les interactions entre les différentes parties du corps. Dans ces travaux, l'estimation temporelle bénéficie de la méthode d'évaluation de [102] pour l'initialisation ou la réinitialisation en cas de perte du suivi.

Méthodes a posteriori

Dans certains travaux, l'estimation est faite globalement sur l'ensemble des images d'une vidéo. Ainsi l'estimation de la pose pour l'une des images bénéficie de la connaissance à la fois des images précédentes et des images suivantes de la séquence. Ces méthodes ne sont pas adaptées pour un suivi en ligne. Dans [20], l'algorithme de Viterbi est utilisé pour retrouver la séquence des états successifs la plus probable d'une chaine de Markov cachée. [97] propose d'adapter des techniques d'ajustement de faisceaux pour retrouver la configuration d'un modèle 3D correspondant à chacune des vues. Dans [96], une première étape de suivi permet d'initialiser un processus de raffinement de la pose effectué sur l'ensemble des vues de la vidéo.

1.4 Position des travaux présentés

Nous proposons un système d'estimation de la pose d'un modèle articulé du corps à partir des images acquises par un système de plusieurs caméras (4 ou 5) statiques et calibrées. Comme beaucoup d'autres travaux sur cette application, nous avons choisi de baser les observations sur les silhouettes extraites dans les différentes images par un algorithme de soustraction de fond. Cette approche est envisageable dans le cas où les caméras sont fixes et où l'environnement dans lequel évolue la personne est relativement stable au cours du temps. La silhouette peut alors être calculée de manière robuste, et représente une primitive intéressante car elle est invariante aux changements d'éclairage et d'habillement.

Les silhouettes 2D sont ensuite combinées par une reconstruction en 3D de l'enveloppe visuelle : la silhouette 3D fusionne toutes les informations du système (données images et calibrage du système) en un seul élément, et rend l'estimation plus indépendante de la configuration des caméras. Il faut préciser que notre système d'acquisition n'est pas synchronisé électroniquement : il s'agit d'un réseau IP représentatif des réseaux de vidéo surveillance. Les images venant des différents capteurs sont associées en fonction de leur moment d'arrivée au PC de calcul. Il en résulte qu'un décalage de quelques ms peut exister entre les acquisitions, ce qui peut induire des imprécisions dans la silhouette 3D reconstruite.

Pour évaluer la pose du corps à partir de l'enveloppe visuelle reconstruite, notre choix s'est porté sur une méthode d'estimation par régression : cette méthode évite d'une part les calculs liés à des prédictions sur la configuration d'un modèle du corps et à l'optimisation d'une fonction de vraisemblance complexe, et permet d'autre part de déterminer la pose à partir d'une seule image, et de se dispenser des problèmes d'initialisation ou de perte de suivi.

Une vue d'ensemble de la méthode proposée est donnée sur la figure 1.5. Pour chaque caméra, on suppose qu'un apprentissage préalable a permis de modéliser une image du fond. La silhouette de la personne est extraite dans chaque image. La connaissance des paramètres de calibrage intrinsèques et extrinsèques nous permet ensuite de synthétiser les données image en une reconstruction 3D de l'enveloppe visuelle. La suite de notre méthode est basée sur un algorithme d'apprentissage. On suppose qu'on dispose d'une base d'exemples contenant d'un côté des données image de silhouettes et de l'autre la vérité terrain sur la posture de la personne. Cette base nous permet de modéliser par apprentissage l'application permettant de passer d'une silhouette 3D à la posture. Les caractéristiques de la forme 3D sont encodées par un

descripteur 3D, qui synthétise en un vecteur compact l'information de la reconstruction. A l'estimation, la reconstruction 3D est calculée à partir des silhouettes extraites et son descripteur est donné directement en entrée du modèle qui a été appris, pour fournir une estimation de la pose.

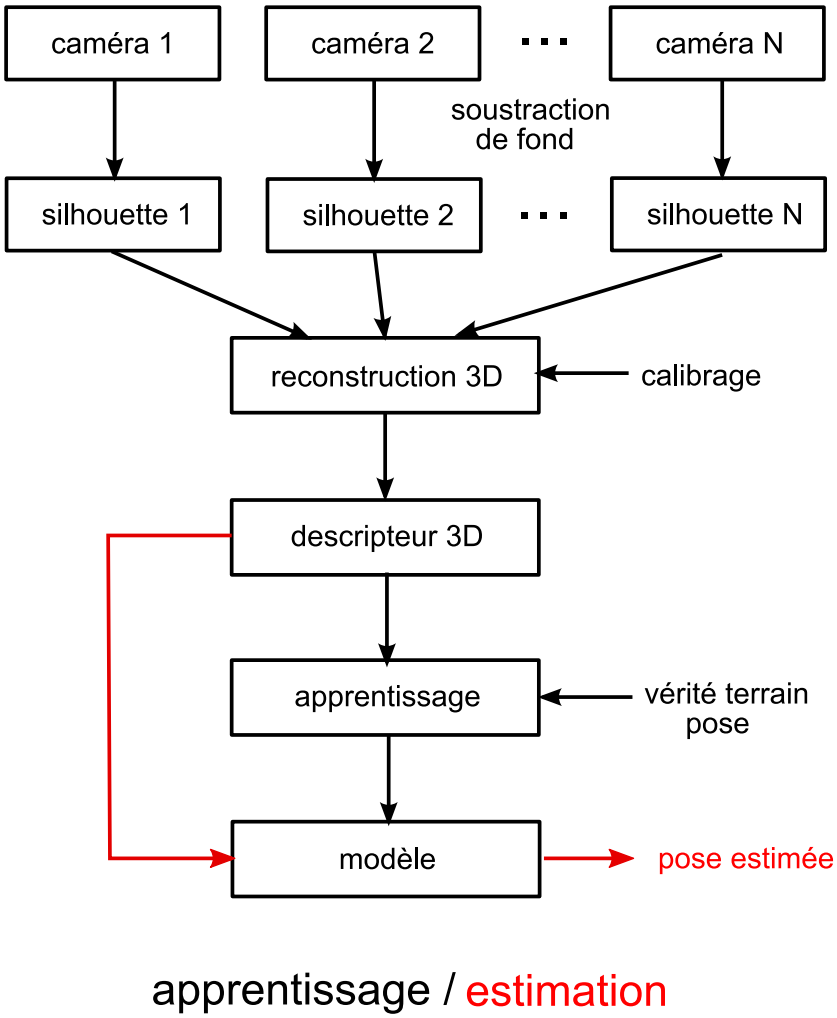


FIG. 1.5 – Vue d'ensemble de notre approche.

Chapitre 2

Outils pour l'estimation de posture

2.1 Introduction

Ce chapitre présente différents outils impliqués dans notre système d'estimation. La méthode choisie suppose de modéliser la relation entre les données image et la pose du corps humain. Dans la première partie 2.2 sont présentées les différentes paramétrisations possibles du mouvement d'un humain et le modèle qui a été retenu dans cette thèse. La deuxième partie 2.3 décrit l'ensemble du processus menant à la reconstruction 3D de la silhouette du corps à partir des images acquises par les caméras : la méthode d'extraction des silhouettes 2D d'une part, et la manière dont elle sont fusionnées pour reconstruire l'enveloppe visuelle d'autre part. La dernière partie 2.4 explique comment ont été synthétisées les bases d'exemples utilisées pour l'apprentissage et les tests.

2.2 Paramétrisation des mouvements

Dans notre approche, la posture est définie par un vecteur contenant un certain nombre de paramètres dont les valeurs déterminent la configuration spatiale des membres du corps. Les paramètres de ce vecteur constituent la sortie de la machine de régression. Différentes formes de paramétrisation des mouvements du corps humain sont possibles.

2.2.1 Etat de l'art

Le corps est généralement représenté par un arbre cinématique constitué de segments rigides de tailles différentes, reliés entre eux par des articulations qui possèdent chacune un certain nombre de degrés de liberté. Le vecteur de pose englobe à la fois l'orientation du corps dans l'espace (ou le point de vue par rapport à la caméra), et la configuration de chacune des articulations du squelette. Plusieurs façons de représenter une posture ont été proposées.

Paramétrisation par des angles

La configuration de chaque articulation est paramétrée par 3 angles, comme les angles d'Euler ou des angles de rotation autour des axes x, y, z d'un repère local. Ces 3 angles encodent la position d'un segment de l'arbre dans un repère lié au segment parent. La pose est représentée par un vecteur réunissant les angles de toutes les articulations par rapport à une position neutre. Comme cette représentation est indépendante de l'échelle, elle peut facilement être transposée d'un squelette à l'autre. Elle permet aussi de distinguer l'orientation globale du corps (ou le point de vue) des angles internes décrivant la configuration des différents segments, puisqu'elle les encode séparément, par des paramètres différents. Si les repères locaux sont choisis judicieusement, elle peut permettre de bloquer facilement des degrés de liberté inutiles sur certaines articulations (s'il est représenté par un des 3 angles).
Cette paramétrisation possède néanmoins quelques inconvénients. Premièrement, comme les opérations sur les 3 angles ne sont pas commutatives, la paramétrisation dépend de l'ordre dans lequel les 3 angles de rotation apparaissent dans le vecteur ; les trois paramètres d'une articulation dépendent les uns des autres et ne peuvent pas réellement être considérés séparément. Elle peut ensuite souffrir de discontinuités :

- si les valeurs d'un angle peuvent parcourir tout l'intervalle allant de 0° à 360°, la paramétrisation ne peut pas être continue ; elle "saute" lorsque sa valeur passe au voisinage de 360°. Pour y remédier, une solution est de paramétrer cet angle par l'intermédiaire des fonctions *cos* et *sin* (l'angle est alors représenté par deux paramètres au lieu d'un).
- plusieurs combinaisons de 3 angles peuvent représenter la même rotation 3D.

Par ailleurs, un vecteur représenté par la concaténation de tous les angles n'encode pas la notion de hiérarchie entre les articulations. Le squelette a une structure d'arbre et, suivant leur position dans la structure, tous les angles n'ont pas la même influence sur la pose. Une articulation a des réper-

cussions sur tous les membres situés en aval dans la chaine. Par exemple, une rotation de 90° de l'orientation globale du corps ou d'un angle de l'épaule n'a pas le même effet sur l'enveloppe qu'une même rotation sur le coude ou le poignet. Enfin, le changement de la pose associé à l'évolution d'un angle est dépendant de la pose elle-même. Par exemple, si deux personnes sont debout, l'une avec le bras tendu et l'autre avec les bras le long du corps, un changement de l'orientation du corps par rapport à l'axe vertical n'aura pas la même influence sur l'enveloppe. De même, une variation de l'angle de l'épaule autour de l'axe du bras aura un effet sur la pose si le coude est plié, tandis qu'elle restera quasiment imperceptible si le bras est tendu.

La paramétrisation de la pose basée sur les angles est une représentation couramment utilisée. C'est notamment la représentation choisie par [7, 106]. Les auteurs de [106] proposent en outre d'utiliser une ACP sur l'ensemble des vecteurs des poses de la base d'apprentissage pour réduire la dimension des données et ne conserver dans le processus d'apprentissage que les angles significatifs de la pose.

L'utilisation des quaternions peut être une alternative intéressante pour représenter la pose par des angles. Elle évite certains inconvénients de la paramétrisation précédente : les quaternions facilitent par exemple la composition des rotations et évitent les ambiguïtés de *Gimbal Lock* (cas singuliers avec les angles d'Euler). Les quaternions sont par exemple utilisés dans [50], [79] et [71].

Paramétrisation par les positions 3D des points du squelette

Une alternative pour représenter la pose est d'utiliser un vecteur contenant les positions x, y, z des articulations dans un repère 3D. C'est par exemple l'approche adoptée dans [6, 42]. On peut facilement passer d'une représentation à l'autre si on connait la structure de la chaîne cinématique et la longueur des différents segments. Cette représentation est dépendante de l'échelle et de la morphologie du squelette, et doit donc être normalisée si on souhaite la rendre invariante aux changements d'échelle. Un avantage de cette paramétrisation est qu'elle semble plus représentative des variations globales de la pose : la distance euclidienne entre deux vecteurs est à l'image de la différence d'apparence des enveloppes. En revanche, la position de points 3D ne permet pas, contrairement aux angles, de distinguer le point de vue des angles des membres du corps : toutes ces informations sont mélangées dans la position dans l'espace des articulations. Une solution intermédiaire pour y remédier pourrait être de coder d'une part l'orientation du corps dans l'es-

pace par un angle, et d'autre part les positions 3D des articulations dans un repère lié au corps, c'est-à-dire recalé par rapport à cette orientation.
Un autre inconvénient important est que l'utilisation des trois coordonnées x, y et z indépendamment les unes des autres ne garantit pas que les valeurs obtenues en leur appliquant certaines opérations correspondront à une pose valide, c'est-à-dire telle que la longueur des segments sera maintenue constante. Par exemple, dans le cas où ces coordonnées sont évaluées par une application apprise par régression, si seulement l'une des trois coordonnées est mal estimée, l'estimation peut produire une pose complètement aberrante. Cette représentation contient en fait plus de paramètres qu'il n'en faut réellement pour encoder la pose (puisque la longueur des segments est fixée à l'avance).

2.2.2 Mesure de l'erreur sur la pose

Pour mesurer la qualité d'une pose estimée, il est nécessaire de construire un critère mesurant l'écart entre cette pose et la vérité terrain. Plusieurs façons de mesurer cette erreur ont été proposées. Une mesure de l'erreur peut être associée à chacune des paramétrisations possibles des mouvements, avec généralement les avantages et les inconvénients qui leur sont liés. Pour pouvoir comparer différentes approches, l'idéal est de définir une erreur facilement transposable d'une méthode à l'autre et rapide à calculer.

Mesure de l'erreur sur les angles

Si θ est le vecteur des angles estimés et $\hat{\theta}$ le vecteur des angles de la vérité terrain ($\theta, \hat{\theta} \in \mathbf{R}^m$), une mesure de l'erreur angulaire peut s'écrire :

$$D(\theta, \hat{\theta}) = \frac{1}{m} \sum_{i=1}^{m} min(\left|(\theta_i - \hat{\theta}_i\right|, 360 - \left|(\theta_i - \hat{\theta}_i\right|) \tag{2.1}$$

Une mesure de similarité normalisée est également proposée dans [98] :

$$D(\theta, \hat{\theta}) = \sum_{i=1}^{m} 1 - cos(\theta_i - \hat{\theta}_i) \tag{2.2}$$

L'inconvénient de ce type de mesure est qu'elle dépend du niveau de détail de la paramétrisation (c'est-à-dire des articulations représentées) et des degrés de liberté choisis ; il est alors difficile de comparer des méthodes qui n'utilisent pas les mêmes paramètres angulaires. Par ailleurs, tous les

angles sont mis sur un même pied d'égalité, sans tenir compte de la dépendance des uns par rapport aux autres. Or, toutes les erreurs angulaires n'ont pas la même signification selon la configuration des autres articulations. Par exemple, dans une méthode basée sur la reconstruction du corps humain en voxels, l'angle de l'épaule par rapport à son axe est difficilement estimable si le coude est tendu puisqu'il est quasiment imperceptible. Le critère d'erreur devrait dans ce cas peu prendre en compte une mauvaise estimation. En revanche, si le coude est plié, cet angle a une influence importante sur la forme de l'enveloppe, et une erreur importante doit être fortement pénalisée dans la mesure de l'erreur.

Mesure de l'erreur sur la position des points du corps

Dans [101], les auteurs proposent de définir une mesure qui puisse être facilement calculée quelque soit la paramétrisation et la méthode d'estimation utilisée. Elle s'appuie pour cela sur la position de marqueurs virtuels correspondant à certains points de référence situés sur le corps (à l'image des marqueurs que porte une personne dans une séquence de capture de mouvement).
Si la pose du corps est représentée par un ensemble de M points x_i de $\mathbf{R}^3$, la distance entre deux configurations $X = \{x_1, ..., x_M\}$ et $\hat{X} = \{\hat{x}_1, ..., \hat{x}_M\}$ peut s'écrire comme la moyenne des distances euclidiennes entre les marqueurs :

$$D(X, \hat{X}) = \sum_{i=1}^{M} \frac{\|x_i - \hat{x}_i\|}{M} \tag{2.3}$$

Deux méthodes utilisant des paramétrisations différentes peuvent facilement être comparées avec ce type de mesure, en définissant un ensemble de points du corps dont les positions peuvent être calculées avec les deux modélisations.

Une alternative peut être de mesurer l'erreur de projection de ces marqueurs dans les images (en pixels), par exemple si la méthode d'estimation est basée sur un modèle 2D.

Un inconvénient de ce critère est qu'une erreur sur l'une des articulations du squelette se répercute sur toutes les articulations qui la suivent dans la chaîne. En particulier, si l'orientation du corps est mauvaise, les erreurs sur tous les points seront fortes, même si le reste de la pose (configuration des membres du corps) est correctement estimée. On peut aussi envisager de calculer d'abord une erreur sur l'orientation globale du corps, puis de recaler

le squelette sur l'orientation de la vérité terrain avant de calculer une erreur sur les positions 3D des points.

2.2.3 Paramétrisations utilisées

Modélisation du squelette

Le modèle du corps humain adopté dans cette thèse est le squelette de l'avatar par défaut du logiciel POSER 6 (voir paragraphe 2.4). Dans ce logiciel, les avatars sont animés à partir de fichiers de capture de mouvement au format BVH (pour Biovision hierarchical data). Ce format a été développé à l'origine par la société Biovision, comme moyen de fournir des données de capture de mouvement à ses clients. La structure employée pour animer les personnages est un système hiérarchique, où chaque articulation correspond à un noeud dans la chaîne cinématique. La position et l'orientation de chaque articulation affectent la position et l'orientation des noeuds suivants dans la structure. Chaque articulation a trois degrés de liberté (DDL) correspondant aux angles de rotation autour des axes du système de coordonnées locales (défini par le noeud parent), excepté pour la racine de l'arbre, qui possède trois degrés de translation supplémentaires. La position globale d'un segment dans l'espace peut être obtenue en combinant les matrices de rotation de toutes articulations parentes dans l'arbre, en remontant jusqu'à la racine. Le fichier BVH est divisé en deux parties. La première partie décrit la chaîne cinématique en fixant la hiérarchie et l'échelle des différentes parties du corps. Cette structure est définie en donnant la position 3D des articulations du squelette dans sa position neutre, c'est-à-dire lorsque tous les angles sont égaux à 0. La deuxième partie du fichier donne l'ensemble des angles du squelette par rapport à la position neutre pour chaque pas de temps.

La figure 2.1 présente le modèle que nous avons utilisé. La hanche est la racine de l'arbre cinématique. Ses paramètres de translation et de rotation déterminent donc la position et l'orientation du corps dans le repère du monde. Sur la figure, le squelette est dans sa configuration neutre, c'est-à-dire que les angles de toutes articulations sont nuls ; cette pose représente la position de référence par rapport à laquelle sont exprimés les angles d'une pose quelconque. Dans POSER, lorsque la chaîne cinématique est dans sa position initiale, les repères locaux sont alignés avec le repère associé à la racine de la chaîne.

Dans nos expérimentations, deux types de paramétrisation ont été testées.

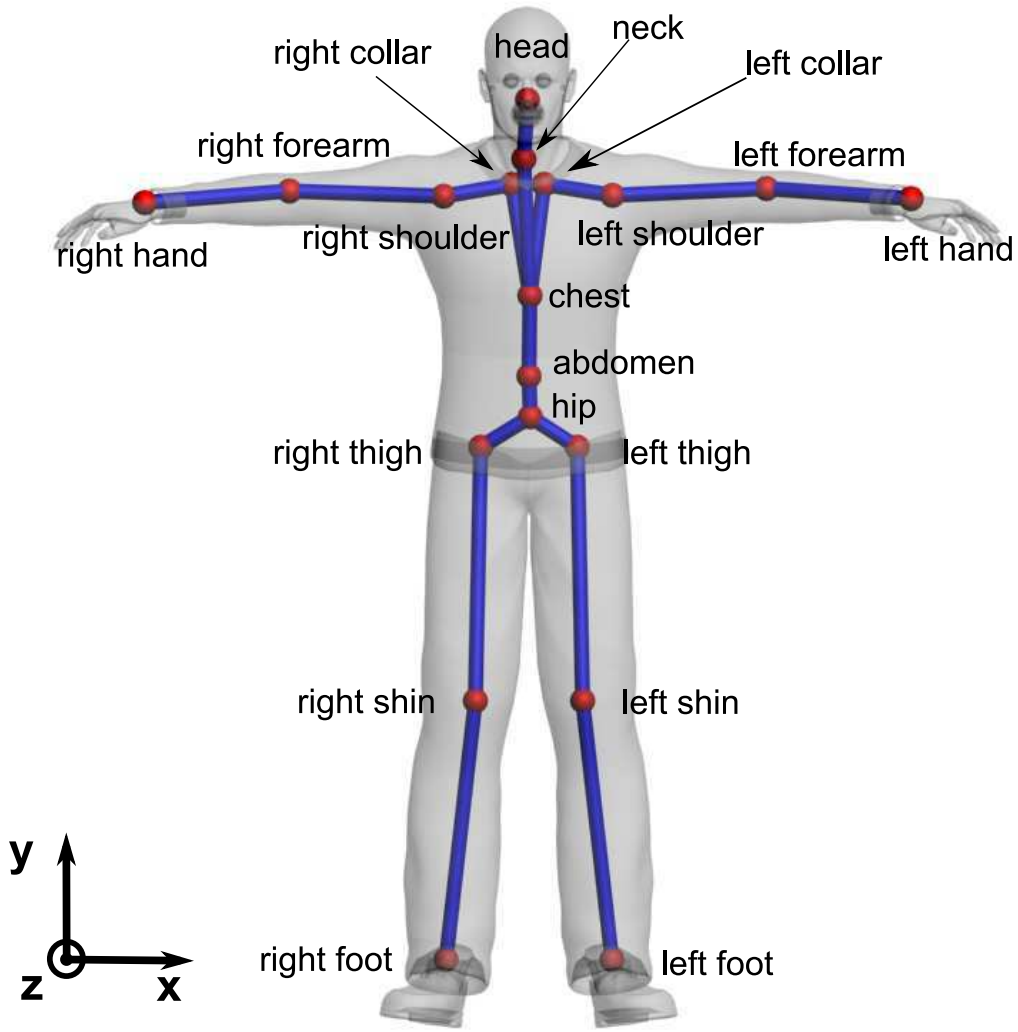


FIG. 2.1 – Modèle du corps humain utilisé dans cette thèse (squelette de l'avatar par défaut du logiciel POSER 6).

Paramétrisation par des angles

Pour les essais sur les angles, les principales articulations du corps humain ont été retenues, et pour chaque articulation, seuls les principaux angles sont utilisés (ceux ayant une amplitude significative). Dans cette paramétrisation, beaucoup de degrés de liberté peuvent être facilement bloqués car ils correspondent en général à l'un des trois axes du repère local. Par exemple, pour

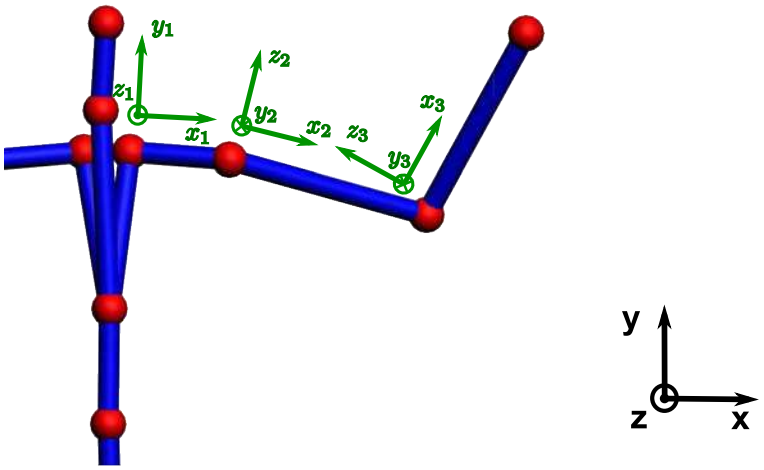


FIG. 2.2 – Chaque articulation définit un repère local par rapport auquel est exprimée la rotation de l'articulation suivante dans la chaîne.

les mouvements du coude ou du genou peuvent être bien approximés avec un seul degré de liberté. Le tableau 2.1 donne le nombre de DDL utilisés pour chaque articulation de la figure 2.1 (les noeuds de la figure 2.1 situés en bout de chaîne n'apparaissent pas dans la paramétrisation par des angles).

Paramétrisation avec les points 3D

Dans les expérimentations, cette paramétrisation a été principalement utilisée dans les essais sur les bras. Pour les données d'apprentissage, les positions des points 3D sont évaluées à partir des paramètres angulaires de la chaine cinématique (l'échelle des différents segments et la position initiale du squelette sont connues). Comme il a été dit précédemment, pour que la représentation soit invariante par changement d'échelle et ne dépende pas des dimensions de l'avatar considéré, le vecteur des positions doit être normalisé. Les positions des points 3D d'un avatar sont donc exprimées en se ramenant toujours au même squelette. Cette normalisation suppose que les proportions entre les parties du squelette sont constantes. Or, le ratio entre les longueurs des segments du corps peut varier légèrement d'un individu à l'autre. C'est une limite de cette paramétrisation.

Comme on l'a vu en 2.2.1, la pose estimée après régression n'est pas

articulation	nombre de DDL	axes de rotation
hanche (hip)	1	Y
jambe gauche (left thigh)	3	X,Y,Z
jambe droite (right thigh)	3	X,Y,Z
genou gauche (left shin)	1	X
genou droit (right shin)	1	X
abdomen	0	-
torse (chest)	0	-
cou (neck)	0	-
clavicule gauche (left collar)	3	X,Y,Z
clavicule droite (right collar)	3	X,Y,Z
épaule gauche (left shoulder)	3	X,Y,Z
épaule droite (right shoulder)	3	X,Y,Z
coude gauche (left forearm)	1	Y
coude droit (right forearm)	1	Y
total	23	

TAB. 2.1 – Nombre de DDL par articulation dans la paramétrisation par des angles (les axes de rotation sont parallèles au repère défini sur la figure 2.1).

forcément une pose valide : comme les trois coordonnées sont estimées indépendamment, rien n'est fait pour que la longueur des membres du corps corresponde aux proportions réelles du squelette. Nous avons donc appliqué une correction après l'estimation pour ramener les longueurs des segments à des valeurs correctes. Pour cela, nous avons modifié les positions 3D des articulations estimées en cherchant une pose valide qui soit la plus proche possible de la pose estimée. Un schéma de la méthode est donné sur la figure 2.3. Par exemple, pour corriger la position du coude, le point est recherché sur la demi-droite partant de l'épaule et passant par le point estimé pour le coude : le coude est ramené au point de la demi-droite situé à une distance de l'épaule égale à la longueur originale du segment. La même correction est ensuite appliquée à la position 3D de la main à partir de la nouvelle position 3D du coude. Cette méthode de correction a été choisie pour sa simplicité et représente bien sûr une solution approximative. En toute rigueur, cette contrainte sur la longueur des segments devrait être intégrée soit directement dans la paramétrisation, soit dans l'estimation (ce qui n'est pas évident dans le cas d'une approche par régression), soit en recherchant a posteriori la position des points qui minimise la distance aux points estimés tout en respectant les contraintes sur la longueur des segments. Nous verrons toutefois dans les

expériences du chapitre 4 que cette correction ne dégrade pas la précision de l'estimation, et aurait même au contraire tendance à rapprocher les points estimés de la vérité terrain.

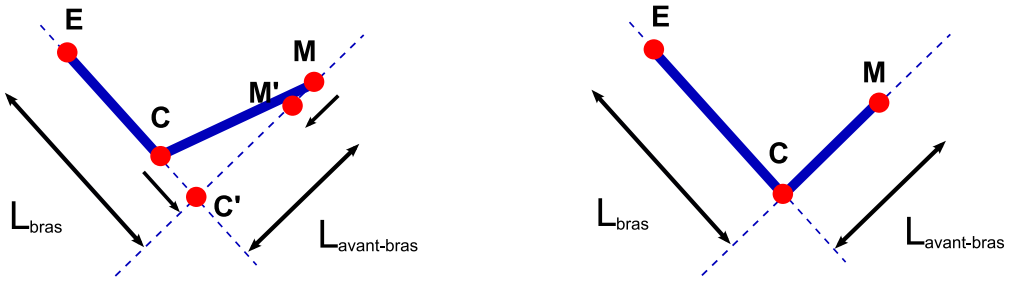


FIG. 2.3 – Correction de la longueur des segments après estimation des points 3D.
a : position du coude (C) et de la main (M) après estimation. Les points C et M sont ramenés respectivement en C' et M'.
b : position du coude et de la main après correction.

2.3 Reconstruction des silhouettes 3D

Cette partie décrit les différentes étapes permettant de reconstruire la silhouette 3D sur laquelle s'appuie notre méthode d'estimation. Le paragraphe 2.3.1 expose la méthode de soustraction de fond qui a été employée pour extraire les silhouettes 2D des images. Nous présentons la notion d'enveloppe visuelle dans le paragraphe 2.3.2, et détaillons l'algorithme de reconstruction utilisé au paragraphe 2.3.3. Des exemples de silhouettes reconstruites sont donnés dans la partie 2.3.4. La partie présente 2.3.5 différentes perspectives d'amélioration de notre technique de reconstruction 3D.

2.3.1 Soustraction de fond

L'algorithme de soustraction de fond utilisé s'appuie sur la minimisation d'une fonction de coût basée sur la différence d'intensité entre une image du fond et l'image courante, ainsi qu'un critère de cohérence spatiale prenant en compte la proximité entre les pixels et la valeur du gradient dans la zone de l'image considérée. Cette énergie est minimisée par *Graph Cuts*. La méthode est exposée plus en détails dans l'annexe A.

La figure 2.4 montre les performances de notre méthode d'extraction de silhouettes. Les contours des personnes sont déterminés de façon satisfaisante : les contraintes de cohérence spatiales permettent d'obtenir une extraction précise et robuste. Il subsiste quelques cas d'ambiguïtés lorsque la couleur du premier plan est similaire à celle du fond sur une large surface. Ces ambiguïtés pourraient être levées en faisant intervenir soit des connaissances a priori sur la scène, notamment des modèles sur l'apparence humaine, soit en analysant le mouvement d'une image à l'autre pour définir de nouvelles contraintes.

FIG. 2.4 – Exemples de silhouettes extraites par soustraction de fond.

2.3.2 Enveloppe visuelle

Le concept d'enveloppe visuelle a été introduit dans [61]. Pour chaque caméra, la silhouette extraite permet de définir un cône généralisé, appelé *cône de vue*, dont le sommet est le centre optique de la caméra, et dont la surface est définie par l'ensemble des rayons partant du centre optique et passant par l'un des points du contour externe de la silhouette. L'enveloppe visuelle se définit comme l'intersection de tous les cônes de vue associés aux différentes caméras. L'enveloppe visuelle d'un objet S est l'approximation la plus proche de S que l'on peut obtenir à partir d'un ensemble de silhouettes. Elle représente le volume maximal qui est équivalent à S pour les silhouettes, c'est-à-dire qui peut être substitué à S sans affecter aucune des silhouettes. Plus les points de vue sont nombreux, plus l'enveloppe visuelle constitue une bonne approximation de l'objet. L'enveloppe visuelle dépend à la fois de l'objet S lui-même et des régions de l'espace couvertes par les différents

points de vue.

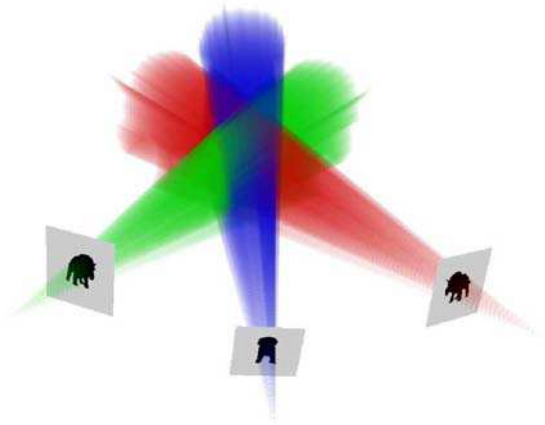


FIG. 2.5 – Intersection des cônes de vue de 3 caméras (figure extraite de [69]).

Une méthode d'estimation de la pose basée sur l'enveloppe visuelle du corps a été retenue d'une part parce qu'elle fusionne toutes les informations dont on dispose sur le système (données image avec les silhouettes 2D, paramètres internes pour le calcul des cônes de vue et géométrie 3D avec le calibrage des caméras), et d'autre part parce qu'elle rend le système d'estimation plus indépendant de la configuration des caméras (voir paragraphe 3.2.2). Le fait de passer par l'enveloppe visuelle permet en outre de visualiser les ambigüités visuelles engendrées par une configuration particulière du système (en considérant chaque vue séparément, ces ambigüités existent mais ne sont pas perçues de façon explicite).

2.3.3 Reconstruction de l'enveloppe visuelle en voxels

En pratique, l'enveloppe visuelle est rarement estimée en calculant l'intersection des cônes de vue car le calcul de l'intersection des rayons 3D est numériquement instable ([25]). A l'exemple de [25], une implémentation en voxels a été choisie. Une alternative, utilisée par les auteurs de [26] pour de la classification de postures, est d'approximer l'enveloppe par un ensemble de surfaces polyédriques (voir [68]).

Partant d'une grille 3D de voxels, la méthode d'estimation choisie consiste à sculpter les voxels, c'est-à-dire à éliminer pour chaque caméra les voxels situés en dehors du cône de vue. Le volume d'intérêt (cube centré sur la per-

sonne) est divisé en une grille de $N \times N \times N$ voxels de tailles égales. Chaque voxel v de la grille est projeté dans les images pour tester son appartenance aux différentes silhouettes extraites.
En notant $Proj^k(v)$ la région obtenue en projetant le voxel v sur l'image de silhouette de la caméra k ($k \in \{1, ..., K\}$), et $Ch^k(A)$ la fonction retournant ***VRAI*** si la région A chevauche la k^e silhouette, et ***FAUX*** sinon, un voxel est classé comme appartenant ou non à la reconstruction ENV par l'algorithme suivant ([25]) :

Pour chaque voxel v,

1. Fixer l'indice k de l'image à 1.
2. **si** $Ch^k(Proj^k(v))$ est ***FAUX***
   alors $v \notin ENV$, **fin**
   sinon si $k = K$
   alors $v \in ENV$, **fin**
   sinon $k = k + 1$, aller en 2.

Avec cette implémentation, si un voxel est projeté en dehors de la silhouette pour une des images, les autres images n'ont pas besoin d'être testées.

La région $Proj^k(v)$ peut être obtenue en projetant les 8 sommets du voxel v, et en calculant l'enveloppe convexe des points projetés. Pour évaluer la fonction $Ch^k(Proj^k(v))$, il faut théoriquement tester l'appartenance de tous les pixels situés à l'intérieur de l'enveloppe convexe à la silhouette de l'image k pour classer le voxel. Comme le parcours exhaustif de tous les pixels de l'enveloppe convexe n'est pas envisageable pour des applications temps-réel, les auteurs de [25] proposent de tester l'appartenance de Q pixels distribués uniformément dans l'enveloppe, et de classer le voxel comme appartenant à la reconstruction 3D si au moins ϵQ ($\epsilon < 1$) des pixels sélectionnés appartiennent à la silhouette (*Sparse Pixel Occupancy Test*, SPOT).

Dans nos expérimentations, une reconstruction simplifiée utilisant seulement la projection du centre d'un voxel a été utilisée (voir figure 2.6). Si la résolution de la grille est suffisamment fine, cette méthode permet d'obtenir une approximation correcte de la reconstruction proposée dans [25].

Extraction des voxels de la surface

Pour certaines applications, il est utile de déterminer quels sont les voxels de la surface de l'enveloppe, par exemple pour gérer l'affichage de la forme

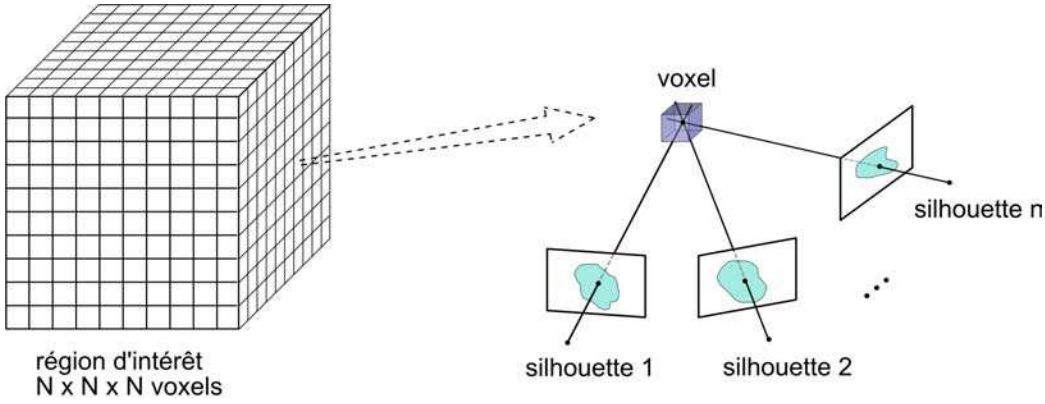


FIG. 2.6 – La reconstruction 3D est estimée en projetant le centre des voxels dans les images de silhouettes.

3D (les voxels internes n'ont pas besoin d'être rendus) ou pour calculer un descripteur basé sur la surface, comme le Shape Context 3D (voir paragraphe 3.2.3). Un voxel est classé comme appartenant à la surface de la reconstruction si un ou plus de ses voxels voisins (pour la 6-connexité) est vacant.

Influence de la résolution de la reconstruction

Dans nos essais, nous avons utilisé comme région d'intérêt un cube de 2 m de côté et une résolution par défaut de 128, ce qui donne des voxels d'environ 1.5 cm de côté. On peut penser que le choix de la taille des voxels joue sur la qualité de l'estimation. Plus la résolution est grande, et plus la forme reconstruite est proche de l'enveloppe réelle, et plus elle a de chances de capturer des détails fins. D'un autre côté, lorsque la silhouette est reconstruite à partir de silhouettes bruitées, il est inutile de chercher à accéder à des détails fins de l'enveloppe et d'essayer de reconstruire une information dont on ne dispose pas réellement. Nous étudierons dans les tests du chapitre 5 l'influence de la résolution de la reconstruction sur la précision de la pose estimée.

2.3.4 Exemples de reconstructions en voxels

L'enveloppe visuelle représente une approximation de la forme qui est observée par les caméras. L'objet reconstruit est inclus dans l'enveloppe visuelle, mais le volume de l'enveloppe visuelle est plus grand que l'enveloppe réelle de l'objet. La reconstruction 3D calculée à partir d'un nombre limité

de silhouettes 2D présente la plupart du temps des artéfacts, c'est-à-dire des morceaux de matière supplémentaires qui ne font pas partie de l'objet réel. La figure 2.7 illustre ce phénomène dans le cas d'un système de 2 caméras. Sur cette figure, on peut en particulier noter la formation de deux volumes "fantômes" déconnectés de l'objet réel. La forme complexe du corps humain et les occultations entre les différentes parties du corps dans les images accentuent ces singularités. Dans certains cas, ces éléments supplémentaires peuvent fortement ressembler à des parties du corps (un bras ou une jambe), si bien que la reconstruction 3D ne permet parfois pas de déterminer visuellement où sont réellement positionnés les membres du corps. Evidemment, plus les points de vue sont nombreux, et plus l'enveloppe visuelle approxime bien la forme réelle du corps, et plus les artéfacts sont limités.

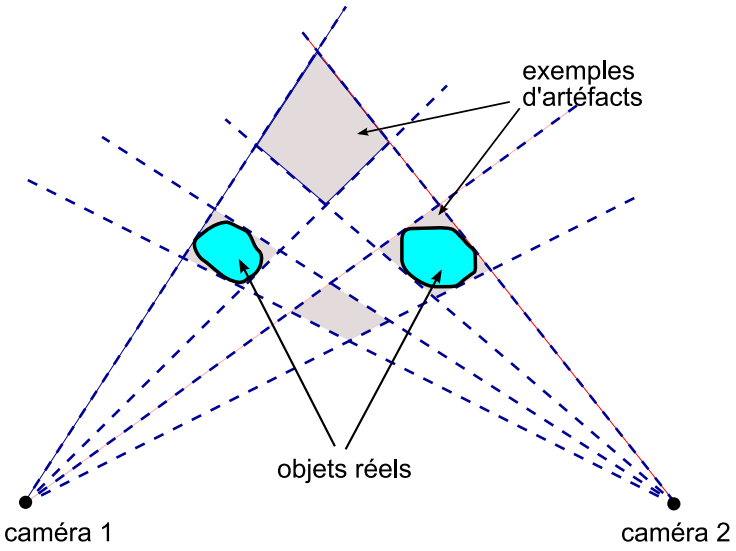


FIG. 2.7 – Artéfacts créés avec un système de 2 caméras.

Exemples de reconstructions sur données de synthèse

Dans ce paragraphe, les enveloppes visuelles sont reconstruites à partir des images binaires de silhouettes "parfaites" d'un avatar, générées par un logiciel de synthèse. La figure 2.8 montre des exemples de la même posture reconstruite avec deux systèmes de caméras : l'un contenant seulement 3 caméras, et l'autre 13 caméras. La première reconstruction présente de gros artéfacts. Dans le second cas, on peut penser que la silhouette reconstruite doit être assez proche de la silhouette idéale.

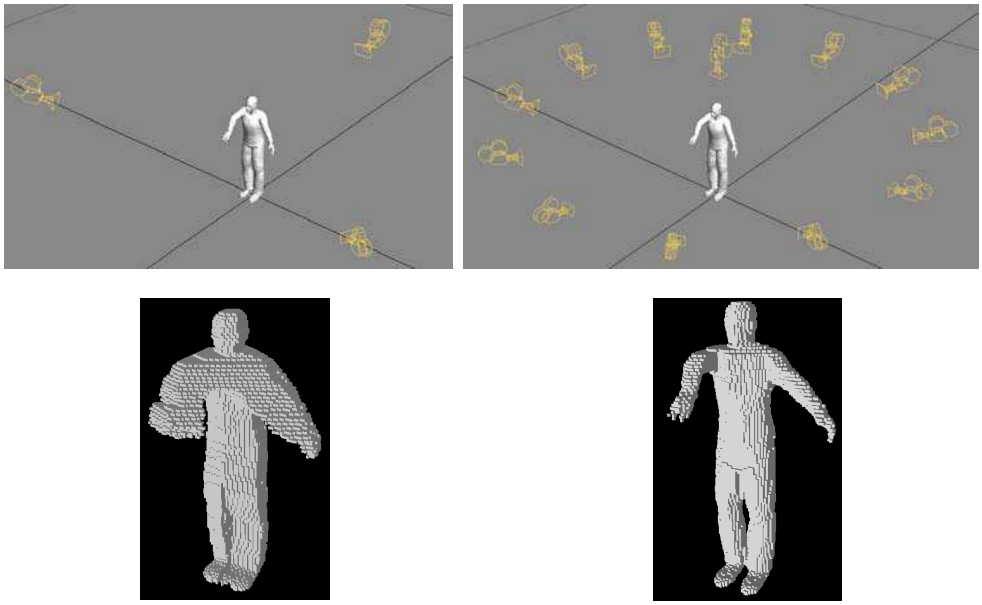


FIG. 2.8 – Exemples de silhouettes 3D reconstruites à partir de données de synthèse.

La figure 2.9 donne un exemple d'une silhouette 3D présentant des artéfacts importants au niveau des jambes. Sur la figure 2.10, la même posture avec une orientation différente par rapport au système de caméras n'a pas engendré les mêmes artéfacts.

Exemples de reconstructions obtenues à partir d'images réelles

Les silhouettes 2D extraites des images réelles par soustraction de fond présentent souvent des défauts liés à des ambiguïtés de couleur ou d'intensité entre le fond et la forme que l'on cherche à extraire. Ces erreurs de segmentation ont des implications sur la forme 3D reconstruite. Quelques exemples sont donnés dans cette section.

Un problème fréquent avec les méthodes de soustraction de fond est qu'elles ne parviennent souvent pas bien à éliminer les ombres au sol de la silhouette. Ce défaut est parfois atténué dans la reconstruction 3D en combinant les silhouettes des différentes caméras, mais bien souvent, si l'ombre n'a pu être éliminée de la silhouette pour aucune des caméras, la reconstruction 3D contiendra également ce type de bruit. La figure 2.11 en donne un exemple.

La figure 2.12 montre un autre type de difficulté sur une image de la base

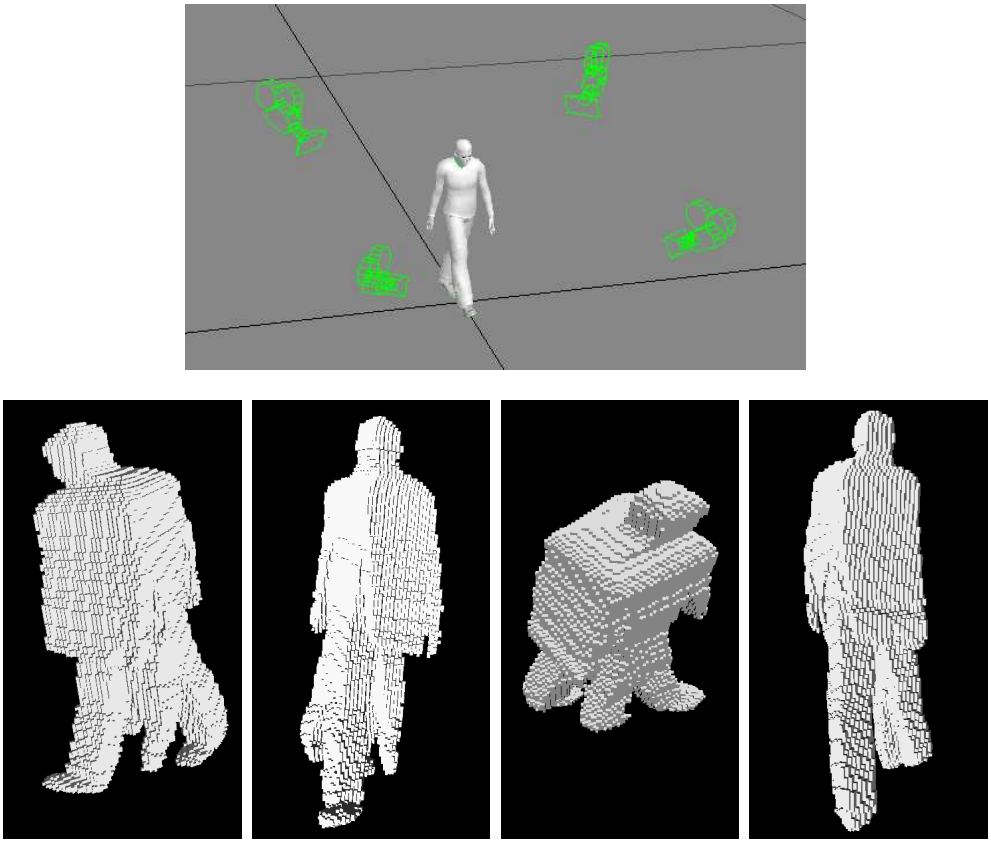


FIG. 2.9 – Artéfacts générés sur la reconstruction 3D d'une posture de marche.

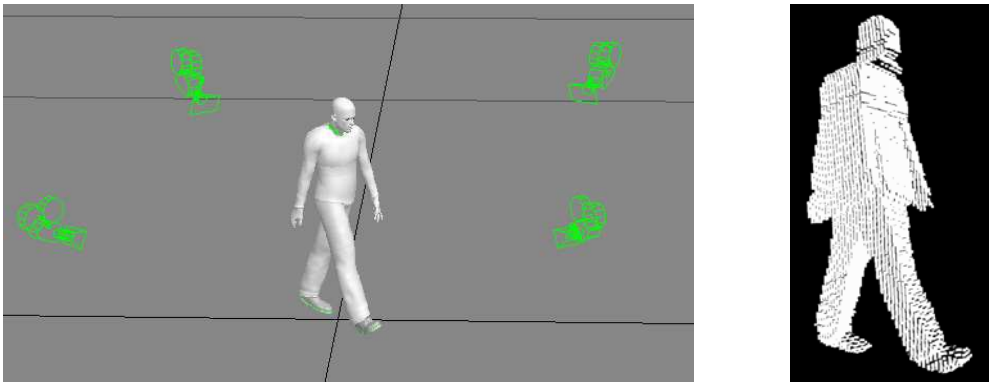


FIG. 2.10 – Reconstruction 3D de la même posture orientée différemment par rapport aux caméras.

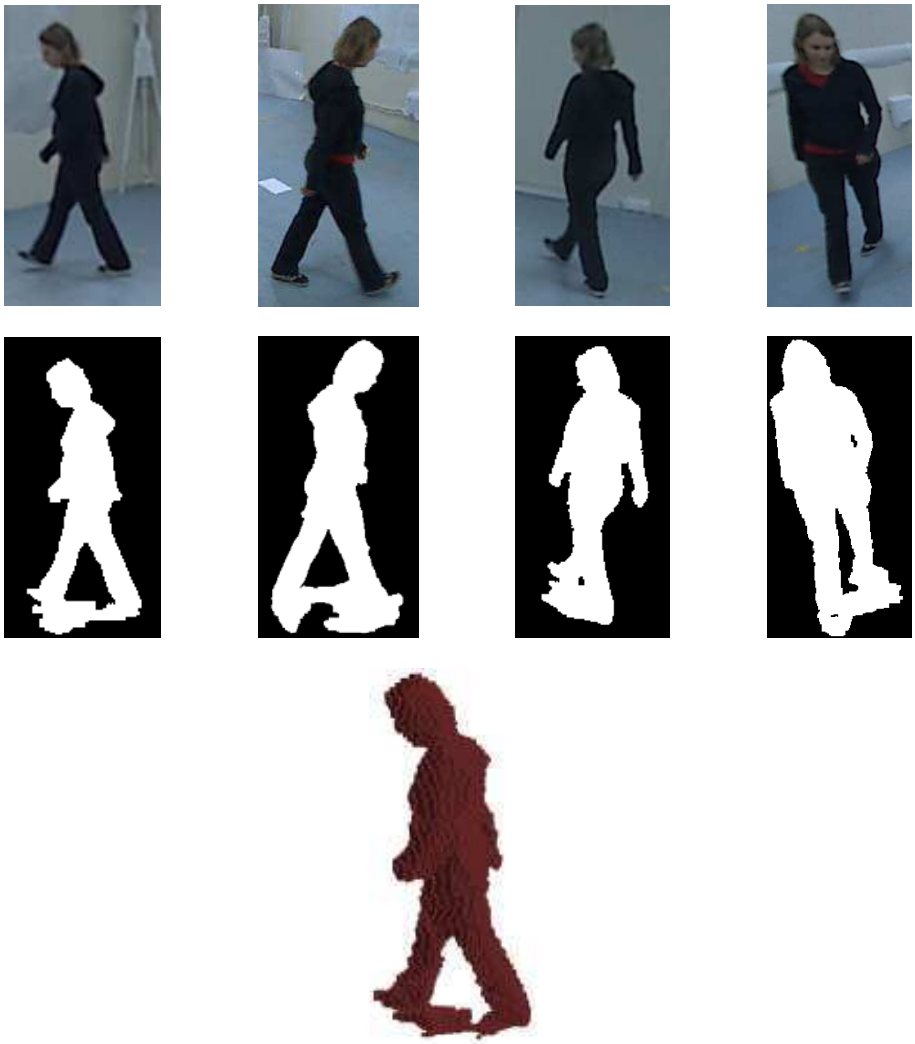


FIG. 2.11 – Effet des ombres au sol sur la reconstruction 3D.

HumanEva [101]. La silhouette 3D est reconstruite à partir des 4 caméras noir et blanc. Une différence d'intensité trop faible entre les vêtements et l'arrière plan a engendré après soustraction de fond une silhouette 2D très détériorée pour l'une des caméras. Notre algorithme de reconstruction ne permet pas de bien gérer ce type de situation : si la silhouette 2D d'une seule des caméras est trouée, toutes ces erreurs de segmentation se retrouvent dans la silhouette 3D.

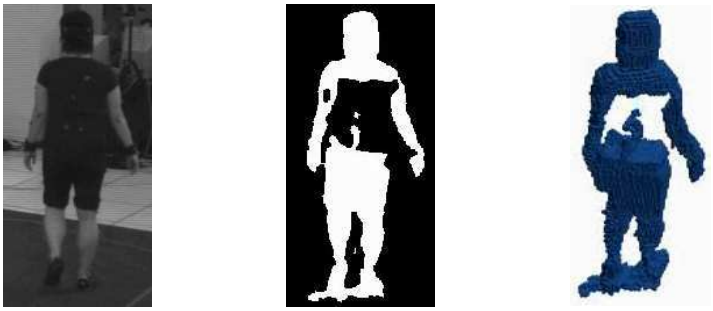


FIG. 2.12 – Influence des erreurs de segmentation sur la silhouette 3D.

2.3.5 Améliorations possibles de la reconstruction

Plusieurs améliorations pourraient être apportées à l'algorithme de reconstruction 3D que nous utilisons. Le processus de reconstruction pourrait tout d'abord être optimisé afin de réduire son temps de calcul. Dans [25], la personne dont on reconstruit les mouvements est supposée rester toujours à l'intérieur d'un volume d'intérêt prédéfini. La projection des voxels de cette zone sur chacune de images est calculée à l'avance et stockée dans une table de correspondance. Le processus de reconstruction est ensuite nettement accéléré car l'utilisation de cette table de correspondance supprime la plupart des opérations nécessaires au calcul de l'enveloppe. Dans notre cas, la construction d'une table de correspondance est plus délicate puisque la personne se déplace et ne reste pas dans un volume fixe : il faudrait donc construire une table sur tout l'espace de capture. Une méthode de reconstruction à plusieurs résolutions, par exemple basée sur les *Octrees*, pourrait aussi être utilisée [107] : le volume d'intérêt, représenté par un cube, est progressivement divisé en 8 sous-cubes. Une fois qu'il a été déterminé que l'un des cubes est entièrement à l'intérieur (ou à l'extérieur) de la silhouette 3D, les subdivisions s'arrêtent. Cette méthode permet d'éviter une analyse exhaustive de l'appartenance de tous les voxels à l'enveloppe à la résolution la plus fine.

Une première possibilité pour gagner en robustesse pourrait être d'utiliser un algorithme de soustraction de fond multi-vues. Dans notre approche, des silhouettes 2D binaires sont extraites séparément dans chaque image, et les informations du système ne sont fusionnées qu'après la soustraction de fond. Or, dans un système de caméras calibrées qui observent simultanément le même objet, des informations supplémentaires sur la cohérence spatiale entre les différentes images pourraient être prises en compte. Par exemple,

dans [64], les auteurs proposent une méthode d'extraction du fond exploitant seulement les contraintes de cohérence spatiale et de couleur que plusieurs images d'une même région doivent satisfaire. Dans ces travaux, aucune hypothèse n'est faite sur une connaissance a priori d'un modèle du fond. On peut aussi citer [60], dans lequel les auteurs proposent de fusionner les masques de chaque image en une vue virtuelle d'en haut qui permet de filtrer les variations d'illumination ou les ombres apparaissant sur le sol.

Une autre perspective envisageable est d'utiliser une grille probabiliste comme celle présentée dans [35] au lieu d'une reconstruction binaire (voxel allumé ou éteint). Dans ces travaux, les auteurs exploitent toutes les sources d'incertitudes du système d'acquisition : des cartes de probabilité de présence calculées dans chaque image sont fusionnées pour reconstruire une grille 3D contenant en chaque voxel une probabilité d'occupation. Dans [44] les auteurs proposent une extension de la méthode pour gérer les occultations et reconstruire les objets occultants dans la scène.

Une autre possibilité proposée dans [41] est de modéliser par apprentissage (avec une base de données POSER) un a priori sur la forme que doit avoir l'enveloppe visuelle. Cette méthode permet de débruiter les silhouettes 2D extraites et l'enveloppe visuelle. La méthode d'extraction de silhouettes perd toutefois de sa généralité car elle ne fonctionne que sur des silhouettes dans les mêmes postures que celles qui ont été apprises. Dans [42], la méthode est étendue pour estimer une pose 3D conjointement à l'enveloppe visuelle.

2.4 Construction des bases d'apprentissage

2.4.1 Animation des avatars

Pour modéliser par apprentissage l'application permettant de passer d'un descripteur à une pose (chapitre 4), il faut construire une base d'exemples contenant d'un côté les images d'une personne (c'est-à-dire l'ensemble des images acquises simultanément par les caméras du système) et de l'autre la vérité terrain sur sa pose. L'efficacité de la méthode d'estimation est conditionnée par la qualité de la base d'apprentissage. En effet, même si l'algorithme d'apprentissage a de bonnes capacités de généralisation, il est nécessaire de bien couvrir dans la base l'espace des mouvements qu'on souhaite reconnaître, et de l'échantillonner de façon suffisamment dense. Pour réaliser des bases de données réalistes (à la fois pour ce qui est de la pose, de la morphologie, de l'habillement et du bruit dans les images...), l'idéal serait

de les construire à partir de données réelles. Ce type de données est très difficile à obtenir : il nécessiterait d'acquérir les données simultanément avec les caméras et un système de capture de mouvement, en effectuant un ensemble de mouvements suffisamment variés pour constituer une bonne base d'exemples. Certaines universités ont mis à disposition des bases de données contenant à la fois des images et des résultats de capture de mouvement (CMU Graphics Lab Motion Capture Database [1] et HumanEva [101]). Cependant, l'utilisation de ces bases pour l'apprentissage nous restreint sur la gamme de mouvements disponibles, les points de vue des caméras, la qualité des images... Nous avons choisi d'utiliser le logiciel POSER 6, qui permet de synthétiser et d'animer des avatars en 3D de façon réaliste.

Dans POSER, les avatars ont été animés par l'intermédiaire de fichiers au format BVH, dans lesquels les angles des articulations peuvent être fixés par l'utilisateur. Réciproquement, les données angulaires des articulations peuvent être récupérées à partir d'un fichier. Parmi les configurations qui peuvent être obtenues en jouant sur les différents degrés de liberté du corps humain, seule une petite partie correspond à des mouvements réalistes, c'est-à-dire qui peuvent être effectués de façon plausible par un humain. Pour réaliser des bases d'exemples pour les mouvements de marche, des données issues de séquences de Capture de Mouvement disponibles sur internet (comme [4] et [3]) ont été employées. Utiliser de telles données permet d'obtenir des mouvements réalistes, typiques de ceux réalisés par un être humain. Pour la reconnaissance de geste, la même technique n'est pas facilement transposable, puisque la gamme des mouvements qui peuvent être réalisés avec les bras est très étendue. Il est très difficile de couvrir un espace assez large pour reconnaitre des gestes quelconques. Nous avons donc choisi de définir des intervalles réalistes pour les différentes articulations des bras, et de réaliser les bases d'apprentissage en sélectionnant aléatoirement des angles dans ces intervalles.

Pour tester et améliorer la robustesse de la méthode vis-à-vis de la variabilité de morphologie et d'habillement entre les personnes, 8 avatars différents ont été utilisés pour générer les bases d'apprentissage (voir figure 2.13).

2.4.2 Construction des silhouettes 3D

Les avatars animés dans POSER sont ensuite exportés dans 3DStudioMax pour effectuer le rendu des silhouettes 2D. Ce logiciel permet de manipuler facilement des caméras, de les positionner et d'effectuer des rendus avec les points de vue désirés. L'avatar peut aussi être placé à un endroit précis de la

(a) 1 (b) 2 (c) 3 (d) 4 (e) 5 (f) 6 (g) 7 (h) 8

FIG. 2.13 – Les 8 avatars utilisés pour générer des bases de données synthétiques.

scène. A partir d'un système de caméras réelles, un système identique peut facilement être reproduit dans le logiciel pour synthétiser une base d'apprentissage avec la même configuration 3D. Les silhouettes obtenues (voir figure 2.14) sont des silhouettes parfaites : elles ne présentent pas les imperfections qu'on peut trouver sur les silhouettes extraites des images réelles avec un algorithme de soustraction de fond (trous, ombres...). Pour chaque exemple, la silhouette 3D est reconstruite, puis le descripteur est calculé. On dispose ainsi d'une base d'exemples constituée d'un ensemble de paires descripteur/pose.

2.5 Conclusion

Nous avons présenté dans ce chapitre différents outils sur lesquels s'appuie notre système d'estimation. Chacun de ces éléments a une importance sur les performances de la méthode. L'extraction des silhouettes 2D dans les images réelles est un point critique dans notre cas, car l'apprentissage est réalisé à partir de silhouettes parfaites obtenues par des logiciels de synthèse, et l'estimation par régression impose que les silhouettes réelles ne soient pas trop éloignées de celles qui composent la base d'entrainement. L'étape de reconstruction 3D, qui fusionne les silhouettes 2D, a des implications à la fois sur le temps de calcul du système et la précision de la pose qui peut être atteinte par l'estimation. La paramétrisation de la pose peut aussi être choisie de manière plus ou moins judicieuse en fonction du type de mouvement que l'on cherche à reconnaître. Tous ces idées seront mises en évidence de manière quantitative dans les expériences sur données de synthèse des chapitres 4 et 5. Les expériences du chapitre 5 montreront aussi à quel point la qualité de la base d'apprentissage influence les performances du système sur les images réelles.

Notre méthode d'estimation est basée sur la mise en relation par appren-

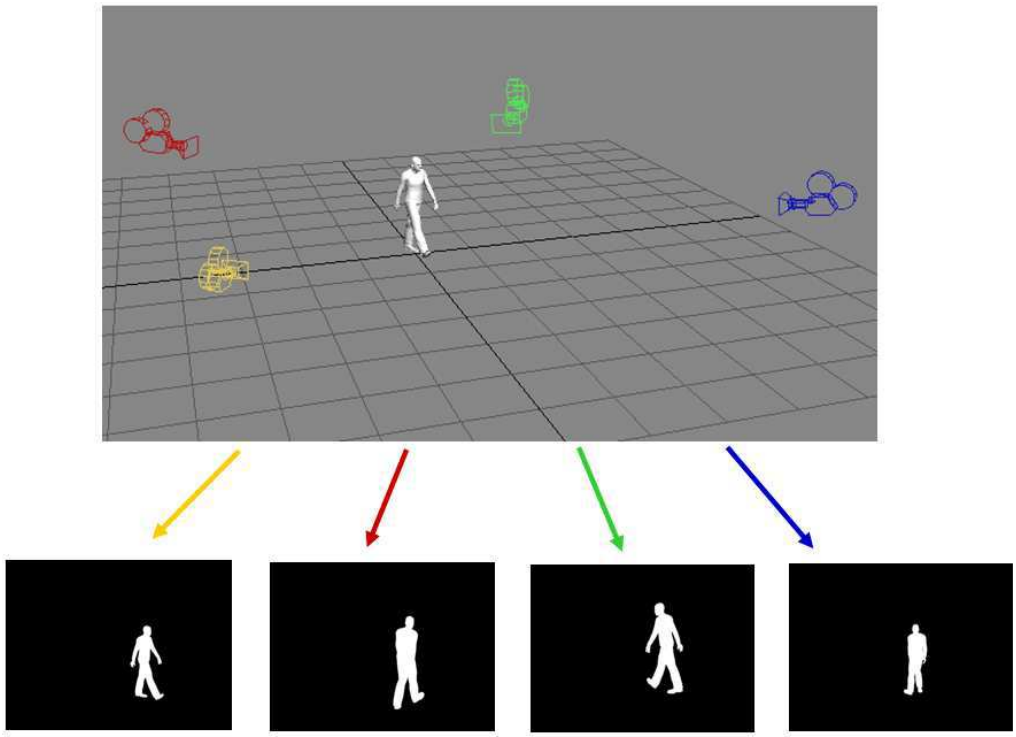


FIG. 2.14 – Rendu des silhouettes d'un avatar avec 3DSMax.

tissage de deux vecteurs, l'un décrivant les données image via une reconstruction 3D, et l'autre la pose. Nous avons vu dans cette partie qu'il existe différentes façons de paramétrer la pose du corps. Le chapitre suivant présente de quelle manière la géométrie de la silhouette 3D est encodée dans le vecteur descripteur.

Chapitre 3

Descripteur

3.1 Introduction

Pour effectuer la régression, il est nécessaire de construire une représentation plus compacte que les données brutes de la reconstruction en voxels. Le rôle du descripteur est d'encoder les caractéristiques de la forme 3D dans un vecteur, qui sera ensuite fourni en entrée de la machine de régression, afin d'estimer la pose 3D.

Dans les méthodes d'estimation basées sur un apprentissage, le choix du descripteur est un élément crucial. Ce choix doit tenir compte de plusieurs critères importants. Tout d'abord, la plupart des algorithmes d'apprentissage sont plus performants sur des données d'entrée de petite dimension. Il faut donc trouver un bon compromis entre la quantité d'information qui peut être contenue dans le vecteur, et sa dimension. Par ailleurs, le descripteur doit être à la fois suffisamment discriminant pour permettre de capturer des variations subtiles de la posture, tout en restant robuste au bruit et aux erreurs de segmentation des silhouettes. Pour obtenir cette robustesse au bruit, une propriété importante du descripteur est sa localité. Dans une représentation *globale*, le calcul de chacune des composantes s'appuie sur l'ensemble des pixels de la région à décrire, tandis que dans un descripteur *local*, les différentes composantes utilisent seulement une partie de l'image. Une description globale risque à la fois de ne pas bien rendre compte des petites variations de la pose, et d'être trop sensible au bruit, puisqu'une erreur de segmentation sur une partie de la silhouette (trou, occultation, ombre...) a des répercussions sur toutes les composantes du descripteur. A l'inverse, une représentation trop locale risque elle aussi d'être sensible au bruit dans la mesure où elle capture toutes les petites variations de la silhouette, y compris

les informations inutiles pour estimer la pose. Dans notre cas, le descripteur doit être invariant en translation (le sujet peut se déplacer dans l'espace de capture), et aux changements d'échelle. En revanche, le descripteur ne doit pas être invariant en rotation puisqu'on souhaite estimer l'orientation globale du corps dans l'espace par régression. Une difficulté importante dans le cas du corps humain est la grande variabilité des formes obtenues pour une posture donnée. Deux personnes ayant la même pose peuvent en effet générer des observations très différentes en raison des variations de corpulence et de morphologie, ou d'habillement. Le descripteur doit, dans la mesure du possible, tenir compte de cette variabilité entre les personnes et rester invariant aux différences de morphologie. La représentation doit aussi être continue par rapport aux variations de la pose qu'elle encode : deux poses voisines doivent correspondre à des descripteurs proches (pas de saut dans la représentation). Enfin, dans le cas où l'objectif est d'estimer la pose en temps-réel, le descripteur doit être rapide à calculer, il est donc important de bien contrôler la complexité des opérations nécessaires à son estimation.

Ce chapitre est composé de trois parties. La première partie dresse un rapide état de l'art des descripteurs, 2D ou 3D, qui ont été proposés dans la littérature pour estimer la posture d'une personne à partir d'une ou plusieurs silhouettes. La deuxième partie détaille le descripteur développé dans cette thèse et les différents paramètres dont il dépend. Le troisième paragraphe présente une étude expérimentale visant à évaluer l'influence de ces paramètres sur la qualité de la description.

3.2 Etat de l'art sur les descripteurs

3.2.1 Descripteurs 2D

On s'intéresse dans cette partie aux descripteurs d'une silhouette binaire 2D qui peuvent être exploités pour estimer la posture d'une personne. Comme il a été vu dans l'état de l'art, la silhouette est une primitive très fréquemment utilisée pour ce type de problème. La variété des descripteurs proposés dans la littérature est extrêmement large.

Moments 2D

Les moments géométriques d'une forme sont des descripteurs globaux : leur calcul se base sur l'ensemble des pixels de la région d'intérêt. Ils sont

donc généralement peu robustes aux erreurs de segmentation de la silhouette. Ces moments décrivent la répartition des valeurs d'une image (ou d'une distribution) f par rapport à ses axes x et y.
Le moment $M_{p,q}$, d'ordre $p+q$, s'écrit :

$$M_{p,q} = \iint_D x^p y^q f(x,y)\,\mathrm{d}x\,\mathrm{d}y \tag{3.1}$$

Dans le cas d'une silhouette, la distribution f vaut 1 à l'intérieur de la silhouette et 0 ailleurs. A partir de cette définition, on peut construire des moments invariants aux translations et aux changements d'échelle. Comme les données étudiées dans des séquences vidéo sont bruitées, on ne considère généralement en traitement d'images que les moments d'ordre inférieur ou égal à 3. A partir des moments géométriques, 7 moments invariants en translation, échelle et rotation ont été dérivés : les moments de Hu. Les moments de Hu sont par exemple utilisés dans [93, 94, 95] pour de l'estimation de pose.

De nombreux autres moments peuvent être définis en remplaçant le polynôme $x^p y^q$ par une fonction quelconque de $\mathbf{R}^2$, comme par exemple les moments de Zernike [110], les moments basés sur des ondelettes [99], la transformée en cosinus discrète [118]. Le choix de cette fonction détermine les caractéristiques finales des moments : dans le cas des moments de Zernike, une fonction du type $V_{mn}(r,\theta) = R_{mn}(r)e^{-jn\theta}$ engendre des moments dont la magnitude est invariante par rotation, tandis que le choix d'une fonction à support localisé (comme des ondelettes) permet d'ajouter des propriétés de localité aux moments...

Shape Context

Le *Shape Context* (introduit dans [11]) est une description locale non-paramétrique de la forme d'un objet, s'appuyant sur des points échantillonnés le long de ses contours (internes et externes). Pour chaque point, un histogramme décrit la répartition spatiale des points voisins du contour. L'histogramme est calculé à partir d'une grille découpant l'espace en secteurs angulaires et radiaux (voir figure 3.1(c)). Chaque baquet de l'histogramme représente le nombre de points de contour contenus dans une des cases de la grille. Le Shape Context est paramétré par le nombre r de secteurs radiaux (dans une échelle logarithmique), et le nombre ϕ de secteurs angulaires, ainsi que par la dimension des deux rayons extrêmes r_{inner} et r_{outer}.

Un descripteur s'appuyant sur le Shape Context est utilisé dans [7] pour décrire les contours externes de la silhouette (voir figure 3.1). Chaque point

échantillonné le long du contour est symbolisé par un point de l'espace des Shape Context à 60-D (5 graduations radiales × 12 graduations angulaires). Une silhouette est alors représentée par une distribution de n points dans cet espace (figure 3.1(d)). Pour réduire la dimension de la représentation, un deuxième histogramme est calculé. Un algorithme des K-means (avec $K = 100$) est appliqué à l'ensemble des vecteurs 60-D constitué de tous les points de toutes le silhouettes de la base d'apprentissage. La répartition de ces points dans l'espace est discrétisée en un ensemble de 100 groupes. Chaque point de contour d'une silhouette vote ensuite pour le groupe dont le centre est le plus proche dans l'espace des Shape Context (figure 3.1(e)). Au final, une silhouette est décrite par un histogramme de dimension 100 qui regroupe l'ensemble des votes de ses points de contour. Les histogrammes doivent être normalisés par rapport au nombre de points du contours afin de rendre la représentation invariante aux changements d'échelle. Pour adoucir les effets de la discrétisation spatiale, un point de l'espace des Shape Context peut voter pour plusieurs groupes (ceux dont les centres sont les plus proches) avec des poids différents.

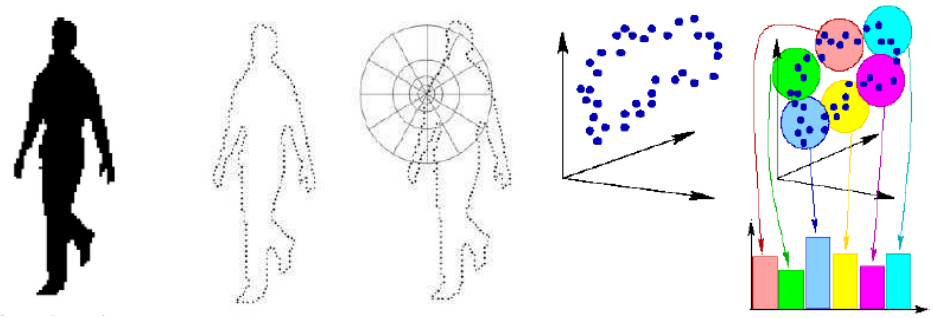


FIG. 3.1 – Descripteur basé sur le Shape Context utilisé dans [7]
a : silhouette extraite. **b :** points échantillonnés le long du contour externe. **c :** diagrammes de Shape Context d'un des point du contour. **d :** ensemble des vecteurs de Shape Context d'une silhouette. **e :** descripteur calculé à partir des secteurs calculés par l'algorithme des K-means.

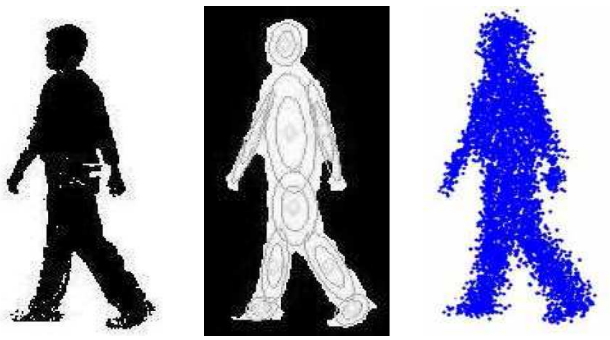
D'autres descriptions basées sur les contours externes de la silhouette sont également possibles, notamment celles qui traitent le contour comme une courbe paramétrée continue, comme par exemple les représentations basées sur les coefficients de Fourier ([88]) ou d'ondelettes. Cependant, comme le soulignent les auteurs de [7], représenter le contour externe de la silhouette

par une courbe paramétrée peut engendrer des discontinuités, puisque les contours de la silhouette changent fréquemment de topologie pour des postures voisines (par exemple si la main touche le torse ou s'en détache).

Mixture de gaussiennes 2D

Cette description est basée sur l'idée que la silhouette peut être vue comme un nuage de points qui a été produit en échantillonnant une distribution de probabilité. Les auteurs de [45, 46] modélisent cette distribution par une mixture de gaussiennes. Les descripteurs de deux silhouettes sont ensuite comparés avec la divergence de Kullback-Leibler. Cette description est utilisée dans [45] pour de la reconnaissance de poses (avec un classifieur RVM) et dans [46] pour de l'estimation de pose 3D par régression (*Bayesian Mixture of Experts*).

L'avantage de ce descripteur est qu'il constitue une représentation locale de la forme et que le fait de représenter les points de la silhouette par une distribution permet de lisser les imperfections de la silhouette provenant de bruit ou d'erreurs de segmentation. L'inconvénient est que le nombre de gaussiennes est fixe (la complexité du modèle ne s'adapte pas à la forme qu'il doit représenter), et que la comparaison entre deux silhouettes doit se faire en échantillonnant la distribution obtenue (puisqu'il n'existe pas de formule analytique de la divergence de Kullback-Leibler dans le cas de mixtures de gaussiennes).



(a) (b) (c)

FIG. 3.2 – Descripteur basé sur les mixtures de gaussiennes ([45]). **a :** silhouette extraite. **b :** moyennes et contour des covariances des gaussiennes de la mixture. **c :** points rééchantillonnés à partir de la mixture de gaussiennes estimée.

Comparaison des descripteurs

On peut trouver dans [88] et [118] des comparaisons sur les performances de quelques uns des descripteurs présentés dans cette partie, pour estimer la pose d'humains à partir de silhouettes binaires (moments de Hu, coefficients de Fourier et Shape Context pour [88] et Transformée en Cosinus discrète, *Lipschitz Embedding* et Shape Context pour [118]). Les auteurs de [88] mettent en évidence la supériorité des histogrammes de Shape Context et des coefficients de Fourier par rapport aux moments de Hu. En revanche, les auteurs de [118] remettent en cause plusieurs des atouts de ce descripteur mis en avant par Agarwal et Triggs dans [7]. Ils ont noté que :

- dans la plupart des cas, la présence de bruits (des trous ou des artéfacts) sur la silhouette 2D modifie le nombre de points de contours de la silhouette, et la normalisation de l'histogramme par rapport au nombre de points de contour implique que toutes ses composantes seront affectées, ce qui réduit les propriétés de localité du descripteur,
- l'utilisation d'une distance euclidienne ne permet pas d'exploiter avantageusement la localité du descripteur,
- dans cette description, aucune différence n'est faite entre l'intérieur et l'extérieur de la silhouette, une partie de l'information utile doit donc être perdue.

Les expériences menées dans ces travaux ont en outre montré la faiblesse de ce descripteur par rapport à d'autres méthodes de description moins élaborées (Transformée en Cosinus Discrète), plus particulièrement dans le cas de silhouettes bruitées extraites d'images réelles.

3.2.2 Cas multi-vues

Dans le cas où l'estimation de la pose est effectuée à partir d'un système de plusieurs caméras, la description doit prendre en compte les mesures provenant des différentes images. Plusieurs façons de combiner ces informations peuvent être proposées.

Une première solution peut être de concaténer les descripteurs obtenus sur les différentes vues pour former un plus grand vecteur réunissant la totalité des informations. L'objectif des travaux présentés dans [29] est d'estimer la pose 3D de la main à partir de plusieurs vues. Les auteurs ont choisi d'utiliser un descripteur basé sur le Shape Context semblable à celui de [7]. Pour combiner les informations des différentes vues, des descripteurs sont calculés individuellement pour chaque caméra, puis concaténés pour former

un nouveau vecteur de plus grande dimension qui résume l'ensemble des mesures obtenues avec toutes les caméras. La régression est ensuite effectuée sur ce vecteur. Dans [42], une silhouette est décrite par un ensemble de points échantillonnés le long du contour externe. De façon similaire à [29], pour décrire l'ensemble des silhouettes des différentes caméras, les auteurs juxtaposent dans un même vecteur la totalité des points de contours des différentes silhouettes. Outre le fait que la dimension du vecteur descripteur augmente avec le nombre de vues, l'inconvénient principal de cette méthode est qu'elle rend le système d'estimation dépendant du positionnement des caméras. La configuration des caméras doit être la même à l'apprentissage et à l'estimation, car l'apprentissage s'est fait avec un ensemble de points de vue fixés. La phase d'apprentissage doit donc être relancée à chaque fois que l'on souhaite estimer la pose à partir d'une nouvelle configuration de caméras.

Lorsque le système de caméras est calibré, une autre façon de fusionner les informations des caméras est de calculer par un algorithme de *Shape from Silhouettes* la reconstruction 3D qui peut être obtenue à partir des silhouettes 2D. Comme indiqué dans [106], si on dispose d'un nombre suffisant de points de vue, l'enveloppe reconstruite est indépendante de la position du système de caméras. Même si la position des caméras a changé, l'estimation peut être faite sur la forme 3D reconstruite sans avoir besoin de relancer un apprentissage. Le nombre de points de vue n'a même pas nécessairement besoin d'être le même entre l'apprentissage et l'estimation. De plus, cette approche permet de fusionner en un seul élément toutes les informations du système, à la fois sur la forme des silhouettes dans les images et sur les différents points de vue. Lorsque cette solution est retenue, un unique descripteur résume toutes les caractéristiques de l'ensemble des images ; sa dimension reste la même quel que soit le nombre de points de vue dont on dispose. Le problème de la description des silhouettes de ramène alors à la construction d'un descripteur 3D. Quelques uns des descripteurs 3D proposés dans la littérature sont présentés dans le paragraphe suivant.

3.2.3 Descripteurs 3D

Moments 3D

L'équivalent en 3D de la formule 3.1 s'écrit (moment d'ordre p+q+r) :

$$M_{p,q,r} = \iiint_D x^p y^q z^r f(x,y,z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z \tag{3.2}$$

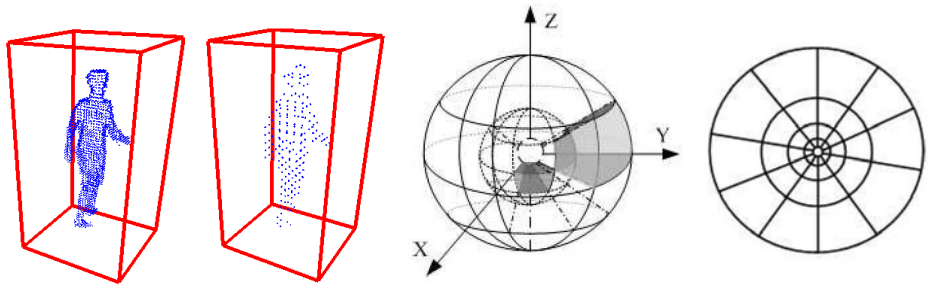
De la même façon qu'en 2D, des moments invariants par translation, rotation et changement d'échelle peuvent être dérivés (voir [66]). De nombreuses variantes de ces moments ont été définis ; on trouve par exemple une extension en 3D des moments de Zernike [23]. Des moments 3D basées sur des ondelettes sont aussi proposés dans [121] ; ces moments sont utilisés pour effectuer une classification de la posture d'humains à partir des nuages de points reconstruits par un scanner 3D. Leurs performances sont comparées à celles des moments de Zernike 3D et des coefficients de Fourier 3D.

Shape Context 3D

Le Shape Context 3D (3DSC) a été introduit dans [57], et utilisé par les auteurs de [106] pour estimer la pose 3D à partir d'une reconstruction du corps en voxels. Cette description est une généralisation en 3D du Shape Context 2D (paragraphe 3.2.1). Pour chaque point échantillonné sur la surface (figure 3.3(a)), un histogramme décrit la répartition dans un volume sphérique des autres points de la surface (figure 3.3(c)). De manière similaire au cas 2D, le Shape Context 3D contient un certain nombre de divisions suivant le rayon (régulièrement espacées suivant une échelle logarithmique), l'élévation et l'azimuth. Un histogramme local est construit en comptant le nombre de point de surface présents dans une des divisions du volume sphérique. Le repère local du volume de support du 3DSC peut être soit aligné à la normale à la surface (ce qui rend la description invariante par rotation), soit aligné avec le repère du monde 3D : le descripteur conserve alors l'information qui permettra d'estimer l'orientation globale de la silhouette dans l'espace ([106]).
D'une manière semblable à [7], une seconde transformation est appliquée par les auteurs de [106] à l'ensemble des histogrammes de Shape Context des points d'une surface pour réduire la dimension des données d'entrée dans la régression. La répartition des points de Shape Context de la base d'apprentissage est modélisée par une mixture d'ACP probabilistes [115] (qui remplace le partitionnement par K-means de [7]). Chaque composante du descripteur d'une silhouette 3D représente la contribution totale des points de 3DSC de la surface de la reconstruction à l'une des n composantes de la mixture.

Descripteur Cohen/Li

Les auteurs de [26] présentent également une façon de décrire la forme 3D reconstruite à partir de silhouettes extraites avec plusieurs caméras. Pour



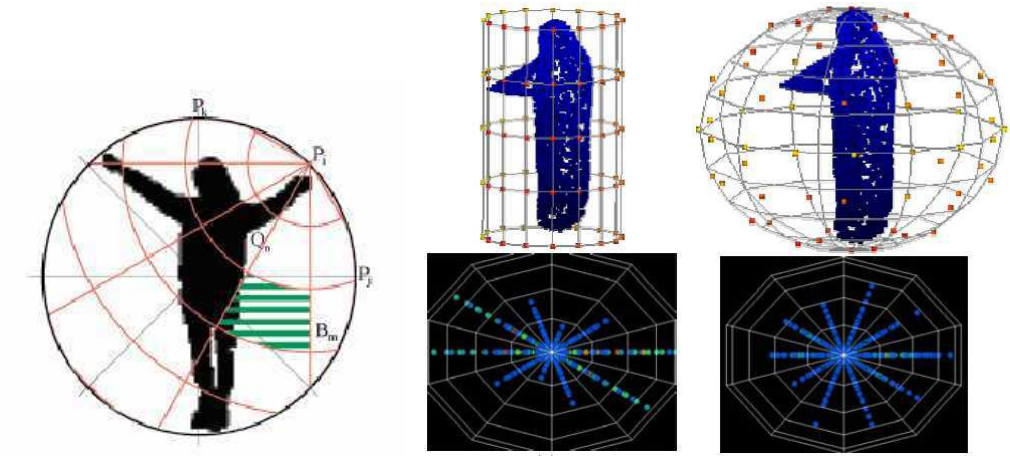
(a) (b) (c) (d)

FIG. 3.3 – Shape Context 3D utilisé dans [106]
a : surface des voxels reconstruits. **b :** voxels échantillonnés sur la surface. **c :** diagramme de Shape Context 3D. **d :** section du diagramme sur le plan $Y - Z$.

une silhouette 2D, les auteurs définissent un cercle de référence (voir figure 3.4(a)), sur lequel sont régulièrement répartis des points de contrôle. Pour chaque point de contrôle P_i, un histogramme décrit la répartition des points de la silhouette en coordonnées polaires dans un repère ayant pour origine le point P_i (voir figure 3.4(a)). L'espace est partitionné suivant un ensemble de subdivisions radiales et angulaires. L'histogramme dénombre ensuite le nombre de points de la silhouette contenus dans ces différentes subdivisions. L'invariance par rotation est obtenue en sommant l'ensemble des histogrammes construits pour tous les points de contrôle. L'histogramme est aussi normalisé en divisant par la valeur du secteur ayant obtenu le score maximum.

Pour généraliser cette description au cas d'une silhouette 3D, le cercle de référence est remplacé soit par un cylindre dont l'axe principal passe par le centre de gravité de la reconstruction (figure 3.4(b)), soit par une sphère (figure 3.4(c)). Dans le premier cas, le descripteur obtenu est invariant par rotation autour de l'axe principal du cylindre, tandis que dans le second cas, le descripteur est invariant par rotation 3D autour du centre de la sphère. A la différence du descripteur 2D, le descripteur 3D s'appuie seulement sur la surface de la reconstruction. Partant d'un des points de contrôle, un système de coordonnées sphériques est construit, et des secteurs 3D sont définis en créant des divisions suivant le rayon r, et les angles θ et ϕ. L'histogramme est construit en comptant le nombre de points de surface présents dans chacun de ces secteurs. Comme en 2D, l'histogramme final est obtenu en sommant l'ensemble des histogrammes des points de contrôle et en divisant par un

coefficient de normalisation.



(a) (b) (c)

FIG. 3.4 – Descripteur 3D présenté dans [26].
a : descripteur 2D. **b :** descripteur 3D avec une surface de référence cylindrique. **c :** descripteur 3D avec un surface de référence sphérique.

3.3 Notre descripteur

Dans notre étude, une réflexion a été menée sur le choix du descripteur à utiliser pour faire la régression. Un descripteur basé sur les moments géométriques 3D nous a semblé trop simpliste pour coder une information aussi complexe que la configuration d'un modèle articulé du corps. Des moments plus riches tels que les moments d'ondelettes proposés dans [121] seraient peut être plus adaptés ; cette étude a montré leur efficacité pour de la classification de postures, et un meilleur pouvoir de discrimination que les moments de Zernike 3D ou les coefficients de Fourier 3D, mais on peut penser qu'ils ne capturent pas assez finement la forme 3D pour estimer des variations subtiles de la pose dans le cas de la régression. Qui plus est, ce descripteur possède des propriétés d'invariance par rotation et symétrie, non compatibles avec ce que nous voulons. Les histogrammes de Shape Context 3D (3DSC) ont été appliqués avec succès par les auteurs de [106] au cas de l'estimation de la pose par régression à partir d'une reconstruction en voxels. Cependant, nous avons vu au paragraphe 3.2.1 que, malgré leur complexité calculatoire plus élevée que certains autres descripteurs plus classiques, leurs performances n'étaient pas nécessairement meilleures dans le cas de silhouettes 2D. On peut penser que la version en 3D de ce descripteur présente les mêmes défauts qu'en 2D. Comme la description se base entièrement sur la surface des voxels reconstruits, elle risque d'être très sensible aux erreurs d'extraction des silhouettes sur les séquences réelles (cette idée est d'ailleurs confirmée par les résultats expérimentaux de [117] dans le cas des histogrammes 2D). La formulation de ce descripteur est en outre assez complexe, et elle ne permet de pas se représenter de manière claire la signification des différentes composantes.

Nous avons donc proposé un nouveau descripteur 3D, basé sur sur la répartition spatiale des voxels à l'intérieur d'un cylindre de référence englobant la forme 3D reconstruite. Ce descripteur se rapproche de celui proposé par Cohen et Li dans [26], mais dans leur cas, le fait de sommer les composantes pour les différents points de contrôle efface une partie de l'information utile sur la pose (dans leur travaux, les auteurs utilisent cette description simplement pour de la classification). La formulation de notre descripteur est assez intuitive et ses composantes rendent directement compte de la répartition de la matière dans l'enveloppe reconstruite.

3.3.1 Principe général

Etant donnée une silhouette 3D reconstruite en voxels (voir figure 3.5), on définit un cylindre de référence dont l'axe principal est l'axe vertical passant par le centre de gravité de la reconstruction (positionnement par défaut, voir partie 3.3.6 sur le centrage du descripteur), et dont le rayon est proportionnel à la hauteur de la reconstruction (on suppose que les points les plus hauts de la reconstruction correspondent à des points de la tête, et donc que la hauteur de la reconstruction est égale à la taille de personne).

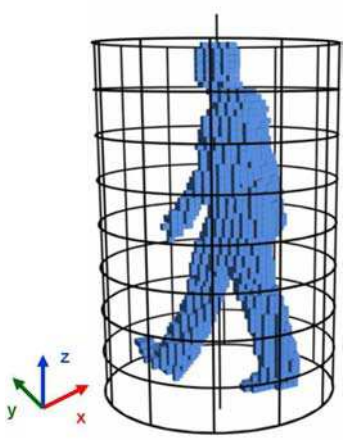


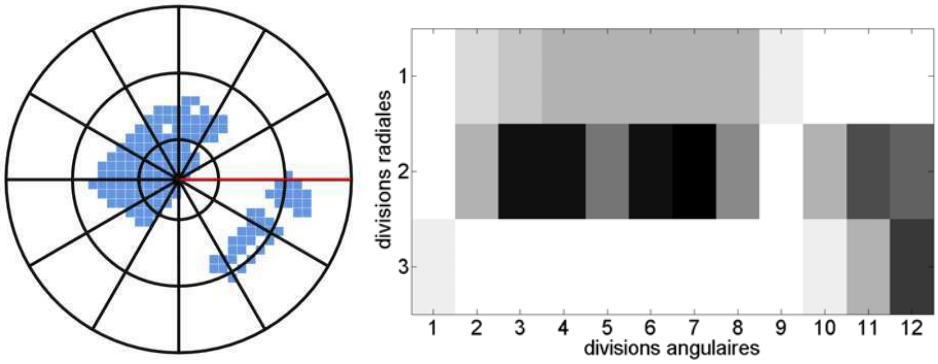
FIG. 3.5 – Exemple de reconstruction 3D en voxels et cylindre de référence du descripteur.

Histogramme 2D

Pour chaque section horizontale des voxels, le disque défini par le cylindre est divisé en une grille composée de graduations suivant le rayon et l'angle. Un histogramme 2D est calculé en comptant le nombre de voxels de la forme 3D contenus dans chaque secteur (la position du centre d'un voxel détermine son appartenance à un secteur). Un exemple d'histogramme 2D est donné sur la figure 3.6.

Descripteur 3D

La hauteur de la forme 3D est ensuite divisée en n_t tranches verticales. Pour chaque tranche, un nouvel histogramme 2D est calculé en moyennant l'ensemble des histogrammes des couches de voxels contenues dans la



(a) (b)

FIG. 3.6 – Histogramme 2D.
a : couche horizontale de voxels. **b :** histogramme 2D correspondant.

tranche. Le descripteur 3D est constitué de la concaténation de ces descripteurs moyens.

3.3.2 Lissage

Dans la description précédente, tous les voxels d'un secteur apportent la même contribution au baquet de l'histogramme correspondant. Un voxel localisé près de la frontière d'un secteur est pris en compte de la même façon qu'un voxel situé au centre. Ce codage peut poser des problèmes, car il engendre des discontinuités dans la description, et l'empêche d'être robuste au bruit et aux erreurs légères de segmentation ou de positionnement du descripteur (comme par exemple à des petites variations du centrage ou de l'orientation, voir paragraphes 3.3.5 et 3.3.6). En effet, si le voxel s'est légèrement déplacé d'une reconstruction à une autre (par rapport au cylindre de référence), son centre peut changer de secteur, et sa contribution au descripteur saute sans transition d'une composante de l'histogramme à une autre.

Par ailleurs, si la distance utilisée pour comparer deux descripteurs est la distance euclidienne, la proximité de forme entre des objets n'est pas correctement représentée par le descripteur. Par exemple, sur la figure 3.7, la comparaison des histogrammes des trois formes en distance euclidienne ne reflète pas la similitude réelle entre les objets, puisque la distance entre les objets a et c sera la même qu'entre les objets a et b, bien que les objets a et b représentent le même objet qui a subi une légère rotation.

Une première façon de gérer ce problème est d'utiliser une distance prenant en compte la proximité de deux secteurs de la grille lors de la com-

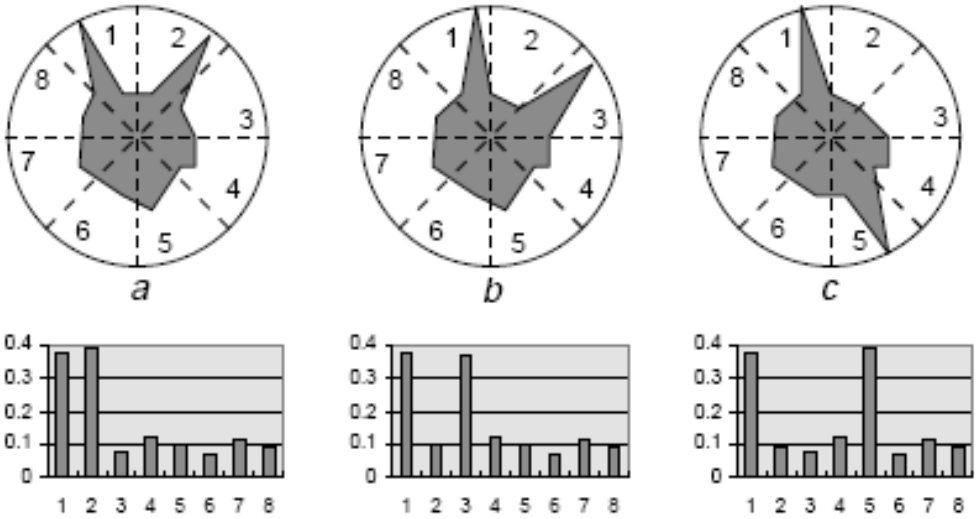


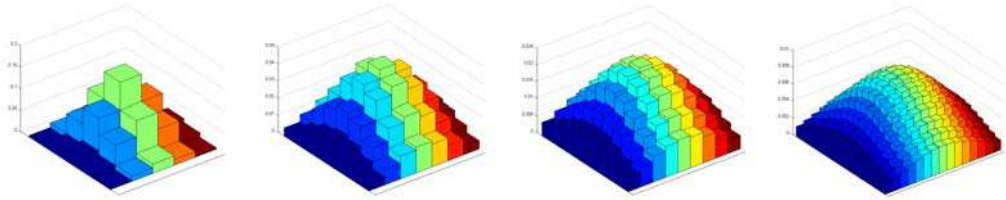
FIG. 3.7 – Illustration des problèmes posés par la discrétisation spatiale et l'utilisation de la distance euclidienne dans un histogramme 2D (figure extraite de [8]).

paraison des histogrammes. Les auteurs de [8] proposent ainsi d'utiliser une distance quadratique de la forme :

$$d_A^2(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^T.A.(\mathbf{x}-\mathbf{y}) = \sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}(x_i-y_i)(x_j-y_j) \qquad (3.3)$$

où la matrice A encode la similarité entre deux composantes i et j du vecteur descripteur. Dans le cas de notre histogramme, les termes de la matrices A peuvent être par exemple de la forme $a_{ij} = e^{-\sigma.d(i,j)}$, où $d(i,j)$ représente la distance entre deux secteurs de la grille. L'inconvénient de cette méthode dans notre cas est qu'elle augmente la complexité des calculs nécessaires à la comparaison entre deux vecteurs, et donc rallonge le temps de calcul aussi bien à l'apprentissage qu'à l'estimation.

Une deuxième solution, retenue dans cette thèse, est de permettre à un voxel d'apporter une contribution dans un voisinage autour de son centre plutôt qu'en un seul point. Dans notre descripteur, un voxel "vote" également pour les voxels voisins, avec des poids inversement proportionnels à leur éloignement. Ainsi, un voxel localisé près de la frontière d'un secteur vote aussi pour les secteurs voisins. Les poids de ses votes sont répartis suivant une gaussienne centrée sur son centre. La figure 3.8 représente les votes d'un voxel en fonction de l'écart-type choisi pour la gaussienne. Sur la figure sont



(a) $\sigma = 1$ (b) $\sigma = 2$ (c) $\sigma = 3$ (d) $\sigma = 5$

FIG. 3.8 – Représentation des votes d'un voxel en fonction de l'écart-type σ de la gaussienne choisi pour le lissage 2D.

représentés les votes du voxel situé au centre de la gaussienne pour les voxels de son voisinage. Plus l'écart-type est élevé, plus l'influence du voxel pour les points de son voisinage est étendue. Cette approche permet de comparer deux histogrammes en utilisant une simple distance euclidienne.

De la même façon, l'utilisation des histogrammes moyens d'une tranche verticale pour former le descripteur final peut poser des problèmes de continuité. Par exemple, si le bras est levé près de la frontière entre deux tranches, un petit mouvement peut faire basculer tous les voxels d'une tranche à l'autre. Deux postures voisines devraient alors générer deux descripteurs assez différents, puisque les voxels passent brutalement d'un secteur à l'autre dans l'histogramme. Pour assurer la continuité du descripteur lors de la concaténation des descripteurs moyens des tranches, une pondération est appliquée aux couches des voxels de la tranche avant de calculer l'histogramme moyen (voir figure 3.9). Les poids sont fixés suivant une gaussienne centrée sur la couche de voxels située au centre de la tranche, et d'écart-type σ égal à la demi-épaisseur d'une tranche. Son support (fixé à 4σ) déborde sur les tranches voisines, de sorte que l'histogramme moyen contient des informations sur les tranches de voxels inférieure et supérieure.

3.3.3 Choix du nombre de divisions

Le descripteur est paramétré par le nombre de divisions verticales n_t (c'est-à-dire le nombre de tranches) et le nombre de secteurs angulaires n_θ et radiaux n_r sur chaque couche horizontale. Intuitivement, plus il y a de divisions dans le descripteur, plus l'information qu'il contient est riche ; un descripteur avec des divisions très fines pourrait contenir la position de tous les voxels de la reconstruction. D'un autre côté, si les divisions sont trop fines, le descripteur risque de devenir sensible au bruit et d'encoder des informations non pertinentes pour estimer la pose, qui pénalisent la régression. L'intérêt

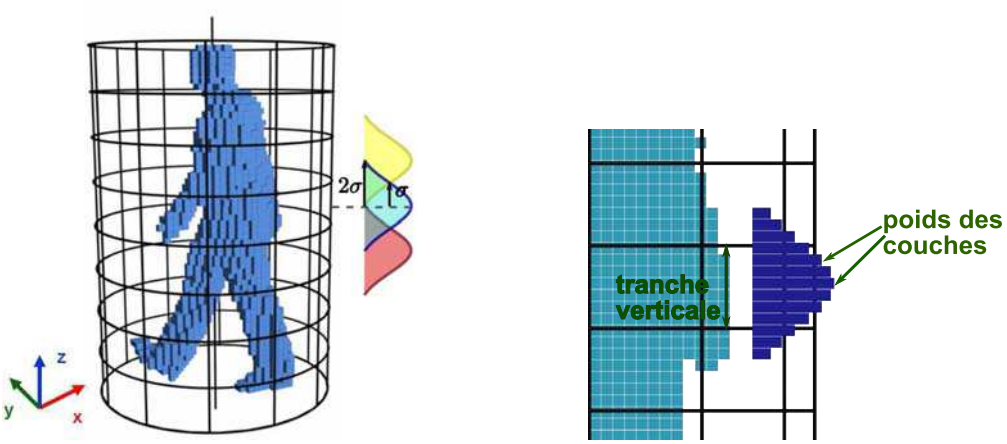


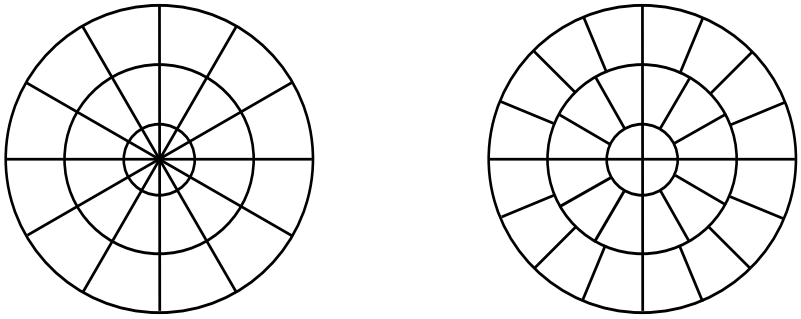
FIG. 3.9 – Représentation des poids associés aux couches de voxels dans le calcul des histogrammes moyens des tranches.

d'utiliser un descripteur est justement de pouvoir capturer l'information la plus importante tout en filtrant les détails superflus. Le descripteur doit être une représentation compacte de la forme 3D : sa dimension doit rester raisonnable sous peine de devenir un handicap pour la phase d'apprentissage. Chaque paramètre doit donc être choisi de manière à trouver le meilleur compromis sur la précision du descripteur, sa dimension et sa robustesse au bruit et aux erreurs de segmentation. Comme les secteurs n'ont pas la même taille suivant la couche radiale sur laquelle ils sont situés, nous avons choisi de fixer des nombres de divisions angulaires différents suivant l'éloignement de la couche radiale à l'axe du descripteur (voir figure 3.10).

3.3.4 Invariance à l'échelle et à la corpulence

Comme il a été dit en introduction, il est important d'essayer de rendre au maximum le descripteur invariant aux changements d'apparence entre les personnes en tenant compte des différences de taille, de corpulence, de morphologie et d'habillement. Une réelle invariance du descripteur à tous ces facteurs est très difficile à obtenir.

Dans la construction de notre descripteur, plusieurs choix ont été faits pour rendre la représentation invariante à l'échelle. Tout d'abord, le descripteur est invariant à la taille du sujet puisque le positionnement des tranches verticales se base sur la hauteur de la tête. Ensuite, l'invariance par rapport au volume total de la reconstruction est obtenue en divisant le score de chaque secteur par le nombre total de voxels présents dans la reconstruc-



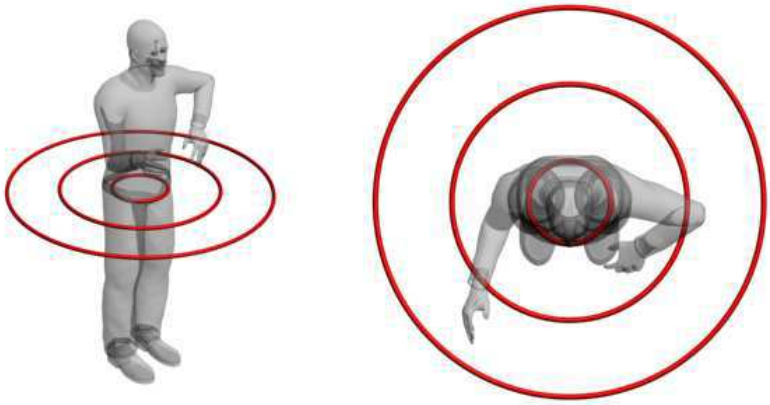
(a) (b)

FIG. 3.10 – Exemples de découpages possibles du disque défini par le cylindre de référence pour le calcul des histogrammes 2D.
a : découpage avec 12 divisions angulaires par couche radiale. **b :** découpage avec respectivement 4, 12 et 16 divisions angulaires sur les trois couches radiales.

tion : chaque composante représente la proportion de la masse totale située dans une certaine partie du cylindre. Enfin, la taille des différentes divisions radiales est fixée de façon à tenir compte des différences de corpulence entre les personnes. Les rayons sont choisis de la façon suivante (voir figure 3.11) :

- le diamètre du cercle intérieur est choisi de façon à approximer l'épaisseur de la taille de l'avatar
- le rayon extérieur est fixé proportionnellement à la hauteur de la personne
- le rayon intermédiaire est fixé à mi-chemin entre le rayon interne et le rayon externe

Un facteur de proportionnalité a été calculé sur les avatars de la base d'apprentissage (ceux de la figure 2.13) pour obtenir une approximation de l'épaisseur de la taille en fonction de la hauteur et du volume de la reconstruction. Le rayon extérieur a été fixé de façon à pouvoir contenir tous les voxels dans le cas où les bras ou les jambes sont écartés. Nous avons considéré que les amplitudes des bras et des jambes d'une personne sont proportionnelles à sa taille, ce rayon est donc proportionnel à la hauteur de la reconstruction.
Avec cette disposition, le rayon intérieur doit contenir principalement des voxels du torse (pour le haut du corps), et les deux couches extérieures doivent donner des informations sur la position des bras et des jambes par rapport au tronc suivant qu'ils sont le long du corps, tendus ou pliés.



(a) (b)

FIG. 3.11 – Représentation des rayons des cylindres du descripteur sur l'un des avatars.
a : vue perspective. **b :** vue d'en haut.

3.3.5 Orientation du descripteur

Par défaut, les axes du descripteur sont orientés parallèlement aux axes du repère du monde : la ligne de référence de l'histogramme 2D (en rouge sur la figure 3.6), est alignée avec l'axe x du repère du monde (voir repère sur la figure 3.5). Ce choix implique que le descripteur n'est pas invariant par rotation : pour une posture donnée, le descripteur varie suivant l'orientation du corps par rapport au repère du monde. Cette dépendance par rapport à l'orientation du corps permettra par la suite d'estimer l'angle global du corps par régression.

Nous verrons dans la partie 4.3.1 que pour faciliter l'estimation de la configuration des articulations des membres du corps, il peut être utile de distinguer dans l'estimation l'angle d'orientation globale du corps (choisie comme étant l'orientation du torse dans nos expériences), et les angles des articulations du modèle. Pour effectuer cette dernière estimation, l'orientation du descripteur est recalée sur l'orientation du torse par rapport à l'axe z du repère du monde (la ligne rouge est alignée avec le plan du torse).

3.3.6 Centrage

Dans cette section, on s'intéresse au positionnement de l'axe principal du cylindre qui définit le descripteur. Le centrage du descripteur est un élément

important car il détermine la position de tous les secteurs du descripteur par rapport au corps. Nous avons vu dans la partie 3.3.1 que l'axe vertical du descripteur est placé par défaut sur les coordonnées horizontales du centre de masse de la reconstruction 3D. Avec ce centrage, on peut constater que l'axe du descripteur se déplace par rapport au corps en fonction de la position des bras et des jambes. Ainsi, en fonction de la position des autres membres, les voxels correspondant à une partie du corps ne sont pas localisés dans le même secteur du descripteur. Par exemple, les secteurs correspondant au bras droit (même fixe) se déplacent en fonction de la position du bras gauche. Nous avons donc cherché à rendre le centrage du descripteur le plus indépendant possible de la posture. Pour cela, nous avons choisi de centrer le descripteur sur le centre du torse de le personne. Dans [38], les auteurs utilisent une ACP itérative pour déterminer l'axe du torse dans les images d'une personne acquises par un système de quatre caméras. Dans ces travaux, l'axe 2D du torse est extrait dans chaque image, et leur combinaison permet de remonter à une estimation de la position de l'axe dans le repère 3D. Dans le même ordre d'idées, un ellipsoïde approximant l'ensemble des voxels du torse est calculé itérativement de la manière suivante :

1. Calculer la moyenne μ et la covariance Σ du nuage N de points 3D défini par l'ensemble des voxels de la reconstruction.

2. Retirer de N tous les voxels v_i tels que $D_M(v_i) = \sqrt{(v_i - \mu)^T \Sigma^{-1} (v_i - \mu)} > dist$

3. Calculer la moyenne μ' du nouveau nuage de points.
   si $\|\mu - \mu'\| < s$, fin
   sinon aller en 1.

Un premier ellipsoïde approximant le nuage global des voxels est estimé, puis les voxels les plus éloignés du centre (pour la distance de Mahalanobis) sont successivement éliminés. L'algorithme s'arrête lorsque la distance entre les centres des deux ellipsoïdes successifs est inférieure à un seuil s, et que le nuage de point résultant est bien approximé par l'ellipsoïde. Dans nos expériences, les seuils ont été fixés à $dist = 2,3$ et $s = 10^{-4}\ m$. Cette approximation itérative du torse se rapproche de [71] dans lequel un template de torse est itérativement ajusté sur les voxels en calculant un nouveau

centroïde.

Quelques exemples de calcul d'ellipsoïdes sont donnés sur la figure 3.12. Sur cette figure, les voxels de la reconstruction situés à l'intérieur de l'ellipsoïde sont représentés en rouge, et les autres voxels en blanc. Si l'avatar a les pieds joints (figure 3.12, centre), l'ellipsoïde englobe à la fois les jambes et le torse. Si les jambes et les bras sont écartés (figure 3.12, gauche et droite), les voxels correspondant sont éliminés de l'ellipsoïde, et seuls les voxels du torse sont retenus. Dans tous les cas, on peut considérer que cet ellipsoïde donne une bonne approximation de la position (x, y) du torse. Une fois l'ellipsoïde estimé, l'axe du descripteur est centré sur les coordonnées horizontales de son centre.

Comme on le verra dans la partie expérimentale de ce chapitre, l'effet du centrage est plus important si la reconstruction présente de gros artéfacts, car la masse de voxels supplémentaire sur la silhouette déplace plus fortement le centre de gravité.

3.3.7 Choix des composantes "utiles"

Comme il a été dit précédemment, la régression fonctionne mieux sur des vecteurs de dimension faible et qui résument de façon compacte l'information pertinente pour estimer les paramètres de sortie, en lissant les effets du bruit. Toutes les informations superflues qu'il encode peuvent gêner l'estimation. Ainsi, si l'information contenue dans un des secteurs du descripteur n'est pas utile pour estimer un paramètre, la composante associée à ce secteur dans le vecteur descripteur représente un bruit pour la machine de régression. Dans le cas où le descripteur est recalé sur l'orientation du torse, et centré sur le centre du torse, les secteurs correspondant à l'un des membres du corps doivent toujours correspondre à une région précise du cylindre de référence. Seule une partie des secteurs du descripteur est affectée par les mouvements d'un membre du corps (un bras, une jambe...). Dans les expérimentations, nous avons donc testé l'influence de la suppression des composantes "inutiles" pour estimer un degré de liberté précis. En particulier, il est possible de n'évaluer les mouvements des jambes qu'avec les secteurs situés dans la partie inférieure du cylindre, et ceux des bras qu'avec la partie supérieure. De même, pour la reconnaissance des gestes, on peut penser que les secteurs les plus opposés à un bras dans la partie haute du descripteur n'apportent pas d'information utile pour estimer la position de ce bras.

La décomposition du descripteur en un ensemble de régions du corps quasi-indépendantes peut aussi permettre de reconnaitre une gamme de pos-

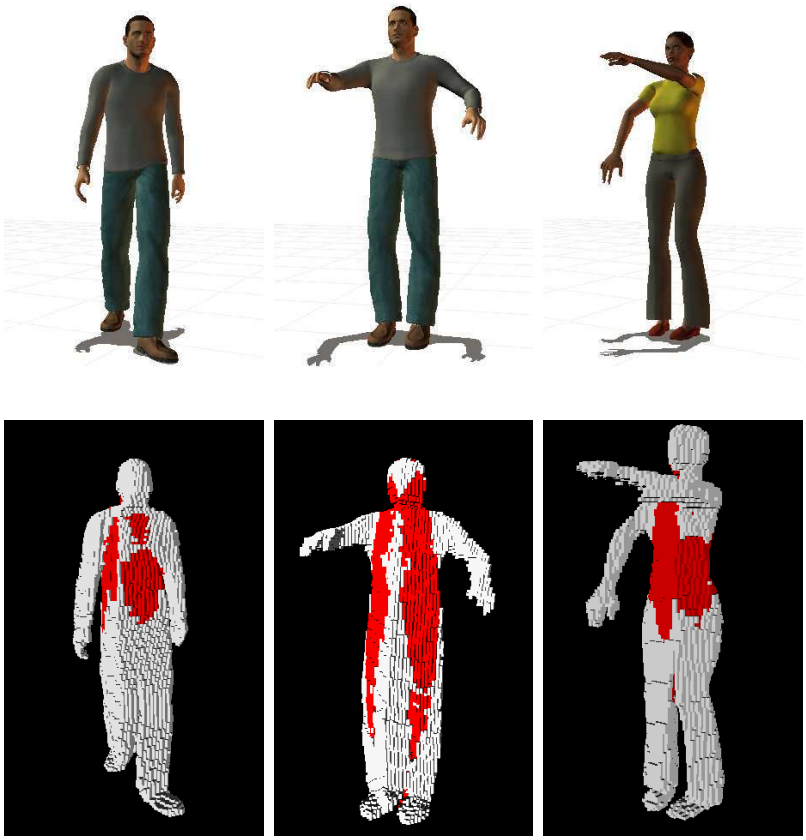


FIG. 3.12 – Exemples d'ellipsoïdes calculés à partir des reconstructions 3D pour positionner l'axe central du descripteur.

tures plus étendues à partir d'une même base d'apprentissage : l'analyse n'est plus limitée uniquement aux postures proches de celles de la base car toutes les poses résultant de la combinaison des mouvements des différentes parties du corps peuvent être prises en compte.

3.4 Evaluation expérimentale des paramètres du descripteur

3.4.1 Présentation des expérimentations

Comme on l'a vu dans la partie précédente, le descripteur dépend d'un certain nombre de paramètres qui peuvent jouer sur la qualité de l'encodage. Le choix de ces paramètres a des implications à la fois sur la dimension du vecteur descripteur (nombre de subdivisions dans les différentes directions), sur la continuité de la description par rapport à la pose (lissage) et plus généralement sur la cohérence de la structure de l'espace d'entrée de la machine de régression, par rapport à l'ensemble des poses que l'on souhaite modéliser. Il conditionne donc le déroulement de la phase d'apprentissage et la qualité de l'estimation.

L'objectif de cette section est d'évaluer expérimentalement l'influence de ces différents paramètres sur la qualité du descripteur, pour avoir une idée de sa configuration optimale. Le choix du descripteur doit se faire en gardant bien à l'esprit qu'une représentation plus concise facilitera par la suite l'apprentissage.

Dans ces expériences, une difficulté vient du fait que le nombre de paramètres à faire varier est très important (nombre de subdivisions, paramètres de lissage, centrage, normalisation...) et que certains d'entre eux dépendent les uns des autres. Pour trouver le meilleur descripteur, il faudrait théoriquement pouvoir faire varier chaque paramètre indépendamment des autres, et garder la meilleure configuration parmi toutes les combinaisons. Par exemple, pour fixer les nombres de divisions radiales, angulaires et verticales, il faudrait faire des tests avec tous triplets de paramètres. Le nombre de possibilités est toutefois trop important pour tester toutes les combinaisons. Nous avons donc choisi d'évaluer individuellement l'influence de chaque paramètre en partant d'une configuration fixe du descripteur.

Mesure de la qualité du descripteur

Nous souhaitons dans cette partie évaluer différentes formes du descripteur sur leur capacité à encoder des informations pertinentes pour estimer la pose, mais indépendamment de l'algorithme de régression qui servira par la suite à évaluer cette pose. Une méthode pour comparer différents descripteurs pourrait en effet consister à construire des bases d'apprentissage et de tests composées de paires descripteur/pose et de comparer les résultats obtenus avec un algorithme de régression donné, en calculant l'erreur sur les poses de la base de tests. Dans ce cas, l'évaluation porterait sur l'ensemble du processus d'estimation (apprentissage+descripteur). Des tests de ce type seront présentés dans les chapitres 4 et 5.

Quelque soit la méthode utilisée, pour prédire la pose, l'algorithme d'apprentissage se base sur l'analyse de la similitude entre les données dans l'espace du descripteur. Etant donné un nouveau vecteur d'entrée $\mathbf{x}$, l'application apprise cherchera dans les données d'apprentissage les exemples ayant un descripteur proche de cette nouvelle entrée et s'appuiera sur les poses associées aux vecteurs de son voisinage pour effectuer une prédiction. Pour que l'apprentissage se déroule le mieux possible, il faut donc que la structure des données dans l'espace des descripteurs et dans l'espace des poses soient les plus proches possibles, c'est-à-dire que des poses proches correspondent à des vecteurs voisins dans l'espace des descripteurs et que deux poses éloignées génèrent des observations distantes. Pour avoir une idée de la qualité d'un descripteur, on peut donc étudier à quel point il représente les similitudes et les divergences de l'espace des poses. Pour cela, on peut regarder si l'ordre des données dans l'espace des poses est retranscrit dans l'espace des descripteurs. Idéalement, quelque soit la valeur de k, les k plus proches voisins d'un point dans l'espace des descripteurs devraient correspondre aux k plus proches voisins dans l'espace des poses.

Nous avons donc choisi dans nos expériences d'adapter la mesure présentée dans [118]. On se donne une base d'exemples composée de N poses $\{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N\}$ auxquelles sont associés N vecteurs $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ ($\mathbf{x}_i \in \mathbf{R}^n$) dans l'espace des descripteurs. On définit dans l'espace des poses une distance entre deux éléments en calculant la somme des distances euclidiennes entre les points 3D du corps. Si une pose $\mathbf{y}_i$ est déterminée par les positions de M points du corps, $(y_i^1, ..., y_i^M)$, la distance entre deux éléments $\mathbf{y}_i$ et $\mathbf{y}_j$ dans l'espace des poses est donnée par :

$$d(\mathbf{y}_i, \mathbf{y}_j) = \sum_{k=1}^{M} \left\| y_i^k - y_j^k \right\| \tag{3.4}$$

Etant donné un descripteur $\mathbf{x}_q$, les éléments de la base peuvent être ordonnés en fonction de leur proximité à $\mathbf{x}_q$ dans l'espace des descripteurs (distance euclidienne de $\mathbf{R}^n$). On classe ainsi les données de la base en notant $r(1)$ l'indice du vecteur le plus proche et $r(N)$ l'indice du vecteur le plus éloigné. Pour un entier k donné ($k \in \{1, ..., N\}$)), on regarde si les k plus proches voisins de $\mathbf{x}_q$ dans l'espace du descripteur correspondent bien aux k plus proches voisins dans l'espace des poses. On examine pour cela la distance moyenne dans l'espace des poses du vecteur à ses k plus proches voisins dans l'espace du descripteur :

$$f(k) = \frac{\sum_{j=1}^{k} d(\mathbf{y}_{r(j)}, \mathbf{y}_q)}{k} \tag{3.5}$$

Cette distance $f(k)$ est comparée avec la distance moyenne $f(N)$ de $\mathbf{y}_q$ à tous les éléments de la base. Si les éléments de la base ont été classés dans le bon ordre, alors le rapport $f(k)/f(N)$ devrait être faible. A l'inverse, si l'ordre des éléments ne se correspondent pas du tout dans les deux espaces, les k éléments ne sont pas plus proches de $\mathbf{y}_q$ que si des éléments de la base avaient été tirés au hasard, et le rapport $f(k)/f(N)$ devient proche de 1. Pour représenter le comportement des distances pour différentes valeurs de k, la courbe représentant les valeurs de $f(k)/f(N)$ en fonction de k/N est tracée. La courbe est comparée avec la courbe idéale, c'est-à-dire celle qui est obtenue lorsque les données sont classées dans le même ordre dans les deux espace, et avec la courbe obtenue si les éléments sont classés aléatoirement (i.e. $f(k)/f(N) = 1$). Un exemple de courbe est donné sur la figure 3.13. Chaque courbe peut être interprétée comme une mesure de la corrélation entre les distances dans l'espace des poses et l'espace des descripteurs : une forte corrélation est traduite par une courbe "basse", proche de la courbe idéale, tandis qu'une mauvaise corrélation donne une courbe "haute". Les tests qui suivent s'appuient sur une base d'exemples pour laquelle chacun des exemples a été successivement sorti de la base et comparé avec les autres exemples. Les courbes tracées représentent la valeur moyenne des courbes obtenues sur l'ensemble des exemples.

Base de tests utilisée

Une base de test a été synthétisée avec l'avatar par défaut du logiciel POSER 6. La base est constituée de 500 exemples de l'avatar dans différentes poses. Chaque exemple est généré en tirant aléatoirement les angles des bras dans des intervalles réalistes. Les 8 degrés de liberté utilisés ainsi que les

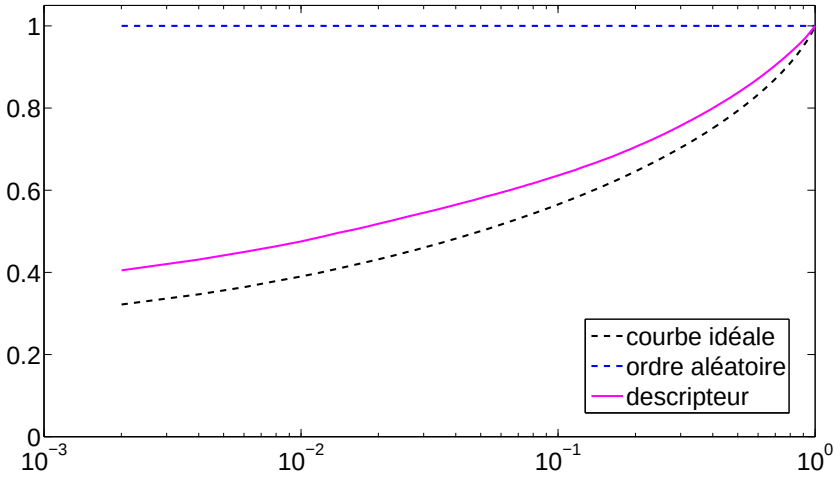


FIG. 3.13 – Exemple de graphe obtenu en représentant les valeurs de $f(k)/f(N)$ en fonction de k/N (échelle logarithmique sur l'axe des abscisses).

(a) (b) (c) (d) (e)

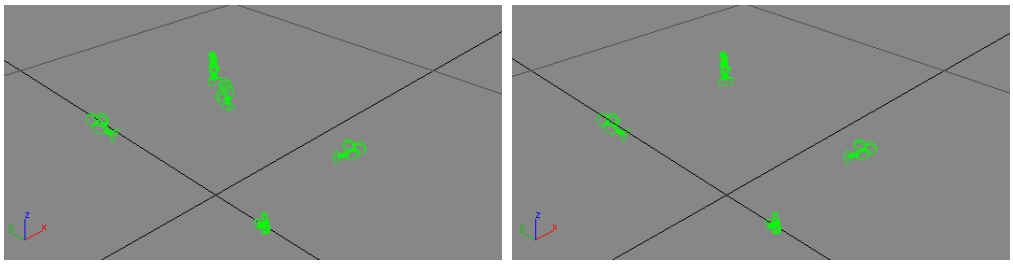
FIG. 3.14 – Quelques exemples de poses de la base de tests.

intervalles des valeurs possibles pour chacun des angles sont détaillés dans le tableau 3.1. Quelques exemples de poses générées sont montrés sur la figure 3.14.

Le système de caméra utilisé dans les tests est le système de 5 caméras présenté sur la figure 3.15. Pour certains tests (voir 3.4.4), seules les 4 premières caméras du système sont utilisées, pour évaluer l'influence du nombre de caméras sur la reconstruction. L'avatar est fixe au centre du système de capture.

articulation	axe de rotation	intervalle des valeurs possibles
épaule gauche	X (autour de l'axe du bras)	[-50 , 80]
épaule gauche	Y (avant-arrière)	[-90 , 40]
épaule gauche	Z (haut-bas)	[-80 , 30]
épaule droite	X (autour de l'axe du bras)	[-50 , 80]
épaule droite	Y (avant-arrière)	[-40 , 90]
épaule droite	Z (haut-bas)	[-30 , 80]
coude gauche	Y (plié-tendu)	[-120 , 20]
coude droit	Y (plié-tendu)	[-20 , 120]

TAB. 3.1 – Degrés de liberté et valeurs des angles (en degrés) utilisés pour générer la base d'exemples.



(a) (b)

FIG. 3.15 – Systèmes de caméras utilisés dans les tests.
a : système de 5 caméras. **b :** système utilisé pour les essais avec 4 caméras (la caméra au plafond est supprimée).

3.4.2 Influence du lissage

Nous avons vu dans la partie précédente que le lissage du descripteur est important pour assurer la continuité de la description par rapport aux changements de la pose. Dans ce paragraphe, on étudie son influence sur la cohérence des distances dans l'espace du descripteur par rapport à la similitude dans l'espace des poses.

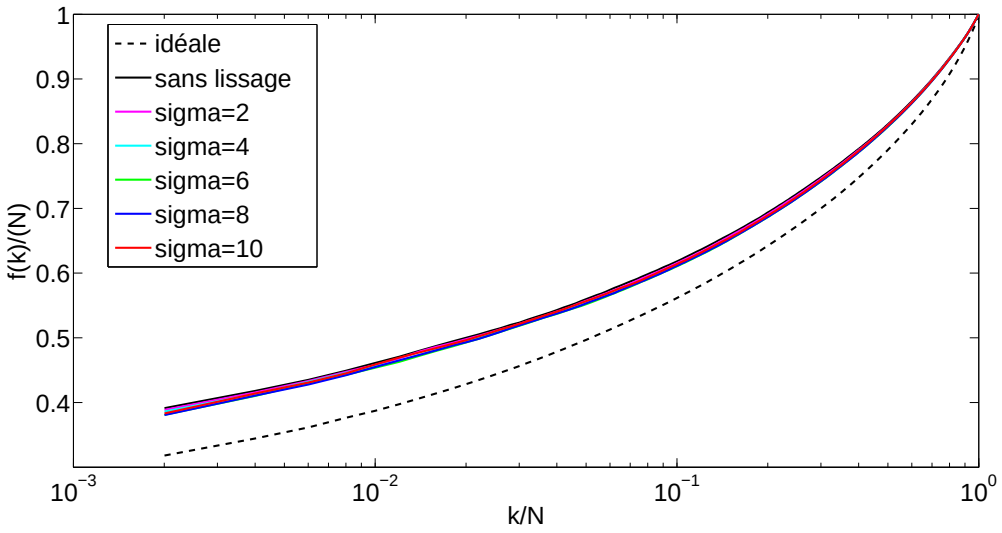
Dans une première expérience, nous avons tracé les courbes obtenues pour différentes valeurs de l'écart type dans le lissage 2D. Le descripteur considéré contient 10 couches verticales, et respectivement 4, 12 et 16 divisions angulaires pour les 3 couches radiales. Pour chaque valeur de σ, le support de la gaussienne est choisi de manière à atteindre des voxels situés entre 0 et 3σ du centre. Le résultat est donné sur la figure 3.16.

L'analyse de ces courbes montre que le lissage 2D des couches horizontales du descripteur est bénéfique. Les résultats s'améliorent lorsque l'écart-type augmente, jusqu'à une certaine valeur au delà de laquelle les courbes semblent remonter. La valeur optimale de σ semble se situer entre $\sigma = 6$ et $\sigma = 8$. Si lissage est trop important, l'information sur la forme de la silhouette est noyée dans les différentes composantes du descripteur. Il faut souligner que les expériences sont menées ici avec des silhouettes 3D reconstruites à partir de silhouettes 2D parfaites ; la présence de trous ou des pixels supplémentaires dans les silhouettes réelles peut engendrer des bruits dans la reconstruction 3D. On peut penser que leurs effets pourront être atténués par le lissage.

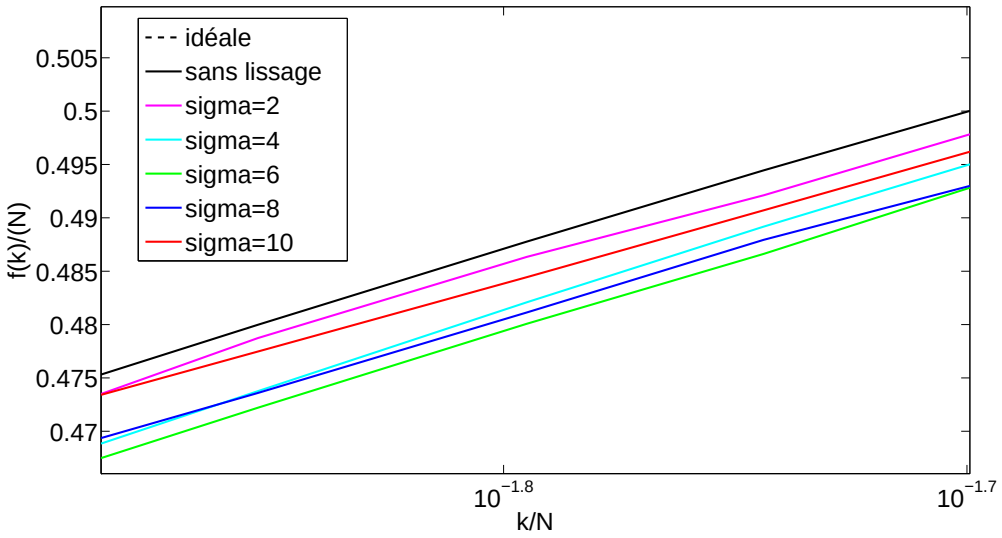
L'expérience qui suit vise à mettre en évidence les bénéfices de l'utilisation du lissage vertical avant de calculer la moyenne des descripteurs 2D d'une tranche. Deux descripteurs sont comparés :

- celui qui est obtenu si on calcule la moyenne des descripteurs 2D des couches d'une tranche sans déborder sur les couches voisines,
- le descripteur présenté dans la partie précédente, c'est-à-dire pour lequel une pondération gaussienne a été appliquée aux couches d'une tranche avant de calculer la moyenne et dans lequel les tranches voisines sont prises en compte pour calculer l'histogramme moyen.

Les courbes obtenues sont présentées sur la figure 3.17. On peut constater une légère amélioration des résultats avec le lissage vertical : les deux courbes sont proches, mais la courbe avec lissage est distincte de l'autre courbe, et plus proche de la courbe idéale.



(a)



(b)

FIG. 3.16 – Tests avec différentes valeurs de σ dans le lissage 2D.
a : Courbes globales obtenues. **b :** Zoom sur une partie de la figure.

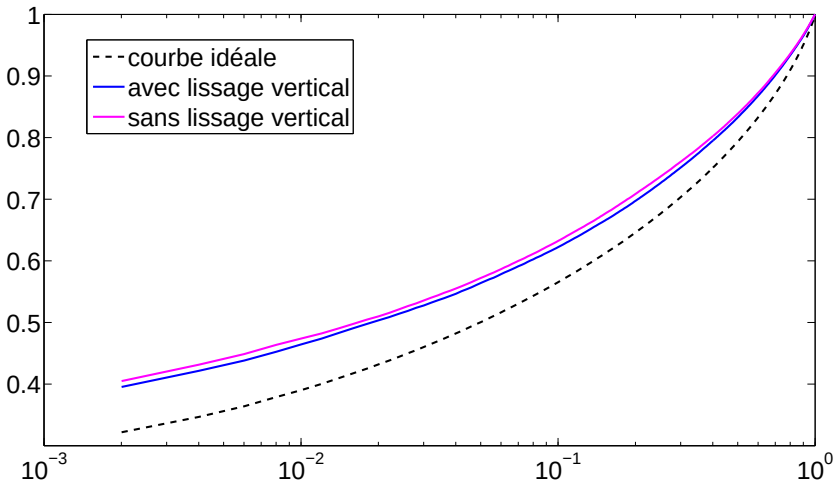


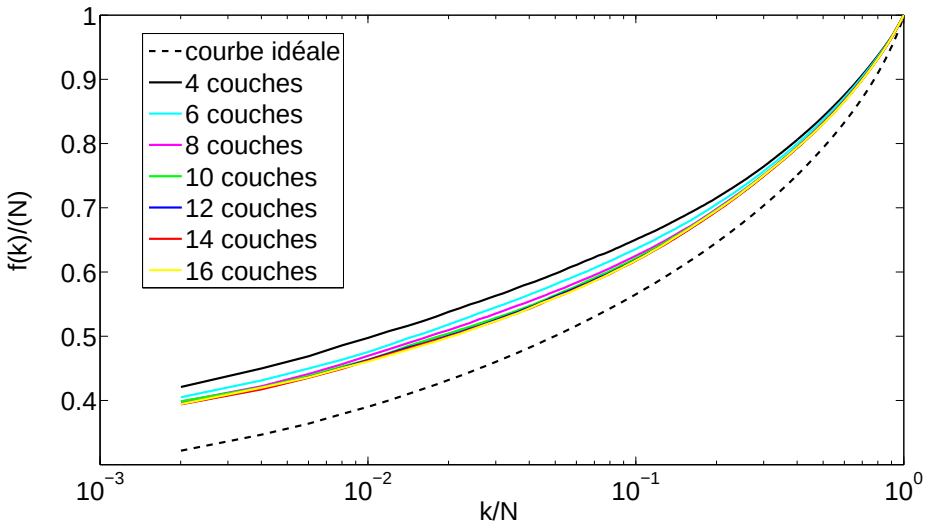
FIG. 3.17 – Courbes tracées avec et sans lissage vertical.

3.4.3 Influence du nombre de subdivisions

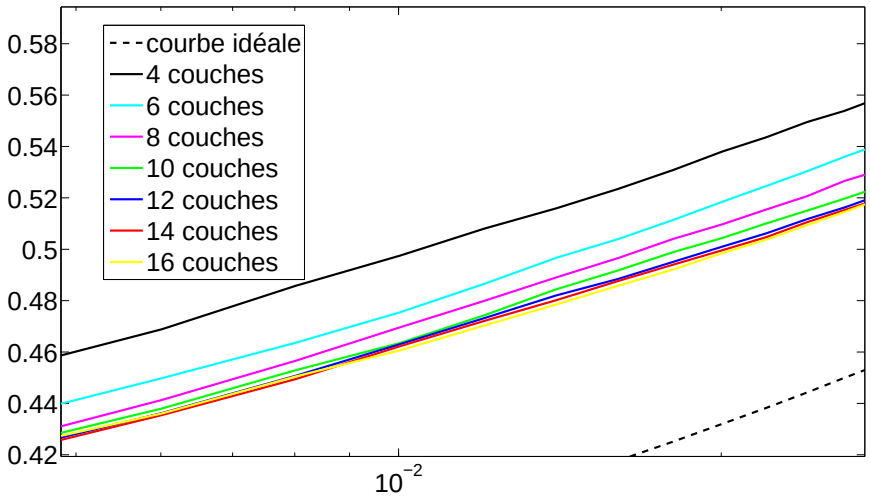
Les expériences de ce paragraphe ont pour un objectif de déterminer le découpage optimal des secteurs définissant notre descripteur, c'est-à-dire le nombre de divisions angulaires et verticales (le nombre de divisions radiales est fixé à 3).

Dans un premier temps, nous avons cherché à évaluer l'influence d'une augmentation du nombre de divisions verticales : des courbes sont tracées avec différents descripteurs qui contiennent le même nombre de secteurs radiaux et angulaires sur les couches horizontales, mais dont le nombre de sections verticales varie. Les résultats sont donnés sur la figure 3.18. On peut voir que l'augmentation du nombre de couches est très bénéfique au début (4, 6, 8 couches), mais qu'au delà de 10 couches (courbe verte), l'ajout de divisions supplémentaires n'apporte plus d'information puisque les courbes ne descendent quasiment plus ou se croisent. Il semble y avoir une sorte de courbe asymptote qui ne peut pas être dépassée quelque soit le nombre de couches dans le descripteur.

Dans une deuxième expérience, nous avons tracé les courbes obtenues pour des descripteurs contenant des nombres de couches verticales identiques, mais en faisant varier le nombre de divisions angulaires. Les courbes sont présentées sur la figure 3.19. Les courbes optimales semblent être les courbes rouge (12-12-12) et cyan (4-12-16). On constate qu'un nombre élevé de divisions sur toutes les couches radiales (courbe jaune 16-16-16) n'apporte pas



(a)



(b)

FIG. 3.18 – Courbes obtenues en faisant varier le nombre de couches horizontales dans le descripteur.

a : Courbes. **b :** Zoom sur une partie de la figure.

	4 caméras		5 caméras	
	écart-type x	écart-type y	écart-type x	écart-type y
centre de gravité	1.4	1.93	0.36	0.74
centre ellipsoïde	0.17	0.14	0.10	0.08

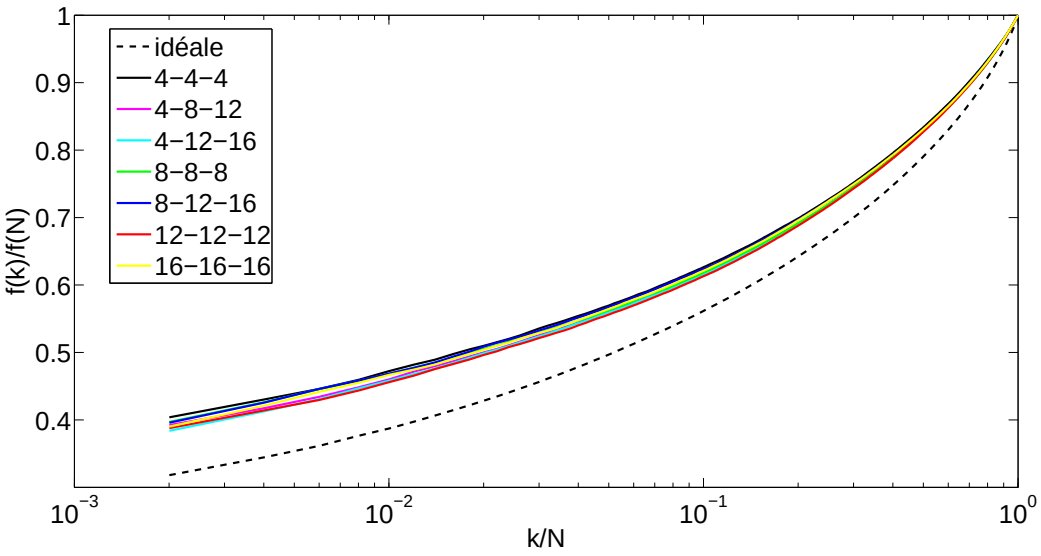
TAB. 3.2 – Déplacement (en cm) suivant les axes x et y du repère du monde du centre de gravité et du centre de l'ellipsoïde en fonction de la posture.

d'information supplémentaire, bien que la dimension du vecteur associé soit plus grande.

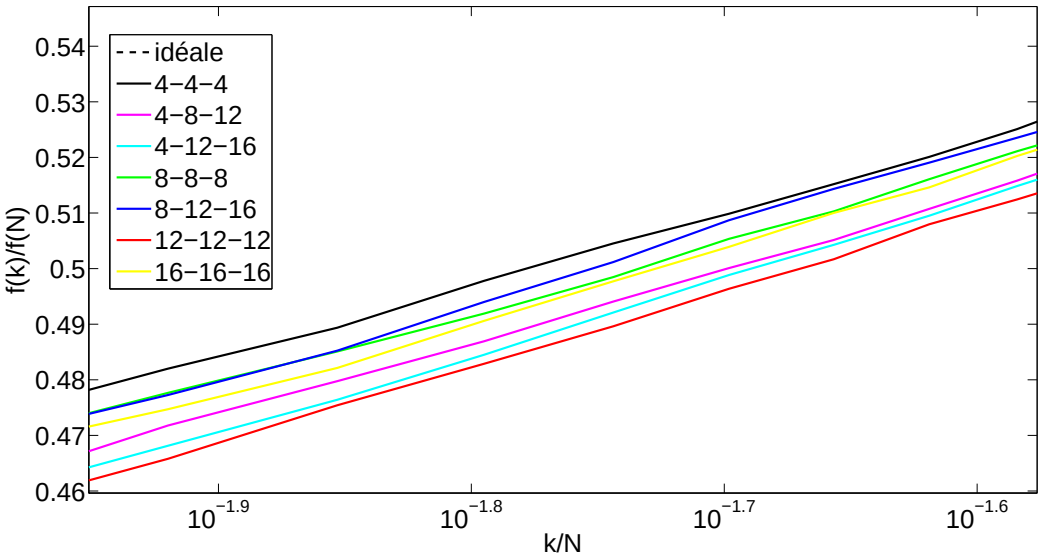
3.4.4 Influence du centrage du descripteur

Une première expérience simple donne une idée de l'impact de l'utilisation du centrage. Dans notre base de tests, l'avatar bouge les bras en gardant toujours la même position dans le repère 3D du monde. Sans connaître la position idéale que devrait avoir l'axe central du descripteur, on sait que si ce positionnement ne dépendait réellement pas de la posture, le descripteur devrait toujours être centré sur un même axe fixe. L'écart-type par rapport au positionnement moyen de l'axe dans la base d'exemples donne donc une idée de la stabilité du centrage. Le tableau 3.2 présente les écart-types sur les positionnements respectifs du centre de gravité et du centre de l'ellipsoïde sur les exemples de la base, avec les deux systèmes de caméras (4 ou 5 caméras).

Les courbes montrées sur la figure 3.20 donnent les résultats obtenus avec les deux types de centrage, pour les deux systèmes de caméras. Avec les 4 caméras, les deux courbes sont proches mais distinctes l'une de l'autre : la courbe pour laquelle le descripteur est centré sur l'ellipsoïde est en dessous, en particulier pour des valeurs de k faibles. Avec 5 caméras, les courbes sont quasiment confondues et se croisent. Ceci peut s'expliquer par le fait que les deux systèmes de caméras ne permettent pas de "creuser" les voxels des bras de la même manière. Avec le système de 4 caméras, la silhouette 3D présente généralement des artéfacts au niveau des bras ; les voxels supplémentaires déplacent plus fortement le centre de masse de la reconstruction en fonction de la pose des bras. La figure 3.21 donne un exemple des différences de positionnement du centre de gravité en fonction des artéfacts de la reconstruction 3D. Sur cette figure sont représentés les axes verticaux passant respectivement par le centre de gravité et le centre de l'ellipsoïde. Dans le cas particulier de cet exemple, les deux axes sont distants de 4 cm l'un de l'autre pour la reconstruction avec 4 caméras, tandis qu'ils ne sont écartés que de 1.5 cm



(a)



(b)

FIG. 3.19 – Courbes obtenues en faisant varier le nombre de divisions angulaires dans le descripteur.

a : Courbes. **b :** Zoom sur une partie de la figure.

lorsque la reconstruction est effectuée avec 5 caméras. On voit sur la figure que l'axe obtenu avec le centre de gravité est décentré par rapport au torse.

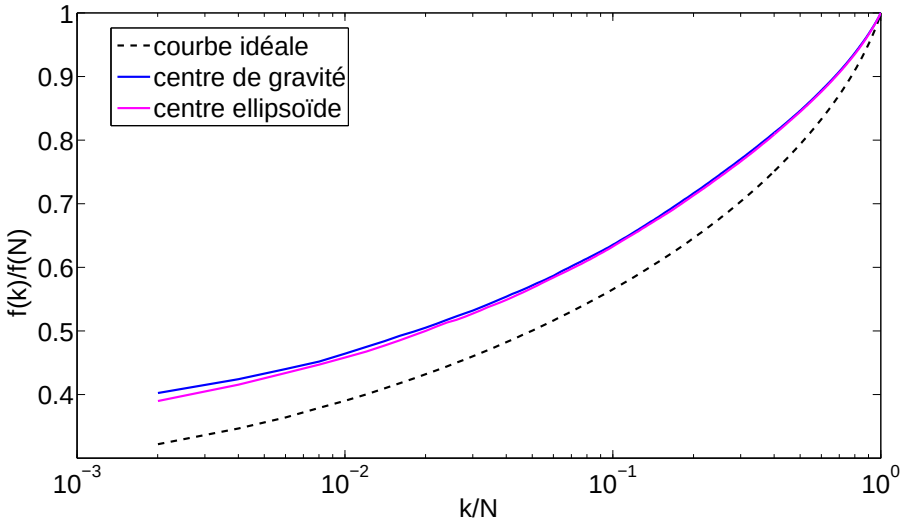
3.4.5 Sélection des composantes utiles

Dans cette expérience, nous souhaitons mettre en évidence l'intérêt de sélectionner les composantes pertinentes du descripteur pour estimer un paramètre particulier. La pose du bras droit a donc été estimée en ne retenant que des composantes du descripteur situées dans la partie supérieure droite du cylindre. Pour ne prendre en compte que les paramètres de la pose correspondant à ce bras, la distance de la formule 3.4 a été modifiée en ne gardant que les positions y_i^k des points du bras droit. Deux courbes sont comparées : celle qui est obtenue en utilisant un descripteur basé sur toutes les composantes de la moitié supérieure du cylindre, et celle qui est obtenue en s'appuyant uniquement sur les composantes situées dans la partie droite du cylindre. Les résultats sont présentés sur la figure 3.22. La sélection des composantes présente manifestement un intérêt.

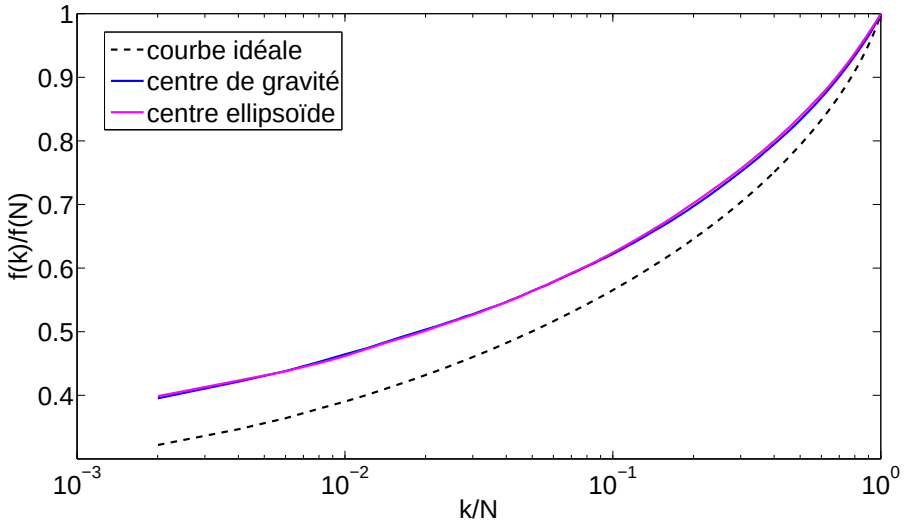
3.4.6 Test sur une séquence de marche

Pour représenter graphiquement la similitude entre l'espace de poses et l'espace du descripteur, les matrices des distances dans ces deux espaces ont été comparées sur une séquence de 418 exemples d'un avatar marchant en spirale (données issues de [4]). La matrice des distances se définit comme la matrice dont le terme général $a_{i,j}$ est la distance entre les éléments i et j de la séquence. Pour chaque exemple de la séquence, la silhouette 3D a été reconstruite avec le système de 5 caméras montré sur la figure 3.15, et le descripteur calculé. Les descripteurs sont comparés avec la distance euclidienne de $\mathbf{R}^n$, et la distance dans l'espace de pose celle de la formule 3.4 (les positions des points du corps sont exprimés par rapport à la hanche, qui est la racine de l'arbre cinématique). Les matrices obtenues sont représentées sur la figure 3.23 : la figure 3.23(a) donne la matrice des distances dans l'espace du descripteur et la figure 3.23(b) dans l'espace des poses. Cette figure donne une idée de l'information capturée par l'ensemble du processus de reconstruction 3D puis de calcul du descripteur.

L'avatar effectue trois tours de rayons décroissants. La similitude entre les postures prises par l'avatar à chaque tour sur sa trajectoire explique les trois bandes diagonales de valeurs plus faibles apparaissant dans les deux matrices. La similitude d'aspect entre les deux figures, notamment les bandes diago-

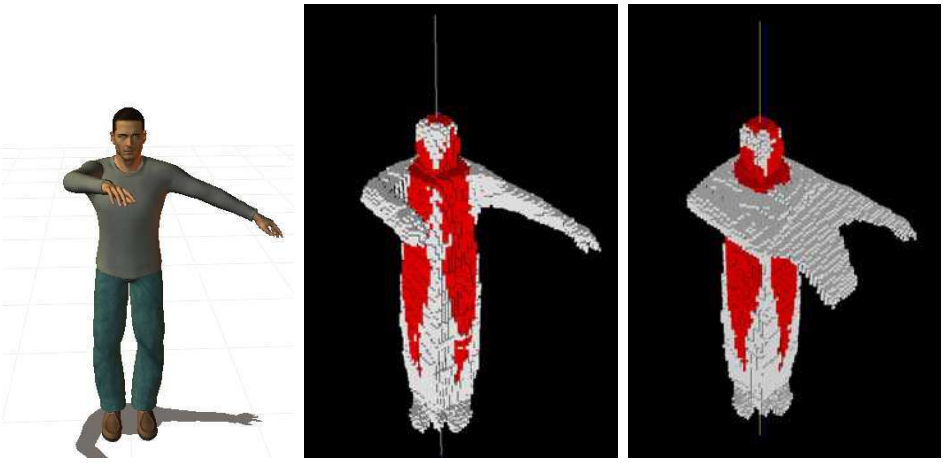


(a)



(b)

FIG. 3.20 – Comparaison des courbes obtenues en centrant le descripteur sur le centre de gravité de la reconstruction ou sur le centre de l'ellipsoïde. **a :** Courbes obtenues avec le système de 4 caméras. **b :** Courbes obtenues avec le système de 5 caméras.



(a) (b) (c)

FIG. 3.21 – Illustration du décalage du centre de gravité en fonction des artéfacts sur la silhouette 3D.
a : posture de l'avatar. **b :** reconstruction obtenue avec 5 caméras. **c :** reconstruction obtenue avec 4 caméras. Sur les deux dernières figures, les axes verticaux centrés sur le centre de gravité et l'ellipsoide sont représentés respectivement en bleu et en jaune.

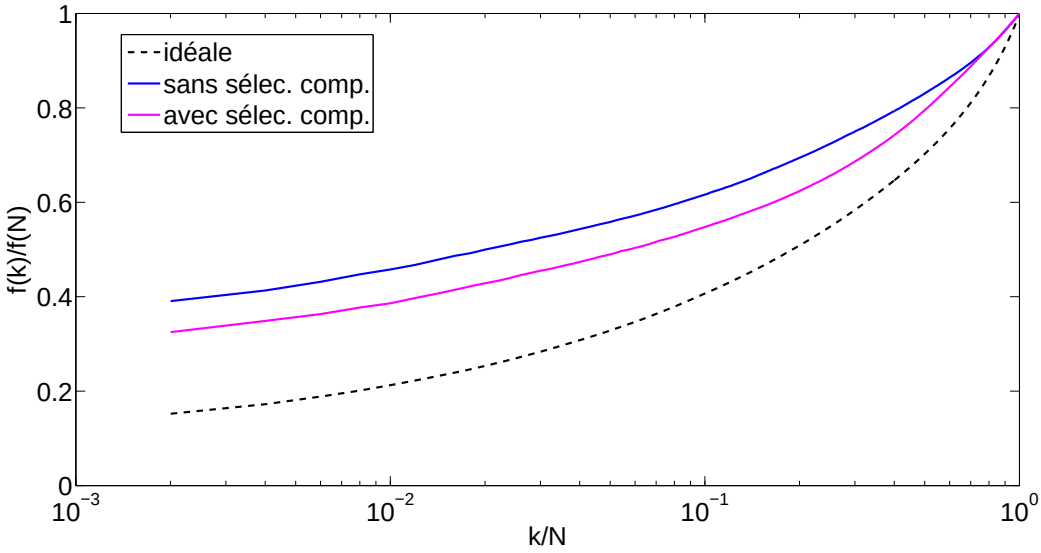
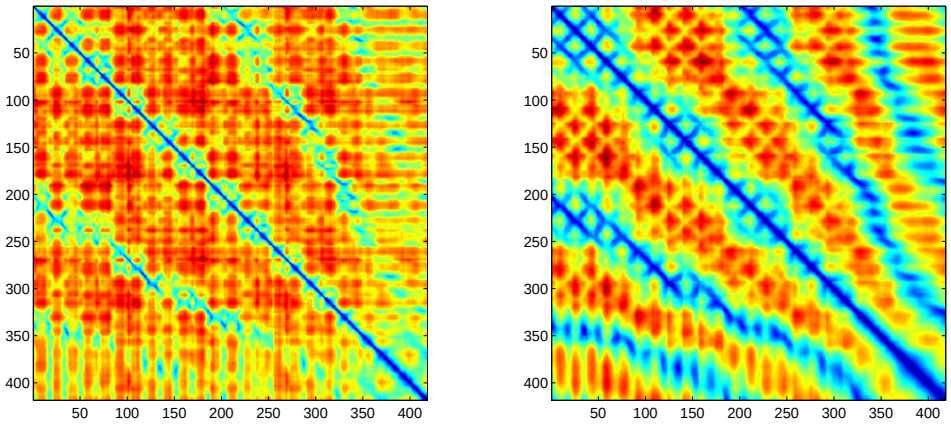


FIG. 3.22 – Courbes obtenues avec et sans sélection des composantes du descripteur pour estimer la pose d'un bras.



(a) (b)

FIG. 3.23 – Matrices des distances terme à terme sur une séquence de 418 images d'un avatar marchant selon une spirale décroissante.
a : matrice des distances du descripteur. **b :** matrice des distances des poses 3D.

nales indiquant la périodicité du mouvement, suggèrent que notre descripteur a bien capturé l'essentiel de l'information sur la pose.

3.5 Conclusion

Nous avons présenté dans cette partie un descripteur 3D basé sur un codage assez naturel de la répartition spatiale des voxels de la silhouette 3D autour d'un axe vertical : chaque composante représente la quantité de matière située dans une zone particulière du cylindre de référence.

Les expériences de ce chapitre ont montré de manière statistique l'influence des différents paramètres sur la qualité de la description, c'est-à-dire sa capacité à représenter de manière fidèle la structure de l'espace des poses du corps. Dans le chapitre 5, d'autres expériences portant cette fois sur l'ensemble du processus d'estimation (descripteur+régresseur) viendront compléter ces résultats et affiner le choix des paramètres optimaux. Dans le cas de l'estimation par régression, l'augmentation du nombre de divisions du descripteur et le gain en précision qu'elle permet d'obtenir, sont contrebalancées par le fait qu'une représentation de dimension trop élevée peut être pénalisée.

Chapitre 4

Estimation de la pose

4.1 Introduction

Cette partie décrit notre méthode d'estimation de la pose à partir du descripteur issu de la silhouette 3D. Notre méthode est basée sur des techniques d'apprentissage statistique : un apprentissage, réalisé hors ligne, permet de modéliser par une unique application la relation entre le descripteur 3D de l'enveloppe visuelle et le vecteur décrivant la pose. A l'issue de l'apprentissage, cette application génère directement une estimation de la pose à partir des données image encodées par le descripteur, sans passer par des prédictions sur la configuration d'un modèle du corps. Cette application résume en un modèle compact les propriétés des exemples de la base d'apprentissage pour effectuer des prédictions sur des données non-apprises. Cette technique permet donc d'intégrer des connaissances a priori sur la nature des mouvements réalisés grâce aux exemples de la base d'apprentissage : au lieu d'imposer des contraintes par l'intermédiaire d'un modèle détaillé du corps, celles-ci sont implicitement modélisées dans l'apprentissage. On est ainsi assuré, à l'estimation, de produire une pose valide, cohérente avec celles qui ont été utilisées dans les données d'entrainement. C'est aussi une des limitations de ces méthodes : l'estimation n'est valable que pour estimer des poses voisines de celles de la base, et un descripteur trop éloigné de ceux des exemples risque d'être associé à une estimation totalement erronée. Ce type d'approche présente enfin l'avantage de réduire les temps de calcul associés à l'estimation de la pose, car la plus grosse partie des calculs nécessaires est effectuée hors-ligne pendant la phase d'apprentissage. Nous verrons que différentes méthodes de régression peuvent être employées, qui permettent d'aboutir à une modélisation plus ou moins compacte et plus ou moins précise de cette application.

Modéliser la relation entre les données image et la pose par une application est un choix audacieux étant données la forte non-linéarité de la fonction apprise, les ambigüités visuelles et la grande dimension des vecteurs que l'on estime. Il n'est même pas garanti que la relation entre les entrées et les sorties puisse être modélisée par une unique application mono-valuée (voir 4.2.5). Nous verrons que dans certains cas, la régression ne nous permet pas d'accéder à une très grande précision sur la pose, en particulier pour la reconnaissance de mouvements complexes comme des gestes. Dans notre cas, la précision peut toutefois être améliorée en comparant les positions dans l'espace du squelette estimé et des voxels de l'enveloppe.

Ce chapitre est composé de trois parties. La première partie présente un état de l'art des différentes méthodes de régression qui peuvent être utilisées pour estimer la pose. La deuxième partie montre l'application de quelques unes de ces techniques à notre problème, et une comparaison expérimentale des performances de différentes méthodes de régression. La dernière partie détaille la méthode de raffinement que nous avons mise en place pour gagner en précision dans la reconnaissance de gestes.

4.2 Etat de l'art sur les méthodes de régression

Le but de l'apprentissage supervisé est de déterminer une fonction $t = y(x)$ permettant de modéliser la relation entre deux variables x et t, en se basant sur un ensemble d'apprentissage $D = \{(x_1, t_1), ..., (x_N, t_N)\}$, composé de mesures, généralement bruitées, des valeurs $t_1, ..., t_N$ de la fonction sur un ensemble de points d'entrainement $x_1, ..., x_N$. Cet apprentissage doit permettre de prédire la valeur de la fonction en des points arbitraires qui n'étaient pas présents dans la base d'entrainement.
Les données d'entrée sont généralement des vecteurs de dimension d (i.e. $x_i \in \mathbf{R}^d$). Dans un problème de régression, la variable de sortie y est continue (i.e. $y \in \mathbf{R}$ ou $\mathbf{y} \in \mathbf{R}^p$ si la sortie a plusieurs dimensions). En classification, le but est d'attribuer une étiquette à une entrée donnée, y désigne alors l'appartenance à une classe (par exemple $y \in \{-1, 1\}$).
La fonction obtenue par apprentissage est évaluée sur sa capacité à bien se généraliser, c'est-à-dire à bien expliquer (prédire les bonnes valeurs) de nouvelles données non-apprises qui suivent la même distribution que les données d'apprentissage.

Les fonctions recherchées sont souvent supposées avoir une structure particulière fixée ; elles dépendent alors d'un certain nombre de paramètres que l'on cherche à évaluer en se basant sur les données d'apprentissage. Par exemple, dans le cas des modèles linéaires (voir paragraphe 4.2.1), la fonction est modélisée par une combinaison linéaire de fonctions de base, et l'apprentissage sert à déterminer les coefficients de cette combinaison linéaire. Si la fonction est modélisée par un réseau de neurones (4.2.4), l'apprentissage permettra de déterminer les valeurs des poids synaptiques.

Problème de la complexité des modèles

Pour que la fonction estimée par apprentissage se généralise bien à des données non-apprises, il faut contrôler sa complexité. En effet, si cette fonction est trop complexe (qu'elle possède trop de degrés de liberté), elle peut "coller" à des caractéristiques non-pertinentes, comme du bruit ou un biais, propres à l'ensemble sur lequel elle a été apprise. La fonction a tendance à se comporter comme une table contenant tous les échantillons utilisés lors de l'apprentissage et perd ses pouvoirs de prédiction sur de nouveaux échantillons. C'est le problème du surapprentissage (*overfitting* en anglais). Inversement, une fonction trop simple peut ne pas être assez riche pour capturer la véritable relation entre les données d'entrée et de sortie. Il s'agit cette fois d'un problème de sous apprentissage (*underfitting*). Une des difficultés de l'apprentissage statistique est de trouver un juste équilibre sur la complexité du modèle.

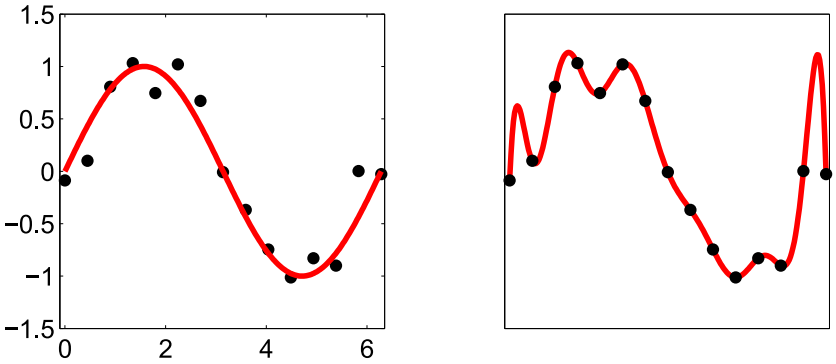


FIG. 4.1 – Illustration du surapprentissage (figure extraite de [114]).

Les deux graphes de la figure 4.1 montrent deux exemples de fonctions qui ont été générées à partir des exemples d'entrainement représentés par

les points noirs. Même si on ne peut pas vraiment a priori porter de jugement sur la qualité de l'une ou l'autre des deux régressions, puisqu'on ne connait pas la nature réelle de la fonction qui a servi à générer les données, on a naturellement envie de privilégier la fonction de gauche. Ce modèle semble en effet mieux tenir compte de la présence de bruit dans les données d'apprentissage, tandis que la fonction de droite colle parfaitement avec les données d'entrainement mais semble avoir moins de chance de bien se généraliser sur de nouveaux échantillons. D'une façon générale, entre deux modèles qui expliquent aussi bien les données d'apprentissage, on a souvent intérêt à privilégier le modèle le plus simple, car il aura tendance à mieux se généraliser. Ainsi, nous verrons que lors de la construction d'un modèle de régression, la préférence est toujours portée a priori sur les modèles les plus simples (principe du rasoir d'Ockham).

Problème de la dimension des données d'apprentissage

Dans le chapitre précédent, nous avons insisté sur le fait qu'il fallait limiter la dimension des données d'entrée dans la machine d'apprentissage. Cette idée peut se justifier intuitivement par le fait que, pour estimer une valeur de sortie $f(x)$, l'algorithme d'apprentissage se base sur la structure du voisinage de l'entrée x. L'application apprise s'appuiera en effet sur les éléments de la base qui ont des entrées similaires à x pour prédire la valeur de la sortie. Plus la dimension est grande, plus le voisinage de x est complexe et son volume important, et plus le nombre de données d'apprentissage nécessaires pour bien connaître sa structure est important. La figure 4.2 donne une petite illustration de ces problèmes : si on souhaitait découper l'espace d'apprentissage en cellules espacées régulièrement, le nombre de cellules nécessaires grandirait de façon exponentielle avec la dimension de l'espace. Pour échantillonner correctement l'espace d'apprentissage, il faudrait que toutes les cellules soient remplies par des données d'apprentissage ; le nombre d'exemples devrait donc augmenter exponentiellement en fonction de la dimension des données d'entrée.

4.2.1 Modèles linéaires

Les modèles linéaires sont des modèles fréquemment utilisés dans les problèmes de régression. La fonction recherchée est supposée pouvoir s'exprimer

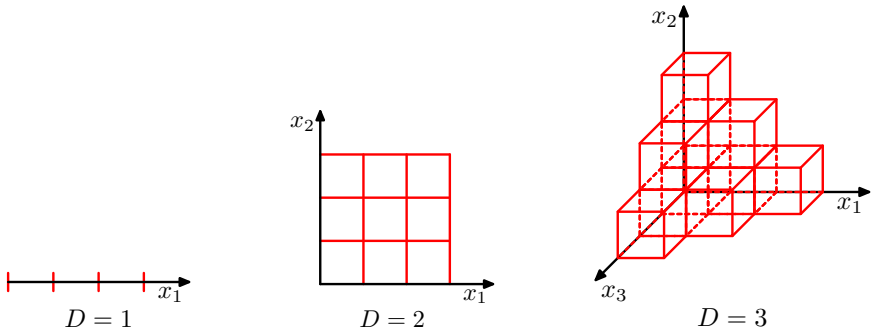


FIG. 4.2 – Illustration des problèmes posés par l'augmentation de la dimension des données (figure extraite de [13]).

comme une combinaison linéaire de M fonctions de base $\phi_m(x)$ fixées :

$$y(x, w) = w_0 + \sum_{m=1}^{M} w_m \phi_m(x), \tag{4.1}$$

Le terme (facultatif) w_0 est un terme de *biais* permettant de prendre en compte un éventuel offset dans les données (pour faciliter les notations, on introduit une fonction de base "artificielle" ϕ_0 telle que $\phi_0 = 1$).
En notant $\mathbf{w} = (w_0, ..., w_M)^T$ et $\phi = (\phi_0, ..., \phi_M)^T$, la formule précédente se récrit :

$$y(x, w) = \mathbf{w}^T.\phi(x) \tag{4.2}$$

Cette hypothèse n'implique pas forcément que la sortie y est une fonction linéaire des données d'entrée x (comme pourrait le laisser penser l'expression "modèle linéaire"). La possibilité d'utiliser comme fonctions de base $\phi_m(x)$ des fonctions non linéaires permet au contraire de modéliser une relation non linéaire entre x et y. En revanche, la fonction y est linéaire par rapport aux paramètres que l'on veut estimer, c'est-à-dire les poids w. Cette linéarité facilite l'analyse de ces modèles, en permettant d'avoir une solution analytique à la recherche de valeur optimale des poids au sens des moindres carrés.
La forme la plus simple de modèle linéaire est celle pour laquelle les fonctions de base sont linéaires, c'est-à-dire $\phi(x) = x$. Dans ce cas particulier, la sortie y est modélisée par une combinaison linéaire des différentes dimensions du vecteur d'entrée :

$$y(x, w) = w_0 + w_1 x_1 + ... + w_D x_D \tag{4.3}$$

Il existe de nombreuses possibilités pour les fonctions de base. Un choix

classique est l'utilisation de noyaux gaussiens centrés sur des exemples d'apprentissage :

$$\phi_m(x) = K(x, x_m) = e^{-\frac{\|x - x_m\|^2}{2\sigma^2}} \tag{4.4}$$

Approximation aux moindres carrés

L'objectif de la phase d'apprentissage est de déterminer les valeurs des poids w tels que la fonction $y(x, w)$ modélise au mieux la fonction qui a servi à générer les données. Une approche classique consiste à chercher la valeur des poids qui minimise l'erreur au sens des moindres carrés sur les données d'apprentissage :

$$E(w) = \sum_{n=1}^{N} \|t_n - \sum_{m=0}^{M} w_m \phi_m(x_n)\|^2 \tag{4.5}$$

En notant $\mathbf{t} = (t_1, ..., t_n)^T$ le vecteur réunissant l'ensemble des sorties des données d'apprentissage et $\mathbf{\Phi}$ la matrice des noyaux définie par $\Phi_{mn} = \phi_m(x_n)$,

$$\mathbf{\Phi} = \begin{pmatrix} \phi_0(x_1) & \phi_1(x_1) & \dots & \phi_M(x_1) \\ \phi_0(x_2) & \phi_1(x_2) & \dots & \phi_M(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_N) & \phi_1(x_N) & \dots & \phi_M(x_N) \end{pmatrix} \tag{4.6}$$

l'équation précédente peut se récrire :

$$E(w) = \|\mathbf{t} - \mathbf{\Phi}\mathbf{w}\|^2 \tag{4.7}$$

et sa solution est donnée de manière analytique par l'équation :

$$\mathbf{w}_{LS} = \mathbf{\Phi}^{\dagger}\mathbf{t} \tag{4.8}$$

où la matrice

$$\mathbf{\Phi}^{\dagger} = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T \tag{4.9}$$

est la pseudo-inverse de la matrice $\mathbf{\Phi}$.

Régularisation

Dans la plupart des applications, la matrice $\mathbf{\Phi}^T\mathbf{\Phi}$ est mal-conditionnée, et son inversion est instable. La résolution telle quelle du système aux moindres carrés risque alors de produire un surapprentissage. Pour pallier ce problème, on introduit souvent une contrainte supplémentaire sur les poids du modèle,

de façon à privilégier les fonctions les plus lisses. Dans l'hypothèse d'un modèle linéaire, les fonctions les plus lisses sont les fonctions dont les poids ont des amplitudes plus faibles. On introduit donc dans la fonction de coût à minimiser un terme $R(\mathbf{w})$ pénalisant les poids élevés :

$$E_{reg}(w) = E(w) + \lambda R(w) \tag{4.10}$$

Le paramètre λ permet d'établir un compromis sur la complexité de la fonction estimée : plus sa valeur est faible, plus la fonction "colle" aux données d'apprentissage (on se ramène au cas précédent lorsque $\lambda \to 0$), plus sa valeur est élevée, plus la fonction estimée est lisse. Un choix classique pour le terme de régularisation est $R(\mathbf{w}) = \|\mathbf{w}\|^2$. On parle alors de *Ridge Regression* ou *Damped Least Squares* (moindres carrés amortis). Ce type de régression est utilisé par Agarwal et Triggs pour l'estimation de pose dans [7]. En posant $\tilde{\mathbf{\Phi}} = (\mathbf{\Phi}\ \lambda I)$ et $\tilde{\mathbf{t}} = (\mathbf{t}\ 0)$, les poids recherchés doivent minimiser le critère :

$$\tilde{E}(\mathbf{w}) = \|\tilde{\mathbf{t}} - \mathbf{w}.\tilde{\mathbf{\Phi}}\|^2 = \|\mathbf{t} - \mathbf{w}.\Phi\|^2 + \lambda\|\mathbf{w}\|^2 \tag{4.11}$$

ce qui revient à résoudre le système linéaire $\mathbf{W}\tilde{\mathbf{\Phi}} = \tilde{\mathbf{t}}$ aux moindres carrés. La solution est donc aussi obtenue sous forme analytique :

$$\mathbf{w}_{PLS} = (\mathbf{\Phi}^T\mathbf{\Phi} + \lambda I)^{-1}\mathbf{\Phi}^T\mathbf{t} \tag{4.12}$$

L'ajout d'un terme de régularisation en $\|\mathbf{w}\|^2$ dans le critère de minimisation implique que toutes les solutions ne sont pas équivalentes par rapport à un changement d'échelle des données d'apprentissage, donc il faut normaliser les données d'entrée et de sortie avant de résoudre (c'est-à-dire multiplier les données par une facteur d'échelle pour se ramener à des données ayant une moyenne nulle et une variance unité).

Formulation probabiliste

Les résultats qui précèdent peuvent être reformulés dans un cadre probabiliste. On introduit ici les notions qui seront reprises dans le paragraphe 4.2.3 sur les RVM. La formulation bayésienne du problème consiste à modéliser toutes les quantités du système avec des densités de probabilité. On représente ainsi par des probabilités le bruit dans les données d'apprentissage, les sources d'incertitudes du problème, et les contraintes a priori que l'on souhaite imposer à certaines quantités.

Les données d'entraînement peuvent être considérées comme des réalisations bruitées de la fonction y qu'on cherche en posant

$$t_n = y(x_n, w) + \epsilon_n \tag{4.13}$$

où ϵ_n suit une loi gaussienne de moyenne 0 et de variance σ^2 :

$$p(\epsilon_n|\sigma^2) = N(\epsilon_n|0, \sigma^2) \tag{4.14}$$

On peut alors facilement montrer (voir [114]) que maximiser la vraisemblance des données d'apprentissage, c'est-à-dire la probabilité $p(t|w, \sigma^2)$, par rapport au vecteur des poids revient à minimiser l'erreur d'apprentissage aux moindres carrés (équation 4.5).

La préférence pour des valeurs faibles des poids peut aussi être formulée de manière probabiliste en spécifiant que les poids sont répartis suivant une distribution gaussienne centrée en 0 :

$$p(w|\alpha) = N(w|0, \alpha^{-1}I) \tag{4.15}$$

où α est un paramètre sur la variance à ajuster en fonction du problème.

La recherche des poids maximisant la probabilité a posteriori

$$p(w|t, \alpha, \sigma^2) = \frac{p(t|w, \sigma^2)p(w|\alpha)}{p(t|\alpha, \sigma^2)} \tag{4.16}$$

revient à résoudre le système pénalisé de la formule 4.10 en posant $\lambda = \sigma^2\alpha$. Modéliser la densité des poids par une gaussienne centrée en 0 est donc équivalent à ajouter un terme de pénalisation sur les poids les plus forts, avec un coefficient de pénalisation inversement proportionnel à l'écart-type de la gaussienne.

Sorties à plusieurs dimensions

On a considéré jusqu'ici le cas où les données de sortie sont mono dimensionnelles ($y \in \mathbf{R}$). Or, dans un problèle comme l'estimation de la pose, on a besoin d'estimer des vecteurs à plusieurs dimensions (la dimension égale le nombre de paramètres dans la modélisation choisie pour le corps humain). Tout ce qui précède peut facilement être généralisé au cas où la variable de sortie y est remplacée par un vecteur $\mathbf{y} \in \mathbf{R}^d$.

Pour traiter le cas de sorties multi-dimensionnelles, une solution possible est d'estimer chaque dimension séparément par des régressions indépendantes, en introduisant différents ensembles de fonctions de base pour chaque dimension du vecteur $\mathbf{y}$. Une alternative plus intéressante est d'utiliser le même ensemble de fonctions de base pour traiter toutes les composantes du vecteur de sortie. La formule 4.2 devient :

$$\mathbf{y}(\mathbf{x}, \mathbf{w}) = \mathbf{W}^T\phi(x) \tag{4.17}$$

où $\mathbf{W}$ est une matrice de taille $M \times d$ et $\phi(x)$ est un vecteur colonne de dimension M composé des éléments $\phi_j(x)$.

Comme précédemment, la solution optimale aux moindres carrés peut être obtenue sous forme analytique :

$$\mathbf{W}_{LS} = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{T} \tag{4.18}$$

où $\mathbf{T}$ est une matrice de taille $N \times d$ réunissant l'ensemble des vecteurs d'observation $\mathbf{t}_1, ..., \mathbf{t}_N$ (la n^e colonne de $\mathbf{T}$ est $\mathbf{t}_n$).

En ajoutant un terme de régularisation $R(\mathbf{W}) = \lambda\|\mathbf{W}\|^2$, la solution est obtenue de la même façon que dans 4.2.1 en résolvant le système linéaire $\mathbf{W}\tilde{\Phi} = \tilde{\tilde{\mathbf{T}}}$ avec $\tilde{\Phi} = (\Phi\ \lambda I)$ et $\tilde{\mathbf{T}} = (\mathbf{T}\ 0)$.

Sélection des fonctions de base et limites des modèles linéaires

On a vu dans les paragraphes précédents que la préférence est a priori portée sur les modèles les plus simples, ce qui implique de privilégier les fonctions les plus lisses en ajoutant par exemple un facteur pénalisant les fonctions avec des variations brusques. Une autre propriété désirable est le caractère "épars" des modèles, c'est-à-dire qu'on cherche également à privilégier les représentations contenant un petit nombre de fonctions de base. Cela revient à fixer les poids associés à la plupart des fonctions de base à 0. La compacité du modèle est une propriété avantageuse pour plusieurs raisons. Premièrement, ces modèles, plus simples, ont tendance à mieux se généraliser à des données non apprises. Deuxièmement, le faible nombre de fonctions de base dans le modèle implique que l'estimation sera moins coûteuse et donc plus rapide. Cette réduction du temps de calcul est particulièrement intéressante si l'on vise des applications temps-réel comme dans notre cas. Enfin, comme il sera montré dans la suite, ces méthodes fournissent des mécanismes de sélection de caractéristiques ou d'exemples pertinents, qui peuvent être utiles dans certaines applications.

Une limitation des modèles linéaires est le manque de contrôle sur la sélection des fonctions de base sur lesquelles s'appuient les modèles : les fonctions de base sont fixées à l'avance avant que les données d'entrainement n'aient été observées, leur choix ne s'adapte pas à la structure des données. On n'a a priori aucun moyen de sélectionner parmi les exemples ceux qu'il serait pertinent de retenir comme support des fonctions de base. Nous verrons dans la

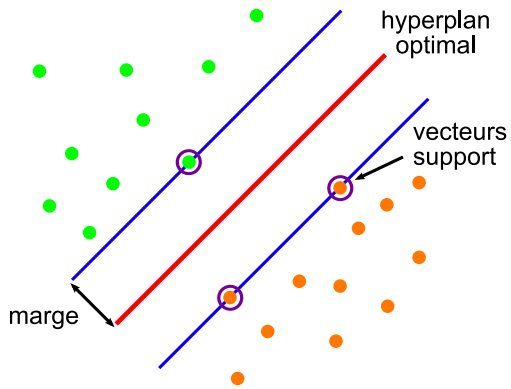


FIG. 4.3 – L'hyperplan optimal avec la marge maximale.

partie expérimentale que de bons résultats peuvent tout de même être obtenus en sélectionnant aléatoirement une fraction des exemples d'apprentissage. Ces limitations motivent l'utilisation de modèles plus complexes, comme les réseaux de neurones, les SVM et les RVM. Nous verrons dans la suite que les SVM et les RVM font partie des méthodes qui permettent d'aboutir à un modèle épars.

4.2.2 Support Vector Machines

SVM pour la classification

Les Support Vector Machines (traduit en français par *Séparateurs à Vastes Marges* ou *Machines à Vecteurs de Support*), sont principalement connus pour les problèmes de classification. Les classifieurs SVM reposent sur la notion clé de *marge maximale*. Dans le cas simple d'un problème de classification entre deux ensembles linéairement séparables, il existe une infinité d'hyperplans séparateurs dont les performances en apprentissage sont identiques, mais dont les performances en généralisation peuvent être très différentes. Il a été montré qu'il existe un unique hyperplan optimal : celui qui maximise la marge entre l'hyperplan et les échantillons les plus proches (voir figure 4.3). Ces derniers sont appelés *vecteurs support*.

Dans le cas linéaire, la fonction de décision s'écrit

$$y(x) = \langle w, x \rangle + b \tag{4.19}$$

Le fait que les points de la base d'entrainement sont correctement classés s'exprime pour tout $n \in \{1, ..., N\}$ par $t_n y(x_n) > 0$, où t_n est le label du point x_n ($t_n \in \{-1, 1\}$). La marge est définie comme la distance à l'hyperplan du

point de l'ensemble d'apprentissage le plus proche de l'hyperplan. La solution maximisant la marge est cherchée en résolvant :

$$\arg\max_{w,b}\left\{\frac{1}{\|w\|}\min_{n}\left[t_n(\langle w,x_n\rangle+b)\right]\right\} \tag{4.20}$$

En multipliant w et b par un facteur multiplicatif, on peut montrer qu'il est équivalent de résoudre

$$\arg\min_{w,b}\frac{1}{2}\|w\|^2 \tag{4.21}$$

sous les contraintes

$$t_n(\langle w,x_n\rangle+b)\geq 1 \qquad \text{pour tout } n\in\{1,...,N\} \tag{4.22}$$

La formulation des SVM permet donc de ramener le problème de classification à un problème d'optimisation sous contraintes d'une fonction convexe. C'est l'un des gros avantages de cette méthode, car elle assure de trouver une solution globale au problème de classification.
Le problème d'optimisation sous contraintes est résolu en introduisant les multiplicateurs de Lagrange. L'équation de l'hyperplan optimal s'écrit :

$$y(x)=\sum_{k=1}^{p}\alpha_k^* t_k\,\langle x,x_k\rangle+b \tag{4.23}$$

où les α_k^* sont les multiplicateurs de Lagrange optimaux. Dans cette équation, les seuls points dont les coefficients α_k^* sont non nuls sont les points situés sur les hyperplans de marge maximale : seuls les vecteurs support participent à la définition de l'hyperplan optimal. Seul un sous-ensemble restreint de points est nécessaire pour le calcul de la solution, les autres échantillons ne participent pas du tout à sa définition. C'est la clé de la compacité de ces modèles. Ajouter de nouveaux exemples d'apprentissage n'a pas d'influence sur la complexité du modèle s'ils ne sont pas des vecteurs support.

Dans le cas où il n'existe pas de séparation linéaire entre les deux classes, des fonctions noyaux permettent de considérer des frontières de séparation plus complexes d'un simple hyperplan. D'une façon générale, l'utilisation de fonctions noyaux permet de transformer une technique linéaire en une technique non linéaire (modèles linéaires, kernel PCA...). Dans le cas des SVM, elles peuvent être vues comme un moyen de ramener le problème dans un espace de redescription où les données sont linéairement séparables. On applique aux vecteurs de description une transformation non linéaire ϕ qui

permet de ramener le problème dans un espace de dimension plus grande dans lequel les données peuvent être séparées par un hyperplan. Comme dans l'équation de l'hyperplan optimal, les vecteurs d'entrée n'apparaissent que sous forme de produits scalaires $\langle x, x_i \rangle$, il n'est pas nécessaire d'expliciter l'application ϕ, mais simplement de définir une fonction noyau K telle que $K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$.

Dans la plupart des cas, il n'est pas non plus possible de trouver une séparation linéaire entre les deux classes, même en se ramenant à un autre espace de dimension plus grande. Pour généraliser la formulation des SVM au cas non-séparable, des variables ressort ξ (*slack variables* en anglais) sont introduites pour relâcher les contraintes sur les vecteurs d'apprentissage en tolérant les mauvais classements :

$$t_n y(x_n) \geq 1 - \xi_n \tag{4.24}$$

où les variables ressort satisfont $\xi_n \geq 0$. Le but est cette fois de maximiser la marge en pénalisant les points situés du mauvais côté de la frontière. On minimise donc

$$C \sum_{n=1}^{N} \xi_n + \frac{1}{2} \|w\|^2 \tag{4.25}$$

sous les contraintes

$$\xi_n \geq 0 \qquad \text{pour tout } n \in \{1, ..., N\} \tag{4.26}$$

Le paramètre $C > 0$ contrôle les compromis qui est fait entre la pénalité sur les variables mal classées et la marge. Ce paramètre est à rapprocher de la constante λ dans les modèles linéaires : une valeur élevée de la constante C fait "coller" le modèle aux données d'apprentissage en pénalisant plus fortement les données mal classées, alors qu'une valeur faible privilégie les modèles avec une marge élevée susceptibles d'avoir une meilleure généralisation. Ce paramètre est généralement fixé par validation croisée.

SVM pour la régression

Les SVM peuvent être étendus à des problèmes de régression (*Support Vector Regression*) tout en conservant leurs propriétés de compacité. Dans le cas d'une simple régression linéaire, on cherche une fonction de la forme

$$y(x) = \langle w, x \rangle + b \tag{4.27}$$

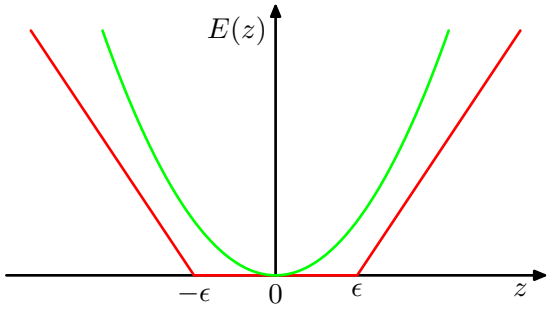


FIG. 4.4 – Fonctions d'erreur quadratique (en vert) et fonction $\epsilon-$*insensible* (en rouge) utilisée pour la régression par SVM (figure extraite de [13]).

où $x \in \mathbf{R}^d$ et $b \in \mathbf{R}$, qui minimise la fonction de coût donnée par :

$$\frac{1}{2}\sum_{n=1}^{N}(y_n - t_n)^2 + \frac{\lambda}{2}\|w\|^2 \tag{4.28}$$

Pour obtenir des solutions éparses, l'erreur quadratique $(y(x)-t)^2$ est remplacée par une fonction d'erreur $\epsilon - insensible$ donnée par :

$$E_\epsilon(y(x)-t) = \begin{cases} 0, & \text{si } |y(x)-t| < \epsilon \\ |y(x)-t| - \epsilon, & \text{sinon} \end{cases}$$

La fonction E_ϵ est représentée sur la figure 4.4. Une erreur n'a pas de poids dans la fonction de coût tant qu'elle est plus petite que ϵ, mais les erreurs supérieures à ϵ sont pénalisées. Il faut donc minimiser la fonction d'erreur donnée par

$$C\sum_{n=1}^{N} E_\epsilon(y(x_n) - t_n) + \frac{1}{2}\|w\|^2 \tag{4.29}$$

De manière analogue à la classification, on introduit des variables ressort $\xi_n \geq 0$ et $\hat{\xi}_n \geq 0$ permettant aux données d'être en dehors du tube de largeur ϵ (au dessus pour $\xi_n > 0$ et en dessous pour $\hat{\xi}_n > 0$) :

$$t_n \leq y(x_n) + \epsilon + \xi_n \tag{4.30}$$
$$t_n \geq y(x_n) - \epsilon - \hat{\xi}_n \tag{4.31}$$

Le problème de régression se ramène à minimiser

$$C\sum_{n=1}^{N}(\xi_n + \hat{\xi}_n) + \frac{1}{2}\|w\|^2 \tag{4.32}$$

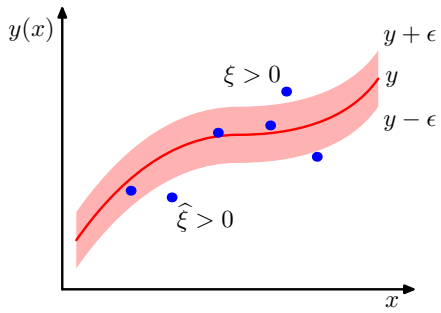


FIG. 4.5 – Illustration d'une régression SVM (figure extraite de [13]).

sous les contraintes

$$\xi_n \geq 0 \quad \text{et} \quad \hat{\xi}_n \geq 0 \qquad \text{pour tout } n \in \{1, ..., N\} \tag{4.33}$$

Les seuls vecteurs qui contribuent aux prédictions sont les points situés à la frontière du tube de largeur ϵ ou en dehors du tube. Tous les points situés à l'intérieur du tube ne participent pas à la solution, ce qui permet encore une fois d'aboutir à un modèle assez compact.

De la même façon que pour les problèmes de classification, l'utilisation de fonctions noyaux permet de traiter la régression avec des classes de fonctions plus générales de la forme :

$$y(x) = \sum_{k=1}^{p} w_k K(x, x_k) + b \tag{4.34}$$

Réglage des hyperparamètres

Dans la méthode SVM, différents paramètres apparaissent et leur choix a des implications à la fois sur le nombre de vecteurs support et la précision des résultats. Le paramètre C représente le compromis entre la complexité du modèle (qui joue sur son pouvoir de généralisation) et l'erreur sur les données d'apprentissage. En régression, la largeur du tube d'insensibilité ϵ peut être réglée en considérant le niveau de bruit dans les données : ce paramètre détermine le niveau d'erreur qui ne sera pas prise en compte dans la fonction de coût.
Ces différents paramètres sont généralement fixés par validation croisée, et ces tests préliminaires représentent l'un des inconvénients de cette méthode.

4.2.3 Relevance Vector Machines

Les modèles RVM [113] sont basés sur la formulation bayésienne d'un modèle linéaire, mais dont la distribution a priori sur les poids a été modifiée pour aboutir à une sélection automatique des éléments pertinents de la base d'apprentissage. Ce processus permet d'aboutir au final à un modèle particulièrement compact.

L'inférence bayésienne consiste à définir des densités de probabilité pour modéliser toutes les quantités du système. Les distributions de probabilité sont mises à jour par la règle de Bayes et toutes les variables qui ne nous intéressent pas directement sont intégrées (principe de marginalisation).

On reprend ici les notations de la partie 4.2.1 : le bruit dans les données est modélisé selon l'équation 4.13. La clé de la compacité des modèles RVM réside dans le choix de la densité a priori sur les poids du modèle linéaire suivant le principe d'*Automatic Relevance Determination* (ARD). L'ARD est une approche bayésienne hiérarchique, dans laquelle des hyperparamètres contrôlent les amplitudes des intervalles dans lesquels varient les composantes des données d'entrée. Par exemple, dans le cas où les quantités sont modélisées avec des distributions gaussiennes centrées en 0, ces hyperparamètres contrôlent l'écart-type des gaussiennes. Si l'écart-type de la gaussienne tend vers 0, le paramètre correspondant est contraint à rester proche de 0, et ne peut pas avoir d'effet sur la prédiction, ce qui rend sa présence dans le modèle superflue.

Ainsi, alors que dans la formule 4.15 toutes les composantes du vecteur des poids partagent le même hyperparamètre α, on introduit dans les modèles RVM un hyperparamètre α_i différent (et indépendant des autres paramètres) sur chaque composante du vecteur des poids w_i :

$$p(w/\alpha) = \prod_{i=1}^{M} N(w_i|0, \alpha_i^{-1}) \tag{4.35}$$

où $\alpha = (\alpha_1, ..., \alpha_M)^T$ et α_i représente la précision du paramètre de poids w_i.

Contrairement à l'estimation MAP des modèles linéaires présentée dans le paragraphe 4.2.1, l'inférence bayésienne n'estime pas la valeur optimale du vecteur des poids, mais cherche à évaluer la forme de sa densité de probabilité. Dans la phase d'apprentissage, les paramètres optimaux du modèle sont obtenus en maximisant la vraisemblance marginale des données (c'est-à-dire la vraisemblance marginalisée par rapport aux poids) par rapport aux

paramètres α et σ (approximation *maximum de vraisemblance de type II*) :

$$p(t|\alpha, \sigma^2) = \int p(t|w, \sigma^2)p(w|\alpha)dw \tag{4.36}$$

Dans le processus d'optimisation, une large proportion des hyperparamètres α_i devient très grande (les paramètres tendent vers l'infini), et les poids correspondant ont une moyenne et une variance nulles. Ces paramètres, ainsi que les fonctions de base $\phi_i(x)$ associées, sont éliminés du modèle et ne joueront donc aucun rôle dans la prédiction. Dans le cas d'un modèle basé sur des fonctions noyaux centrées sur des données d'apprentissage $K(x, x_n)$, les données d'entrainement x_n correspondant aux poids non nuls sont appelées *Relevance Vectors* (vecteurs de pertinence) : ce sont les vecteurs qui ont été identifiés par le mécanisme d'ARD comme "pertinents" dans le modèle linéaire. A noter qu'à l'issue du processus d'apprentissage, on obtient aussi une estimation du taux de bruit σ dans les données d'apprentissage.

Une fois obtenues les valeurs optimales α^* et σ^* des hyperparamètres, il est possible d'évaluer la distribution de la sortie t pour un nouveau vecteur d'entrée x. La prédiction est donnée par :

$$p(t|x, \alpha^*, \sigma^*) = N(t|\mu(x), \tilde{\sigma}(x)) \tag{4.37}$$

avec

$$\mu(x) = \mathbf{m}^T\phi(x) \tag{4.38}$$

où $\mathbf{m} = (\Phi^T\Phi + A)^{-1}\Phi^T t$ et A est la matrice diagonale contenant les hyperparamètres (ceux qui ne tendent pas vers l'infini dans l'optimisation) α_i et

$$\tilde{\sigma}^2(x) = (\sigma^*)^2 + \phi(x)^T\Sigma\phi(x) \tag{4.39}$$

où $\Sigma = (A + \sigma^{*-2}\Phi^T\Phi)^{-1}$.
La valeur moyenne $\mathbf{m}$ du vecteur des poids est à rapprocher de formule 4.12 obtenue dans le cas des modèles linéaires : les hyperparamètres peuvent être vus comme des coefficients de régularisation suivant les différentes dimensions de la matrice des noyaux Φ.

Outre le fait qu'ils produisent un modèle particulièrement épars, l'un des avantages des RVM est qu'ils permettent de prédire de la valeur de la sortie sous la forme d'une distribution de probabilité. Cette distribution peut être utilisée pour avoir une évaluation de l'incertitude de la valeur prédite. Dans [122], cette probabilité est mise à profit pour des applications de suivi d'objets dans des images. On peut toutefois constater dans la formule 4.39 que la variance de la prédiction peut avoir un comportement paradoxal : sa

valeur devient faible dans les régions de l'espace où les fonctions de base $\phi_i(x)$ ont des valeurs faibles. Dans le cas où les fonctions de base sont des fonctions noyaux centrées sur des données d'apprentissage, cela implique que la certitude des prédictions deviendra plus grande si on extrapole en dehors du domaine des données d'apprentissage.

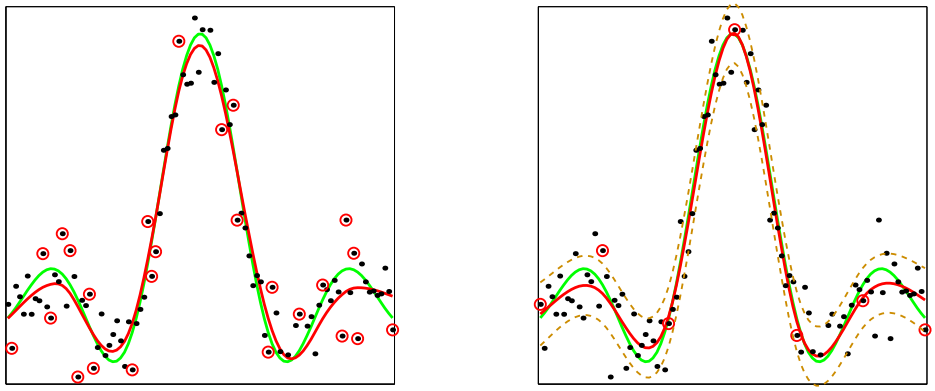
Comparaison avec SVM

Les SVM et RVM ont certaines similitudes. Les deux méthodes permettent de construire un modèle linéaire épars, c'est-à-dire dans lequel une petite fraction des exemples d'apprentissage est sélectionnée comme support des fonctions de base. Sur le plan pratique, les RVM sont plus simples à utiliser : leur mise en oeuvre ne nécessite pas de régler des hyperparamètres (comme les paramètres C et ϵ dans les SVM) par validation croisée (cette procédure peut se révéler très couteuse en temps de calcul, surtout si de nombreux tests doivent être réalisés). En revanche, le processus d'optimisation mis en jeu dans la phase d'apprentissage est plus lourd dans le cas des RVM : il requiert en effet à chaque itération d'inverser des matrices de taille $n \times n$, où n est le nombre de vecteurs support sélectionnés à l'itération courante. Des exemples sont progressivement éliminés du modèle, mais la première itération implique obligatoirement d'inverser une matrice de taille $N \times N$, où N est le nombre d'exemples d'entrainement. Une approche incrémentale permettant de gérer le problème a été proposée dans [116] : le modèle est supposé contenir initialement une seule fonction de base, est d'autres fonctions de base sont ajoutées ou retirées du modèle à chaque itération. L'apprentissage des RVM reste néanmoins très coûteux en temps de calcul sur des bases d'entrainement importantes. A l'inverse, le critère optimisé dans l'apprentissage des SVM est un critère quadratique dont on peut facilement trouver la solution unique.

Une autre différence importante se situe dans le nombre et la répartition des vecteurs support parmi les données d'apprentissage. Des exemples de régression et de classification avec les deux machines sont montrés sur la figure 4.6 et 4.7 respectivement. Avec les SVM, les vecteurs supports sont les exemples "limites", c'est-à-dire les vecteurs mal classés ou situés à la frontière entre deux classes, ou, en régression, les exemples éloignés de la fonction interpolée. A l'inverse, les vecteurs sélectionnés dans les RVM ont tendance à être plus répartis dans les données d'apprentissage : les éléments sélectionnés sont des vecteurs représentatifs, des échantillons modèles de la fonction à apprendre (en régression) ou des classes à séparer (en classification). Le

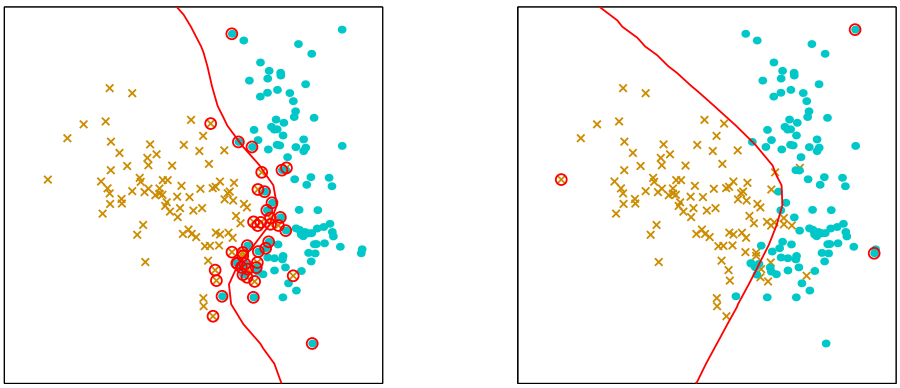
nombre de vecteur support est généralement beaucoup plus faible dans les RVM.

Enfin, contrairement aux SVM, les RVM sont formulés dans un cadre bayésien et fournissent en sortie des prédictions sous la forme de distributions de probabilité.



(a) (b)

FIG. 4.6 – Comparaison des performances des SVM et RVM sur un problème de régression (figure extraite de [114]).
a : régression SVM. **b :** régression RVM.



(a) (b)

FIG. 4.7 – Comparaison des performances des SVM et RVM sur un problème de classification (figure extraite de [114]).
a : classification SVM. **b :** classification RVM.

MVRVM

Les RVM ont été originellement développés pour traiter les problèmes de régression avec des données de sortie unidimensionnelles. Dans le cas où les sorties ont plusieurs dimensions, une machine doit être entrainée pour chaque composante du vecteur de sortie. Les machines sélectionnent des ensembles séparés de vecteurs support, ce qui réduit le caractère "épars" de la régression, car il faudra au final autant de fonctions noyaux qu'il y a au total de vecteurs supports sélectionnés. Une extension des RVM a été développée par les auteurs de [112] pour gérer le cas de sorties à plusieurs dimensions : les *Multivariate Relevance Vector Machines* (MVRVM). Des détails sur l'algorithme peuvent être trouvés dans [111]. La méthode repose sur une adaptation de l'algorithme incrémental proposé dans [116] au cas des sorties multi-dimensionnelles. Cette régression est utilisée dans [106] pour estimer la pose à partir d'un descripteur basé sur le Shape Context 3D.

Approximation MAP

Dans [7], Agarwal et Triggs proposent une variante de l'algorithme RVM initialement introduit par Tipping [113]. Cette formulation peut être vue comme une approximation MAP (maximum a posteriori) de l'apprentissage des RVM. L'algorithme RVM dans sa forme originale introduit une densité a priori sur chaque paramètre du vecteur des poids et utilise une approximation *maximum de vraisemblance de type II*, en optimisant la fonction de vraisemblance par rapport aux hyperparamètres α_i, tandis que les paramètres w_i sont marginalisés. Dans l'approche proposée par Agarwal et Triggs, les hyperparamètres α_i sont d'abord intégrés individuellement pour chaque composante w_i du vecteur des poids, ce qui donne une densité a priori sur les poids de la forme $p(w) \propto \|w\|^{-\nu}$. La valeur optimale de chaque composante du vecteur des poids est ensuite estimée en maximisant la distribution a posteriori des données d'apprentissage.

Selon les auteurs, cette approche conserve le caractère épars du modèle RVM original, car seule une petite partie des poids estimés est non nulle. Un autre avantage de cette méthode est qu'elle est directement applicable au cas de sorties à plusieurs dimensions, elle permet d'estimer une matrice des poids W selon toutes les dimensions dans un processus d'apprentissage unique. En revanche, contrairement aux modèles du paragraphe précédent, les prédictions qu'elle fournit ne sont pas probabilistes : elle donne une unique valeur de la prédiction sans mesure de l'incertitude.

Si on considère le logarithme de la fonction de vraisemblance, utiliser une

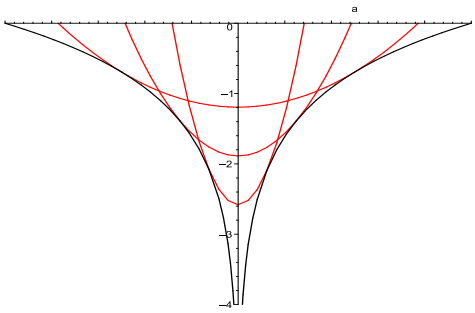


FIG. 4.8 – Approximation par des "ponts quadratiques" (en rouge) du terme de régularisation $\nu log\|w\|$(courbe noire). Figure extraite de [7].

probabilité a priori de ce type revient à introduire dans le modèle un terme de pénalisation de la forme :

$$R(w) = \nu log\|w\| \tag{4.40}$$

Pour l'apprentissage, Agarwal et Triggs choisissent d'approximer ce terme de régularisation par des "ponts quadratiques" : le terme $R(w) = \nu log\|w\|$ est approximé par $\nu(\|w\|/w_{scale})^2 + const$. L'approximation est choisie de manière que les gradients de ces deux fonctions coïncident en w_{scale}. Cette approximation a pour but de permettre aux paramètres de passer par zéro sans être prématurément éliminés (si le terme de pénalisation correspondant devient infini, le paramètre est forcé à 0).

L'algorithme d'apprentissage est le suivant :

1. Initialiser la matrice des poids W par une *ridge regression*. Initialiser le facteur d'échelle $w_{scale} = \|w\|$ pour chaque composante de la matrice W.
2. Approximer le terme de pénalisation $R(w) = \nu log\|w\|$ par des "ponts quadratiques" : $R(w) \approx \nu(\|w\|/w_{scale})^2 + const$
3. Résoudre du système linéaire pénalisé aux moindres carrés en W.
4. Eliminer toutes les composantes w de W devenues trop faibles : les colonnes de W dont la norme est inférieure à un certain seuil T_a sont éliminées.
5. Mettre à jour du facteur d'échelle $w_{scale} = \|w\|$

et retourner en 2.

Le processus est répété jusqu'à la convergence de la matrice des poids W. Dans cet algorithme, deux paramètres sont à fixer : la coefficient ν dans le terme de régularisation, et le seuil T_a, qui détermine à partir de quelle valeur on considère qu'une colonne de W est à éliminer du modèle. Ce dernier paramètre joue sur le nombre de vecteurs support et la précision du modèle final.

L'algorithme d'apprentissage est assez lourd puisqu'il nécessite d'inverser lors des premières itérations des matrices de taille $N \times N$, où N est le nombre d'exemples dans la base, ce qui pourrait représenter un réel handicap sur base d'apprentissage plus importante. Dans [117], une alternative simple basée sur une stratégie *one in, one out* est proposée pour gérer ce problème. L'apprentissage est initialisé avec p exemples de la base choisis au hasard ($p < N$). Lorsqu'un exemple est éliminé du modèle, il est aussitôt remplacé par l'un des exemples restants, et l'algorithme se poursuit jusqu'à ce que chacun des exemples ait été présenté au modèle.

Selon les fonctions de base choisies, l'apprentissage sélectionne automatiquement les caractéristiques des données d'entrée les plus pertinentes pour évaluer les données de sorties. Si la fonction de base choisie est linéaire, c'est-à-dire $\phi(x) = x$, la régression RVM sélectionne les composantes intéressantes (les dimensions) du vecteur d'entrée. Dans le problème d'estimation de la pose à partir d'un descripteur, cela revient à sélectionner les composantes du descripteur intéressantes pour estimer la pose. Si des fonctions noyaux sont utilisées comme fonctions de base, c'est-à-dire $\phi(x) = [K(x, x_1), K(x, x_2), ..., K(x, x_n)]^T$, où $K(x, x_i)$ est une fonction noyau reliant l'exemple x zu vecteur de base x_i, l'algorithme sélectionne les exemples pertinents sur lesquels s'appuie l'estimation. Dans [29], les auteurs proposent de combiner ces deux processus de sélection, en effectuant une première régression avec un noyau linéaire, pour réduire la taille du vecteur d'entrée, puis une seconde régression avec des noyaux gaussiens pour sélectionner les exemples pertinents de la base d'apprentissage.

4.2.4 Réseaux de neurones

Dans ces modèles, la relation non-linéaire entre le vecteur d'entrée et la sortie est définie d'une part par la structure du réseau (nombre de couches, nombre de neurones par couches, fonctions d'activation...) et d'autre part par les valeurs de ses paramètres (poids, biais), ajustées lors de l'apprentissage. La structure la plus simple de réseau neuronal est le *perceptron*. Le perceptron réalise une somme pondérée par les poids synaptiques w_i des données d'entrée

x_i, ajoute un biais b, et utilise une fonction d'activation f pour déterminer la sortie. Le schéma d'un perceptron simple est donné sur la figure 4.9(a). La sortie du perceptron est donc donnée par :

$$y = f(\sum_{i=1}^{n} w_i x_i + b) \tag{4.41}$$

La classe des fonctions qu'il est possible de modéliser avec le perceptron est limitée. Pour modéliser des applications plus générales, on considère généralement des réseaux composés de plusieurs couches de perceptrons avec des poids synaptiques adaptatifs. La première couche est reliée aux entrées, et chaque couche est reliée à la couche précédente. Les sorties des couches autres que la dernière couche ne sont pas visibles à l'extérieur du réseau, on les appelle donc les *couches cachées*. On peut montrer qu'il possible d'approximer n'importe quelle application continue par un réseau de neurones à deux couches ([12]). Un exemple de perceptron multi-couches est donné sur la figure 4.9.

La phase d'apprentissage détermine les valeurs optimales des poids et des biais en fonction des données d'entrainement par des techniques de minimisation non-linéaire (descente de gradient). Des techniques de rétropropagation de l'erreur sont employées pour calculer de manière efficace les dérivées de la fonction d'erreur en fonction des poids et des biais.

La structure la plus communément utilisée est un perceptron à deux couches (une couche de sortie et une couche cachée), avec une fonction de transfert linéaire pour la couche de sortie et une sigmoïde pour la couche cachée. Une structure de ce type est utilisée dans [117]. C'est également la structure retenue par les auteurs de [93, 94] pour construire une architecture d'applications spécialisées (*Specialized Mappings Architecture*) : dans cette approche, l'espace des données d'entrée est partitionné en plusieurs groupes, et pour chaque secteur, un perceptron est appris pour relier les entrées aux sorties. Lorsqu'une nouvelle observation est présentée en entrée du système, des hypothèses de sorties sont générées pour chaque application. Chaque hypothèse sur la pose génère à son tour par projection une observation dans l'espace image, et la pose finalement retenue est celle qui a généré l'observation la plus proche de la nouvelle donnée. Dans [93], les auteurs proposent également de partitionner l'espace des poses avant d'apprendre ces applications spécialisées, afin de mieux gérer les ambiguïtés liées à l'estimation monoculaire. L'idée de ces méthodes, qui consiste à découper l'espace d'apprentissage en plusieurs secteurs et à formuler plusieurs hypothèses pour une

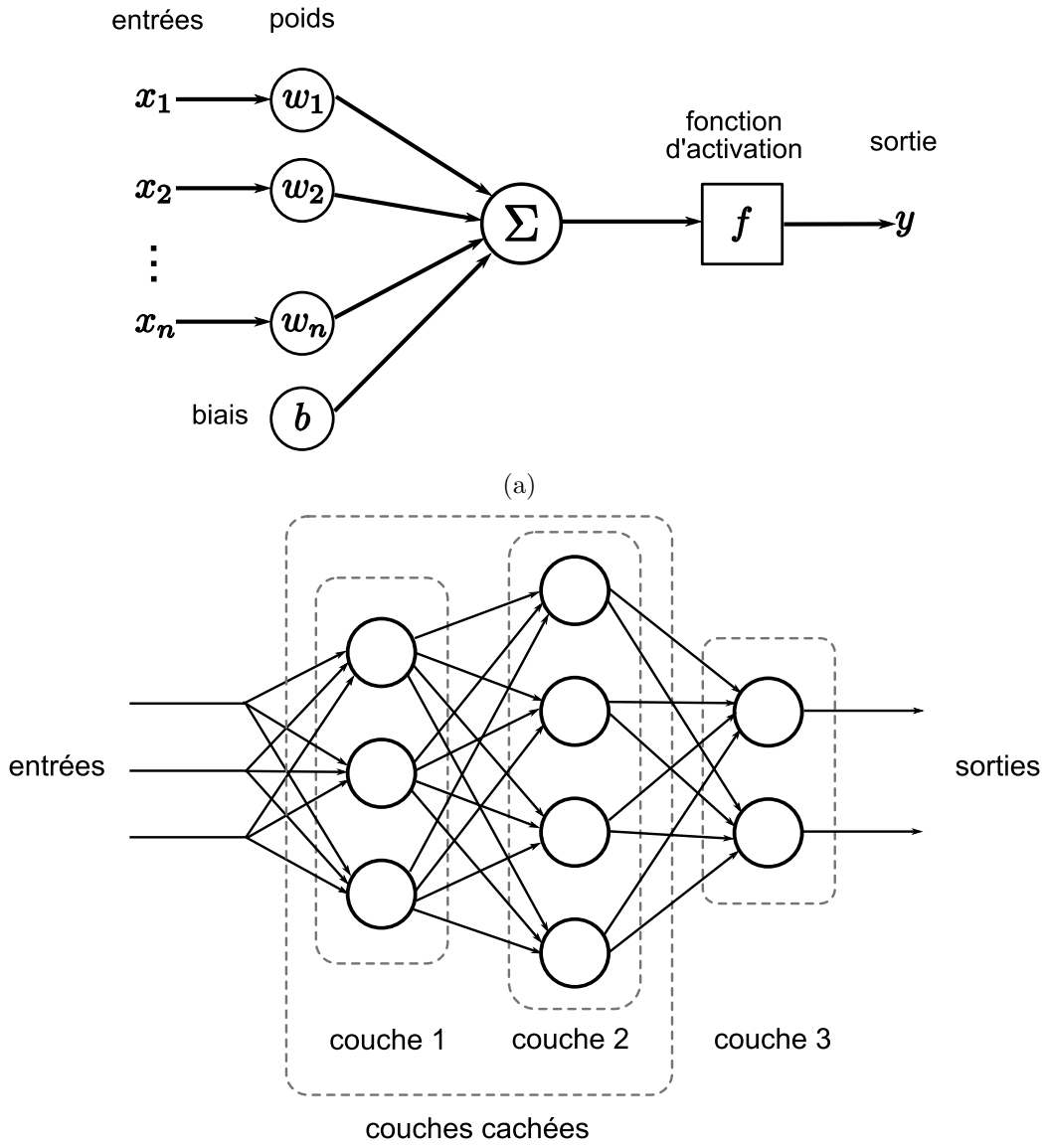


FIG. 4.9 – Structure des réseaux neuronaux.
a : structure du perceptron simple : la sortie du neurone est donnée par : $y = f(\sum_{i=1}^{n} w_i x_i + b)$.
b : exemple de perceptron multi-couches avec deux couches cachées.

nouvelle entrée, rejoint celle des techniques présentées dans le prochain paragraphe.

4.2.5 Modèles à mixtures

Les méthodes de régression décrites dans les paragraphes précédents s'intéressent au cas où l'application que l'on modélise par apprentissage est mono-valuée, c'est-à-dire qu'une seule sortie peut être associée à une entrée donnée. Or, dans notre problème, les ambiguïtés visuelles font que plusieurs hypothèses de sorties doivent parfois être associées à une entrée, en particulier dans le cas où l'estimation est faite à partir d'une seule caméra. Ces ambiguïtés peuvent donner lieu à un phénomène de "moyennage", qui se produit parfois avec les régressions à noyaux : lorsqu'une observation a deux éléments proches l'un de l'autre dans l'espace des descripteurs, mais qui correspondent à des solutions éloignées dans l'espace des poses, cette observation est parfois associée à une sortie moyenne, qui réalise une sorte de compromis entre les différentes solutions possibles. Le résultat peut alors sembler assez paradoxal, car l'observation générée par cette sortie ne correspond généralement pas à la donnée d'entrée ni à aucun des exemples qui lui sont proches.

Pour gérer ce problème, des modèles à mixtures peuvent être employés. Plusieurs régresseurs sont entraînés sur des régions particulières de l'espace des sorties, de sorte que plusieurs estimations peuvent être fournies pour une entrée donnée. On peut trouver des exemples de ce type de modèles dans [5] (mixtures de régresseurs linéaires) et [103] (*Bayesian Mixture of Experts*). Ces méthodes requièrent l'utilisation d'algorithmes de partitionnement dans l'espace de poses, qui est un problème complexe car il faut gérer beaucoup d'exemples dans un espace à grande dimension.

Une question intéressante, soulevée notamment par les auteurs de [29], est de savoir si l'utilisation d'une méthode d'estimation à hypothèses multiples est justifiée dans le cas où plusieurs caméras sont employées. On peut en effet se demander à partir de combien de caméras les ambiguïtés du système deviennent suffisamment faibles pour que la relation image-pose puisse être considérée comme mono-valuée. Dans notre cas, plusieurs caméras sont utilisées pour l'estimation, et on peut tout de même constater en observant la forme de l'enveloppe visuelle (voir paragraphe 2.3.4) que des ambiguïtés subsistent. L'utilisation de modèles à hypothèses multiples pourrait donc être une perspective intéressante à notre problème.

4.3 Estimation de la pose par régression

4.3.1 Régression en deux temps

Le principal problème des méthodes d'estimation basées sur un apprentissage est la complexité de l'espace d'apprentissage et la quantité d'exemples nécessaires dans la base pour bien représenter sa structure. Même en utilisant une machine de régression possédant de bonnes capacités de généralisation, il est nécessaire de bien couvrir dans la base d'apprentissage toutes les possibilités de poses que l'on souhaite reconnaître. Dans le cas d'une personne qui marche, l'estimation de la configuration des membres du corps pourrait être facilitée si l'orientation globale du corps dans l'espace (l'azimtuh) était connue : il serait alors possible de recaler le descripteur sur cette orientation et d'estimer les autres angles avec un descripteur aligné avec l'orientation du corps. L'orientation du descripteur permettrait de réduire la complexité de l'espace des poses. En effet, si le descripteur n'est pas orienté, pour estimer correctement la pose quelque soit l'orientation, il faudrait en théorie que la base d'apprentissage contienne des exemples de chaque pose avec toutes les orientations possibles.

Nous avons donc choisi de décomposer l'estimation de la pose en deux étapes. Dans un premier temps, seule l'orientation du corps est estimée, puis les autres DDL du corps sont évalués grâce à un second descripteur, recalé sur l'orientation globale du corps. Dans nos travaux, l'orientation du torse sert de référence pour l'orientation globale du corps (elle est estimée à partir de l'orientation par rapport à la verticale du segment reliant les deux clavicules - *left collar* et *right collar* sur la figure 2.1). Deux descripteurs sont donc calculés :

- pour le premier descripteur (figure 4.10(a)), la ligne de référence à 0° des histogrammes 2D est aligné avec l'axe x du repère du monde. Ce descripteur est utilisé pour estimer l'angle de rotation α du torse par rapport à l'axe vertical z du repère du monde.
- la ligne de référence pour le second descripteur (figure 4.10(b)) forme un angle α avec l'axe x du repère du monde. Ce descripteur est utilisé pour estimer tous les autres angles du corps (bras, jambes...).

Deux machines de régression sont apprises. La première régression est effectuée avec une base d'apprentissage composée d'exemples de différentes postures ayant des orientations variées, et pour lesquels les axes du descripteur sont alignés avec les axes du repère du monde. Pour la deuxième régression, les descripteurs des éléments de la base sont recalés sur la vérité terrain de

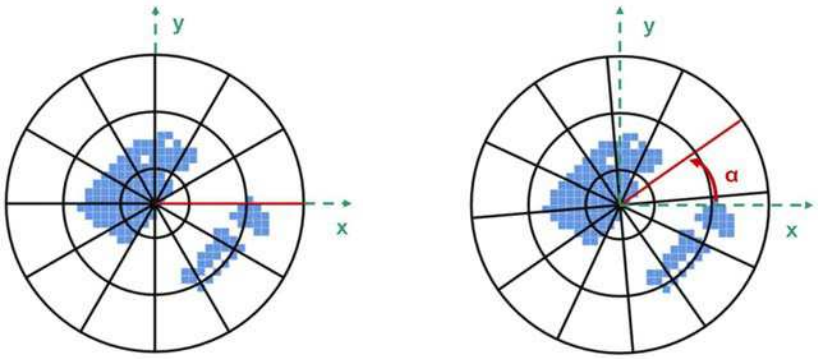
l'orientation du torse (ce qui revient à faire un apprentissage sur des poses dont les orientations sont fixes). L'orientation du descripteur pour la seconde étape permet à la régression de se concentrer sur l'estimation des angles internes et d'atteindre une plus grande précision. On peut de plus penser que l'apprentissage dans la deuxième étape est facilité par le fait que les différentes parties du corps sont toujours situées dans les mêmes secteurs du descripteur (la jambe gauche est dans la partie inférieure gauche du cylindre etc.). Nous avons constaté une réelle amélioration de la précision avec cette régression, particulièrement dans le cas de données réelles assez bruitées. Sur des silhouettes 3D très bruitées, l'orientation du descripteur permet d'éviter l'ambiguïté sur le sens de l'orientation du corps (de face ou de dos).
Comme l'angle d'orientation α peut prendre des valeurs allant 0° à 360°, la régression est effectuée sur le vecteur $(cos\alpha, sin\alpha)$, ce qui permet de maintenir la continuité dans la représentation de la pose. Deux paramètres doivent donc être estimés pour calculer l'orientation.

L'inconvénient principal de cette méthode est que si l'orientation est mal estimée, la régression a peu de chance de retomber sur une pose correcte pour les autres angles, car le descripteur recalé sera mal positionné. D'un autre côté, si l'orientation estimée semble trop imprécise, il est parfois possible de l'évaluer autrement que par régression. Dans le cas d'une séquence de marche, l'orientation peut par exemple être évaluée en se basant sur le vecteur de déplacement entre deux images. Il faut aussi remarquer que dans une séquence vidéo, des orientations aberrantes peuvent facilement être éliminées par un simple filtrage analysant la cohérence temporelle des estimations sur la séquence. Enfin, pour gérer les éventuelles imprécisions dans l'estimation de l'orientation, pour la seconde régression, les orientations des descripteurs des exemples de la base sont légèrement bruitées (voir les expériences du prochain paragraphe).

4.3.2 Evaluation de différentes méthodes de régression

Présentation des expérimentations

Les tests qui suivent sont basés sur les données de Capture de Mouvement provenant de [4] (format BVH). Des tests sur ces mêmes données sont présentés dans [7] et [106]. Ces données sont constituées des différentes séquences de personnes marchant en spirale. Nous avons isolé l'une des séquences pour les tests, et le reste est utilisé pour l'apprentissage. Notre base contient 2534 exemples d'apprentissage et 418 exemples tests. Les mouvements de la base



(a) (b)

FIG. 4.10 – Alignement de la ligne de référence du descripteur avec l'orientation estimée α du torse.
a : Histogramme aligné avec l'axe x du repère du monde. **b :** Histogramme aligné avec l'orientation estimée α du torse.

d'apprentissage sont utilisés pour animer 7 de nos 8 avatars, et les exemples tests sont employés pour animer un 8^e avatar qui ne figure pas dans la base d'apprentissage. Ceci nous permet d'éprouver les performances de la méthode vis-à-vis des changements de morphologie et d'habillement. Les rendus des silhouettes sont effectués dans 3DSMax avec le système de 5 caméras présenté au paragraphe 3.4.1. Quelques exemples de poses générées sont donnés sur la figure 4.11.

FIG. 4.11 – Exemples de postures de marche utilisées dans les tests.

L'objectif de cette partie est de comparer les performances de différentes méthodes de régression dans le cas de l'estimation de la pose. Dans les tests, l'estimation est effectuée en deux temps, comme expliqué dans le paragraphe précédent : l'algorithme estime d'abord l'orientation du corps, puis un second descripteur, recalé sur l'orientation du corps, est calculé pour estimer

méthode	implémentation	sorties multi dim.
modèles linéaires	implémenté suivant 4.2.1	oui
SVM	SVMTorch [27]	non
MVRVM	implémentation Matlab de [2]	oui
approximation MAP	implémenté d'après [7] et [29]	oui

TAB. 4.1 – Algorithmes d'apprentissage utilisés dans les tests.

les angles internes. Cependant, pour avoir une comparaison objective des performances des différentes méthodes, pour l'estimation des angles internes, le descripteur a été recalé sur la vérité terrain de l'orientation. En effet, si une méthode a été mauvaise pour estimer l'orientation, elle sera d'emblée pénalisée pour estimer les autres angles, et la comparaison pourrait être biaisée.

Les méthodes de régression testées, ainsi que l'implémentation utilisée pour chaque méthode sont résumées dans la tableau 4.1. Pour chaque méthode, les erreurs moyennes sur les 418 exemples de la séquence de test ont été calculées pour tous les angles présentés dans le tableau 2.1 du chapitre 2. Comme il a été dit dans ce chapitre, l'erreur moyenne sur tous les angles du corps peut être sujette à certaines ambiguïtés, et des mouvements semblables peuvent parfois être encodés de différentes manières dans la chaîne cinématique. On observe parfois un décalage entre les zones d'amplitudes de certains angles d'une séquence à l'autre, par exemple certains mouvements du bras peuvent être générés soit par une rotation de l'articulation située au niveau de la clavicule (*collar*), soit par une rotation au niveau de l'épaule. Nous avons donc choisi de reporter, pour chaque méthode, à la fois la moyenne des erreurs sur les 22 angles internes, et les erreurs moyennes pour deux angles fortement impliqués dans la marche et qui ne semblent pas présenter ces ambigüités (rotation avant-arrière du bras autour de l'épaule et balancement avant-arrière de la jambe).

Pour chacune des méthodes testées, des noyaux gaussiens ont été utilisés comme fonctions de base. L'écart-type σ des noyaux a été fixé en fonction de l'écart-type des vecteurs d'entrée de la base d'apprentissage, selon la même heuristique que dans [54] (c'est-à-dire en prenant $\sigma = \|\vec{\sigma_{desc}}\|$, où $\vec{\sigma_{desc}}$ est le vecteur réunissant l'ensemble des écarts-type suivant les différentes dimensions des données d'entrée). En toute rigueur, ce paramètre devrait être fixé pour chaque méthode par validation croisée, mais cette procédure est très

coûteuse en temps de calcul, et la valeur du paramètre σ n'est pas capitale pour la précision, du moment qu'elle est choisie dans un ordre de grandeur raisonnable. La plupart des méthodes de régression dépendent d'hyperparamètres qui conditionnent la qualité des résultats. Pour chaque machine, ces hyperparamètres ont été réglés au mieux par des tests préalables, mais il faut garder à l'esprit que la précision et le nombre de vecteurs support sélectionnés peuvent dépendre de ces choix.

Modèles linéaires

Dans une première expérience, nous avons utilisé pour la régression un simple modèle linéaire, constitué d'une combinaison linéaire de noyaux gaussiens. Le calcul des poids optimaux de la combinaison se fait suivant la méthode présentée en 4.2.1, c'est-à-dire en calculant la pseudo-inverse de la matrice des noyaux. Nous n'avons pas introduit de terme de régularisation dans la fonction de coût à minimiser. Cependant, seule une partie des exemples d'apprentissage est sélectionnée pour servir de support aux fonctions de base, ce qui permet tout de même d'obtenir un modèle assez épars et de se prémunir du surapprentissage. Ces vecteurs support sont sélectionnés aléatoirement parmi les exemples. Les tests présentent les résultats obtenus en fonction du nombre d'éléments sélectionnés dans la base. Comme le résultat peut dépendre des exemples sélectionnés, plusieurs tirages ont été réalisés pour chaque valeur N_f du nombre de fonctions de base. Les résultats sont reportés dans le tableau 4.2. Les valeurs inscrites dans le tableau sont les erreurs moyennes sur les 418 exemples de la séquence test, et l'écart-type de l'erreur sur les 10 tirages est donné entre parenthèses. La colonne "corps entier" donne les erreurs moyennes sur l'ensemble des 22 angles du tableau 2.1. Le tableau 4.2 donne quelques unes des valeurs de l'erreur obtenues pour différentes valeurs de N_f, et la figure 4.12 présente une courbe montrant l'évolution des erreurs moyennes en fonction de N_f.

La première constatation que l'on peut faire est que l'angle d'orientation du corps semble être le plus difficile à estimer, ce qui peut s'expliquer d'une part par le fait que cet angle décrit un intervalle d'amplitude plus élevée que tous les autres angles du corps (de 0° à 360°), et d'autre part par la variété des poses qui peuvent être prises par l'avatar pour une orientation donnée. On observe aussi sur les courbes qu'à partir d'une certaine valeur de N_f (environ 500 ou 600 exemples), il ne sert plus à rien d'ajouter de nouveaux exemples dans le modèle car le gain en précision n'est plus significatif. L'ajout de fonctions de base supplémentaires risque même au contraire de dégrader

nb. fcts base N_f	orientation	corps entier	épaule gauche	jambe droite
20	21.7° (4.6°)	4.6° (0.5°)	4.9° (0.5°)	4.5° (0.6°)
50	11.1° (1.8°)	3.7° (0.2°)	4.2° (0.2°)	3.7° (0.2°)
100	8.3° (0.5°)	3.4° (0.2°)	3.8° (0.1°)	3.4° (0.2°)
200	7.2° (0.6°)	3.1° (0.2°)	3.5° (0.2°)	3.1° (0.1°)
500	5.9° (0.3°)	3.0° (0.1°)	3.1° (0.1°)	3.0° (0.1°)
700	6.0° (0.4°)	2.9° (0.1°)	3.1° (0.1°)	3.0° (0.1°)
1000	5.8° (0.2°)	2.9° (0.1°)	3.0° (0.2°)	3.0° (0.1°)
1500	5.8° (0.3°)	2.9° (0.1°)	3.0° (0.1°)	3.0° (0.1°)
2000	5.8° (0.1°)	3.0° (0.1°)	3.0° (0.1°)	3.0° (0.1°)

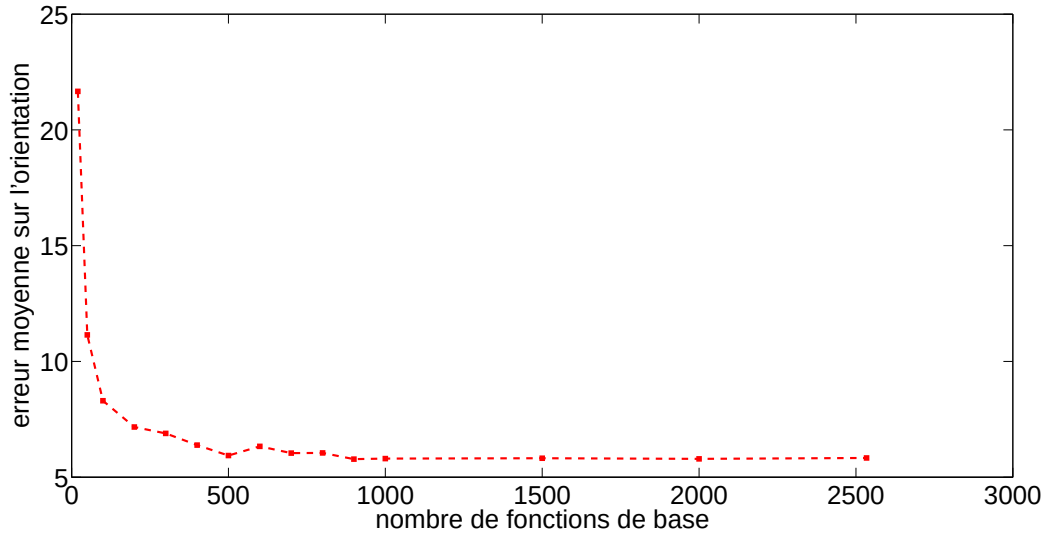
TAB. 4.2 – Précision du modèle linéaire en fonction du nombre d'exemples sélectionnés (parmi les 2534 exemples d'apprentissage) pour les fonctions de base.

les performances du modèle en causant un surapprentissage (comme peut le laisser penser la courbe bleue sur la figure 4.12(b), pour laquelle l'erreur a tendance à augmenter lorsque tous les exemples sont sélectionnés).

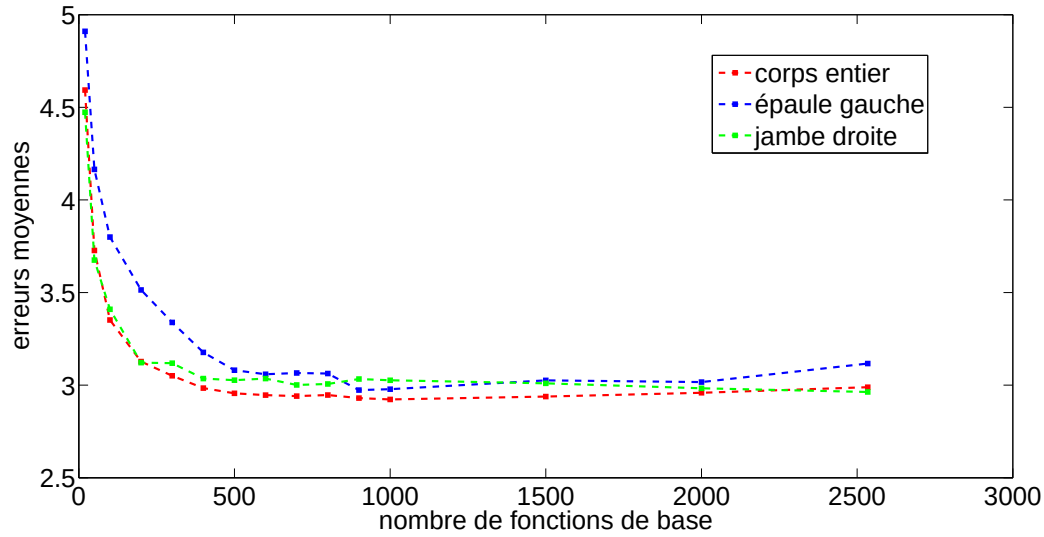
MVRVM

Nous avons effectué des tests avec une régression MVRVM. Une version de cette méthode est disponible en ligne [2] (implémentation fournie par les auteurs de [112]). Comme il a déjà été dit, les RVM sont assez commodes à mettre en oeuvre puisqu'ils ne nécessitent le réglage d'aucun hyperparamètre (hormis l'écart-type des noyaux dans le cas de noyaux gaussiens). Une machine est utilisée pour apprendre les deux paramètres de l'orientation ($cos\alpha$ et $sin\alpha$), et une autre pour tout le reste du corps. Il est possible d'utiliser une machine RVM "classique" pour chaque dimension, mais cette alternative n'a pas été testée ici. Nous avons fixé à 200 le nombre d'itérations maximal dans le processus d'apprentissage. Les résultats sont présentés dans le tableau 4.3. Les vecteurs supports sélectionnés pour les angles internes sont les mêmes pour tous les angles du corps.

On peut constater que la méthode a été très efficace pour estimer l'orientation, car avec seulement 200 vecteurs support, le modèle atteint une précision aussi bonne qu'avec une modèle linéaire avec 1000 vecteurs support tirés aléatoirement. En revanche, les bénéfices de la méthode sont moins évidents pour les autres degrés de liberté. Dans le cas de l'orientation, la régression s'effectue avec des vecteurs de sortie à deux dimensions, tandis que pour les



(a)



(b)

FIG. 4.12 – Evolution de l'erreur moyenne (en degrés) du modèle linéaire en fonction du nombre d'exemples sélectionnés pour les fonctions de base.
a : erreur moyenne sur l'orientation. **b :** erreur moyenne sur les autres angles du corps.

angles internes, les sorties sont de dimension 22. Il semble donc que lorsque la régression s'effectue sur de nombreux paramètres de sortie en même temps, les avantages de cette méthode pour construire un modèle épars sont plus dis-

	orientation	corps entier	épaule gauche	jambe droite
nb. RV	200	199	-	-
erreur	5.8°	3.0°	3.4°	3.1°

TAB. 4.3 – Erreurs sur les angles estimés avec une régression MVRVM.

	orientation	corps entier	épaule gauche	jambe droite
nb. SV	1090	-	1852	1631
erreur	5.5°	2.8°	3.1°	2.9°

TAB. 4.4 – Erreurs sur les angles estimés avec une régression SVM.

cutables. On peut penser que la machine RVM doit faire des compromis pour sélectionner les mêmes vecteurs supports pour toutes les dimensions. Lorsque les vecteurs atteignent une certaine dimension, il devient moins évident de sélectionner un petit nombre de vecteurs pertinents pour toutes les dimensions à la fois.

SVM

L'implémentation utilisée pour les SVM est celle de la librairie Torch [27]. Une machine est apprise pour chaque degré de liberté. A notre connaissance, il n'existe pas de formulation des SVM pour traiter des sorties à plusieurs dimensions. Le paramètre de régularisation C a été fixé par des tests préalables, et, à l'exemple de [7], la largeur ϵ du tube dans la fonction de coût est réglée pour correspondre à une erreur de 1° pour chaque angle (pour l'orientation α, ϵ a été réglé pour correspondre à une erreur de 0.1 dans les régressions sur $cos\alpha$ et $sin\alpha$). Les résultats sont donnés dans le tableau 4.4. Ici, des ensembles de vecteurs support différents sont sélectionnés pour chacune des dimensions.

On peut observer que la précision atteinte est globalement meilleure qu'avec les précédentes méthodes. Les auteurs de [7] attribuent les meilleures performances des SVM à la forme différente de leur fonction de coût. En revanche, le nombre de vecteurs support sélectionnés pour chaque degré de liberté est très important (plus de 50% des exemples d'apprentissage), et ces vecteurs support n'étant pas les mêmes pour chaque dimension, l'estimation doit au final estimer les valeurs des fonctions noyaux pour presque tous les exemples de la base.

seuil T_a	nb. VS	erreur (en degrés)
0.5	1712	5.9
1	551	6.4
2	37	7.2
3	19	11.8
4	3	53.6

TAB. 4.5 – Nombre de vecteurs support sélectionnés et précision du modèle en fonction du seuil T_a pour l'estimation de l'orientation du corps.

seuil T_a	nb. VS	corps entier	épaule gauche	jambe droite
1	2534	2.9	3.2	2.9
5	2425	2.9	3.2	2.9
10	1404	2.9	3.2	3.1
15	219	3.1	3.8	3.8
20	54	3.6	5.4	5.3

TAB. 4.6 – Nombre de vecteurs support sélectionnés et précision du modèle (erreurs en degrés) en fonction du seuil T_a pour l'estimation des angles internes.

Approximation MAP des RVM

L'approximation MAP des RVM introduite par Agarwal et Triggs a été implémentée suivant la méthode présentée dans [7] et [29]. Sa mise en oeuvre nécessite de régler d'une part le coefficient ν du terme de pénalisation, et d'autre part le seuil T_a en dessous duquel une colonne de la matrice des poids est éliminée du modèle. Dans nos tests, le premier paramètre a été réglé par des tests préalables, et, à l'exemple de [29], nous présentons les résultats avec différentes valeurs du seuil T_a. Ce paramètre joue un rôle important car il établit un compromis entre la complexité du modèle (le nombre de vecteurs support) et la précision atteinte. Les résultats sont présentés dans les tableaux 4.5 et 4.6, respectivement pour l'estimation de l'orientation et des angles internes.

Comme pour les MVRVM, il semble que l'algorithme ait plus de difficultés à sélectionner un nombre réduit de vecteurs support lorsque l'estimation porte sur des sorties de plus grande dimension.

Ces expériences ne permettent pas de montrer un avantage significatif de cet algorithme par rapport à un choix aléatoire des vecteurs support d'un modèle linéaire (tableau 4.2) : à nombre de vecteurs support égal, l'estimation

RVM n'a pas été plus performante. Il faut toutefois considérer ces résultats avec prudence car certains détails ou paramètres de l'algorithme n'ont peut être pas été réglés de manière optimale.

Conclusion des expériences

En conclusion de ces expériences, on peut penser qu'aucune de ces méthodes de régression ne se dégage réellement. Tous les algorithmes d'apprentissage ont leurs avantages et leurs inconvénients. Les modèles linéaires simples manquent de contrôle sur le mode de sélection des fonctions de base, à la fois pour ce qui est du nombre d'exemples à sélectionner et de leur répartition dans la base. Dans nos expériences, des tests préliminaires ont été effectués pour fixer ces paramètres. D'un autre côté, l'apprentissage de ces modèles est particulièrement commode puisqu'il ne requiert l'inversion que d'une seule matrice, et la précision atteinte est tout de même satisfaisante en comparaison des autres méthodes. Les SVM ont permis d'obtenir la meilleure précision, mais leur utilisation implique d'apprendre une machine par degré de liberté, et le caractère épars de la régression s'en trouve fortement réduit. Les RVM réalisent un bon compromis, mais là encore, l'efficacité de la méthode semble diminuer lorsque la dimension des sorties augmente.

Dans les tests qui suivent, nous précisons à chaque fois le type de méthode utilisée pour la régression. La plupart des tests ont été réalisés avec de simples modèles linéaires, en particulier lorsque l'évaluation visée ne concerne pas réellement la précision de la régression, mais plutôt l'influence d'autres facteurs sur l'estimation, comme par exemple le choix de la paramétrisation des mouvements (voir paragraphe 4.3.3). Pour les tests nécessitant une mesure exacte de la précision (comme en 4.3.2), notre choix s'est porté sur les SVM.

Evaluation des gains en précision avec une régression en deux temps

Nous reprenons ici les données d'apprentissage et de tests utilisées dans les paragraphes précédents. L'objectif est d'évaluer le gain en précision obtenu sur l'estimation des angles internes en recalant le descripteur sur l'orientation du corps. Pour cela, nous comparons les précisions atteintes avec 3 méthodes :

– une première régression où les angles internes sont estimés directement avec le descripteur aligné sur les axes du repère du monde (le même descripteur que celui qui est utilisé pour estimer l'orientation),

	sans rec.	rec. estim.	rec. estim.+ bruit	recalage VT
corps entier	3.4°	3.1°	3.0°	2.8°
épaule gauche	4.0°	3.9°	3.7°	3.1°
jambe gauche	3.3°	3.0°	3.0°	2.9°

TAB. 4.7 – Comparaison des erreurs sur les angles internes avec et sans recalage du descripteur.

- une régression où les angles internes sont estimés avec un descripteur recalé sur l'orientation estimée,
- la même régression que la précédente, mais pour laquelle les orientations des exemples de la base d'apprentissage ont été légèrement bruitées,
- une régression où les angles internes sont estimés avec un descripteur recalé sur la vérité terrain de l'orientation.

Pour la troisième méthode, les descripteurs des exemples de la base d'apprentissage ont été recalés sur l'orientation de la vérité terrain à laquelle un petit angle a été ajouté, dont la valeur est tirée aléatoirement dans l'intervalle $[-10°, 10°]$. Ceci nous permet de tenir compte des imprécisions dans l'estimation de l'orientation et d'apprendre à la deuxième partie de l'apprentissage à être robuste à un bruit sur le recalage du descripteur. Nous avons utilisé des SVM comme machines de régression. Les résultats sont présentés dans le tableau 4.7.

On peut constater que le recalage du descripteur a bien permis d'améliorer l'estimation des angles internes. Ajouter un bruit sur l'orientation dans la base d'apprentissage semble aussi présenter un intérêt. Si les écarts obtenus peuvent sembler faibles, il faut souligner que les expériences sont effectuées ici sur des données "parfaites"(les silhouettes 2D sont idéales) et donc très favorables. Dans le cas de silhouettes 3D très bruitées, nos expérimentations ont mis plus clairement en évidence les avantages du recalage du descripteur.

4.3.3 Comparaison des deux paramétrisations sur des séquences de gestes

Les données utilisées pour les tests de cette partie sont analogues à celles du paragraphe 3.4.1. Les 8 avatars (figure 2.13) sont animés en tirant aléatoirement les angles des bras (intervalles donnés dans le tableau 3.1). Quelques exemples des postures générées sont donnés sur la figure 4.13.

Les angles étant tirés aléatoirement, les avatars sont animés chacun avec des jeux d'angles différents. L'objectif de cette partie est de comparer sur des

FIG. 4.13 – Exemples de poses générées pour l'apprentissage et les tests sur les gestes.

séquences de gestes les performances obtenues en régression avec les deux paramétrisations : avec les angles et avec les positions 3D des articulations du squelette. Chacun des avatars a été tour à tour retiré de la base d'apprentissage : ses mouvements ont été utilisés comme données tests, et les mouvements des 7 autres avatars constituent la base d'apprentissage. 500 exemples ont été générés pour chaque avatar : pour chaque régression, 3500 exemples sont utilisés pour l'apprentissage, et 500 pour les tests. Le système de caméras utilisé est à nouveau celui de la figure 3.15(a) (5 caméras). La régression est effectuée soit en estimant les 3 angles des deux épaules et l'angle des deux coudes, soit en calculant la position 3D des coudes et des mains. Dans le premier cas, 8 paramètres sont estimés par régression, et dans l'autre cas 12 paramètres (comme il a été dit au chapitre 2.5, la représentation par les positions 3D est surparamétrée). Pour l'estimation des points 3D, le squelette doit être normalisé pour garantir l'invariance aux changements d'échelle, toutes les postures sont donc exprimées en se basant sur le même squelette (celui de l'avatar par défaut de POSER). Tous les tests sont effectués avec le même descripteur et la même machine de régression, c'est-à-dire un modèle linéaire s'appuyant sur des exemples de la base. Comme les estimations portent sur des gestes, seule la partie supérieure du descripteur est prise en compte.

Pour comparer les performances des deux estimations, on se ramène dans les deux cas à une erreur sur la position 3D des points du corps. Pour l'estimation sur les angles, le squelette de l'avatar est animé avec les angles trouvés par régression, et les positions 3D des points du squelette sont calculées. Toutes les erreurs 3D sont exprimées sur ce même squelette (l'avatar mesure environ 2m).

Comme il a déjà été dit au chapitre 2.5, la paramétrisation par les positions des points 3D semble intuitivement plus à l'image des variations 3D de l'enveloppe et donc du descripteur. Avec une paramétrisation par des angles,

beaucoup d'enveloppes très différentes peuvent avoir certains paramètres de sorties en commun. Par exemple, une même valeur de l'angle d'un coude peut correspondre à des poses très différentes en fonction de la configuration de l'épaule. Comme chaque angle est appris indépendamment des autres, cela signifie que deux paramètres de sortie similaires peuvent être associés à des observations très différentes, et cette ambiguïté doit sans doute pénaliser la régression. Les courbes de la figure 4.14 montrent l'évolution des erreurs sur les exemples tests d'un avatar (avatar 1) en fonction du nombre de vecteurs support sélectionnés dans la base d'apprentissage.

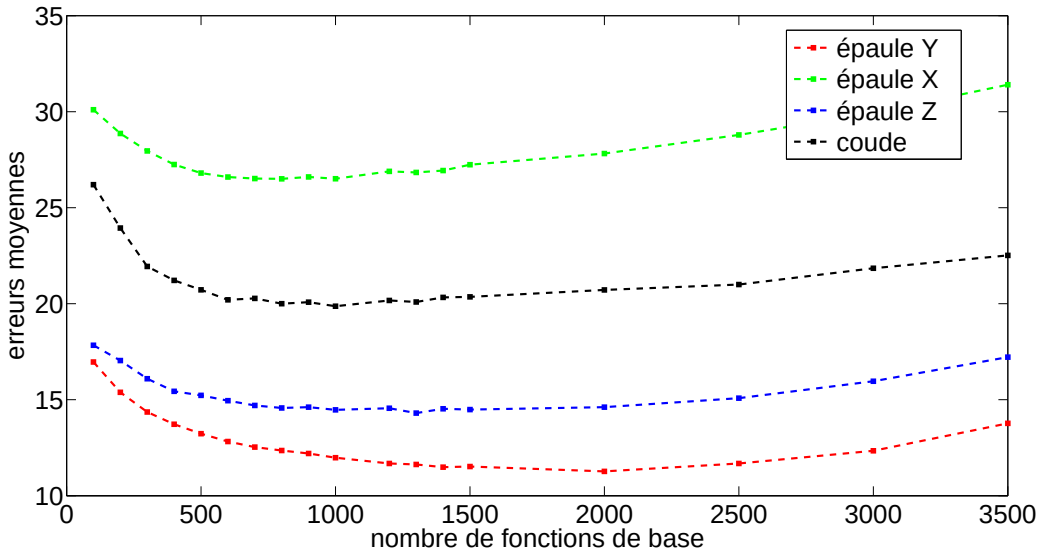


FIG. 4.14 – Erreurs moyennes (en degrés) sur les angles du bras droit en fonction du nombre de fonctions de base sélectionnées.

On constate tout d'abord que les valeurs des erreurs angulaires sont beaucoup plus élevées sur ce type de mouvement que sur les séquences de marche du paragraphe précédent. Ce phénomène semble logique puisque les gestes sont des mouvements beaucoup plus complexes que la marche, et génèrent des enveloppes beaucoup plus variées. Dans la marche, les amplitudes des intervalles angulaires sont d'une part plus restreintes, et certaines couches du descripteur varient quasiment linéairement en fonction des angles (par exemple les couches situées au niveau des jambes). Les positions des différents membres du corps sont de plus très corrélées dans une posture de marche. Pour maintenir l'équilibre du corps, les mouvements des deux jambes doivent être coordonnés : si la jambe gauche est en avant, la jambe droite doit être en

arrière, etc. Les mouvements des bras sont aussi souvent corrélés avec ceux des jambes. Dans les postures de gestes présentées ici, chaque angle a été sélectionné aléatoirement et indépendamment des autres angles : la régression ne peut pas s'appuyer pour prédire un angle sur une partie du descripteur autre que celle correspondant au membre du corps concerné.

On peut aussi remarquer sur ces graphes que les angles les plus difficiles à estimer sont l'angle de rotation de l'épaule autour de l'axe du bras et l'angle du coude. Les ambiguïtés sur l'estimation de l'angle de l'épaule autour de l'axe x ont déjà été abordées dans le chapitre 2.5. L'erreur sur l'angle du coude peut elle aussi s'expliquer par le fait que le coude est plus éloigné que l'épaule dans la chaîne cinématique, et que sa position est dépendante de la configuration de l'épaule. En toute rigueur, il faudrait estimer l'angle du coude une fois connus les angles de l'épaule, plutôt que de considérer ces paramètres indépendamment les uns des autres.
On peut enfin noter que les erreurs moyennes diminuent lorsque le nombre de fonctions de base augmente, mais seulement jusqu'à un certain point : si le nombre de vecteurs support est trop grand, la matrice que l'on inverse est mal conditionnée et cela donne lieu à un surapprentissage.

Pour les tests visant à comparer les deux paramétrisations, nous avons utilisé pour l'apprentissage des modèles linéaires s'appuyant sur 1500 exemples sélectionnés aléatoirement (parmi les 3500 exemples). Pour éviter que l'une des deux méthodes ne soit pénalisée par le tirage des vecteurs support, le même ensemble d'exemples est utilisé dans les deux régressions. Le tableau 4.8 donne les erreurs 3D obtenues successivement sur chaque avatar.

Comme on pouvait s'y attendre, la paramétrisation par les points 3D a donné de meilleurs résultats que la paramétrisation par des angles. On peut observer que pour deux des avatars (avatars 2 et 4), les erreurs 3D sont sensiblement plus grandes que pour les autres, ce qui laisse penser que ces deux avatars doivent avoir une morphologie un peu différente des autres. On constate aussi que la correction appliquée aux positions 3D des points après l'estimation par régression n'a pas perturbé les résultats, mais a au contraire permis d'améliorer la précision sur la position des coudes et des mains (excepté pour les deux avatars "problématiques", pour lesquels la position des mains a été légèrement dégradée).

	angles		pts 3D sans corr.		pts 3D avec corr.	
	coudes	mains	coudes	mains	coudes	mains
avatar 1	8.7	15.2	8.2	12.6	7.7	12.5
avatar 2	13.5	20.8	13.5	16.5	13.4	18.01
avatar 3	7.9	16.0	7.5	13.4	6.9	13.2
avatar 4	11.6	21.9	11.1	16.9	10.3	17.5
avatar 5	7.5	16.5	7.1	13.7	6.5	13.5
avatar 6	10.1	17.4	9.9	14.3	9.5	14.2
avatar 7	8.8	17.2	8.3	15.1	7.7	15.0
avatar 8	10.2	16.3	9.3	13.8	8.7	13.2
moyenne	9.8	17.7	9.4	14.5	8.8	14.6

TAB. 4.8 – Erreurs moyennes (en cm) obtenues sur les positions des coudes et des mains avec les deux paramétrisations.
1ère colonne : erreurs 3D avec la paramétrisation par les angles. **2e colonne :** erreurs 3D avec la paramétrisation sur les positions (x, y, z) des coudes et des mains. **3e colonne :** même paramétrisation que la précédente, mais avec une correction sur la position des points pour ramener les segments à la bonne longueur.

4.4 Raffinement de la pose

4.4.1 Etat de l'art

Nous avons présenté dans le paragraphe 1.3.3 de l'état de l'art des méthodes d'estimation basées sur un modèle du corps humain, dont la configuration spatiale est ajustée pour correspondre au mieux aux données de l'image. Parmi ces méthodes, certaines se basent comme nous sur l'enveloppe 3D reconstruite à partir d'un système multi-caméras. Dans [71], le modèle du corps est constitué de 10 parties : un cylindre pour le torse et des ellipsoïdes pour les autres membres du corps. Dans la première vue de la séquence, le modèle est ajusté séquentiellement sur la reconstruction en voxels : la tête est localisée en premier, puis le torse et enfin les autres membres du corps. Un filtre de Kalman étendu est ensuite employé, à la fois pour corriger l'estimation de la première vue et s'assurer de sa validité par rapport au modèle, et pour reconstruire la pose dans la suite de la séquence. Dans [14], le modèle utilisé est composé d'un squelette comprenant 15 segments (modèle à 32 DDL), recouvert par une surface modélisée par un maillage triangulaire, dont les paramètres sont ajustés pour correspondre aux caractéristiques de

l'acteur dont on reconstruit les mouvements. Dans cette approche, on minimise par une descente de gradient une fonction de coût basée sur la somme des distances entre le centre de chaque voxel et le segment du modèle le plus proche. A chaque vue (excepté pour la vue initiale), l'estimation de l'image précédente sert de point de départ pour l'optimisation. Les auteurs de [72] utilisent comme modèle un simple squelette du corps humain. Ce modèle est ajusté en s'appuyant les points obtenus par squelettisation de l'enveloppe. A chaque vue, la pose optimale est recherchée en partant de la pose précédente, par un processus d'optimisation en deux temps (algorithme EM) : l'étape E estime l'appartenance des points $\{X_0, ..., X_n\}$ de la squelettisation à l'un des segments du modèle, et l'étape M optimise la pose du modèle par un algorithme de Levenberg-Marquardt, en supposant la correspondance entre les points X_i et le modèle établie à l'étape E.

Dans le cas d'une méthode basée sur un modèle, il est intéressant de s'appuyer sur une reconstruction 3D pour optimiser le modèle plutôt que d'ajuster sa projection en 2D dans chacune des images. Cette dernière solution suppose en effet que le modèle doit être projeté plusieurs fois dans les images (pour chaque hypothèse sur la configuration du corps), tandis que la reconstruction 3D est calculée une fois pour toutes. Comme on l'a vu dans l'état de l'art, les méthodes basées sur un modèle sont généralement assez précises, mais elles sont aussi coûteuses en temps de calcul et nécessitent une bonne initialisation pour converger. L'estimation s'effectue le plus souvent dans le cadre d'un suivi, c'est-à-dire que pose courante est recherchée en partant du résultat de l'image précédente. A l'inverse, les méthodes basées sur un apprentissage sont rapides et ne nécessitent pas d'initialisation, mais peuvent manquer de précision. Il peut donc être intéressant de chercher à combiner les avantages des deux types d'estimation, en utilisant le résultat obtenu par la régression pour initialiser un modèle du corps, et affiner l'estimation. Ainsi, au lieu de prendre comme point de départ de l'optimisation l'estimation de la vue précédente, il est envisageable d'utiliser l'estimation fournie par la machine de régression. Il faut souligner que dans notre cas, un avantage de l'apprentissage est qu'il permet de prendre en compte implicitement les artéfacts de la silhouette 3D. En effet, si la configuration du système de caméras est la même à l'apprentissage et à l'estimation, la régression "apprend" ces artéfacts, et peut retourner une pose valide même si l'enveloppe 3D reconstruite est ambigüe, alors qu'un approche basée sur une modèle risquerait au contraire de converger sur les voxels fantômes.

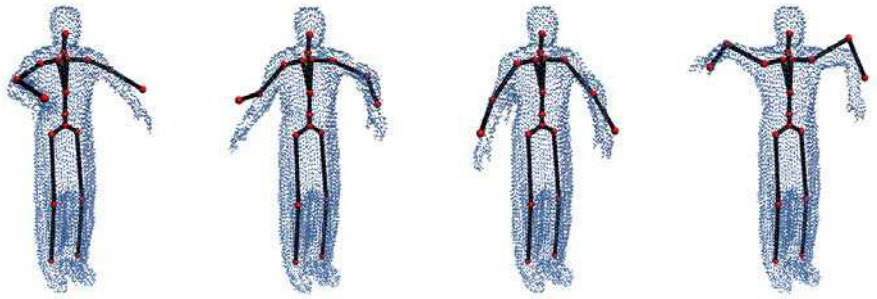


FIG. 4.15 – Exemples d'estimations imprécises qui peuvent être améliorées en ramenant le squelette à l'intérieur des voxels.

4.4.2 Méthode de raffinement

On peut observer que l'estimation de la pose fournie par la régression est souvent correcte mais manque de précision. On reprend ici les données de tests du paragraphe 4.3.3. La figure 4.15 présente des résultats obtenus en animant un squelette avec les positions 3D des mains et des coudes obtenues par régression. Les données utilisées pour ces expérimentations sont les séquences de gestes du paragraphe 4.3.3, et les résultats sont présentés pour les estimations sur l'avatar 1 (l'apprentissage étant réalisé avec les 7 autres avatars). Sur la figure, le squelette estimé est superposé aux voxels de l'enveloppe (pour que la figure soit lisible, seuls des voxels de la surface sont représentés). On peut observer sur ces exemples que le squelette obtenu est souvent légèrement en dehors de l'enveloppe. La précision de l'estimation pourrait facilement être améliorée en affinant la position des segments du squelette et en les ramenant à l'intérieur de l'enveloppe.

Nous avons donc développé une méthode de raffinement se basant sur une comparaison de la position des segments du squelette par rapport aux points les plus proches de l'enveloppe. Notre méthode fait l'hypothèse que la pose a été correctement estimée, c'est-à-dire que l'estimation n'est pas trop éloignée de la pose réelle.

On modélise les membres du corps dont on souhaite affiner la position par des parallélépipèdes (voir figure 4.16(a)) : le segment du squelette est entouré d'un volume approximant la forme d'un membre du corps. Ce parallélépipède est discrétisé, et pour une position donnée du segment, un taux de remplissage est calculé en comptant le nombre de points du parallélépipède situés à l'intérieur de l'enveloppe. Ainsi, si le segment du squelette est to-

talement à l'intérieur de l'enveloppe, son taux de remplissage sera de 100%, tandis qu'il sera plus faible si seulement une partie du segment chevauche l'enveloppe, voire nul s'il est complétement en dehors des voxels. L'objectif est de trouver la position du segment dans le voisinage de la position estimée ayant un taux de remplissage maximal. La position optimale du segment est recherchée dans un cône d'angle θ autour de la position initiale du segment, ayant pour sommet l'articulation parente du segment dans la chaîne cinématique. La méthode de raffinement procède séquentiellement, c'est-à-dire que les segments sont affinés un par un en fonction de leur ordre dans la chaine cinématique. Dans le cas des gestes, la position du coude est d'abord affinée en faisant varier l'angle du bras par rapport à l'épaule, puis le coude est déplacé et la position de la main est recalculée en fonction de celle du coude par la méthode présentée en 2.2.3. La position de la main est ensuite ajustée suivant le même processus, en se basant sur la nouvelle position du coude.

L'angle de recherche θ est estimé en se basant sur le taux de remplissage initial p du segment. En effet, si la position estimée par régression est déjà suffisamment bonne et que le taux de remplissage est satisfaisant, il n'est pas nécessaire de rechercher la position optimale dans un large voisinage car cela risquerait de détériorer l'estimation. Par exemple si le bras est collé le long du corps et que sa position a été correctement estimée, le segment risque d'être attiré par des voxels du torse. En revanche, si le taux de remplissage initial est mauvais, on a tout intérêt à rechercher la position optimale du modèle assez loin de la pose estimée. La figure 4.16(c) présente l'approximation qui a été utilisée pour estimer l'angle θ : le taux de remplissage initial est supposé être proportionnel au rapport l/L, et on peut alors obtenir une approximation de $sin(\theta)$ comme montré sur la figure. L'angle θ choisi vaut $arcsin(\frac{c}{2pL})$, où L est la longueur du segment et c le côté de la section du parallélépipède. Bien entendu, cette valeur représente une approximation assez grossière de l'angle du cône dans lequel se situe la pose optimale.

Une fois l'angle θ estimé, le cône d'angle θ autour du segment initial est discrétisé, et de nouvelles positions du segment, régulièrement réparties dans le cône, sont générées. Pour chaque échantillon, le taux de remplissage du segment est calculé. La position choisie est celle qui a obtenu le score maximal. Enfin, si le score final est encore inférieur à un certain seuil, le processus est relancé une seconde fois.

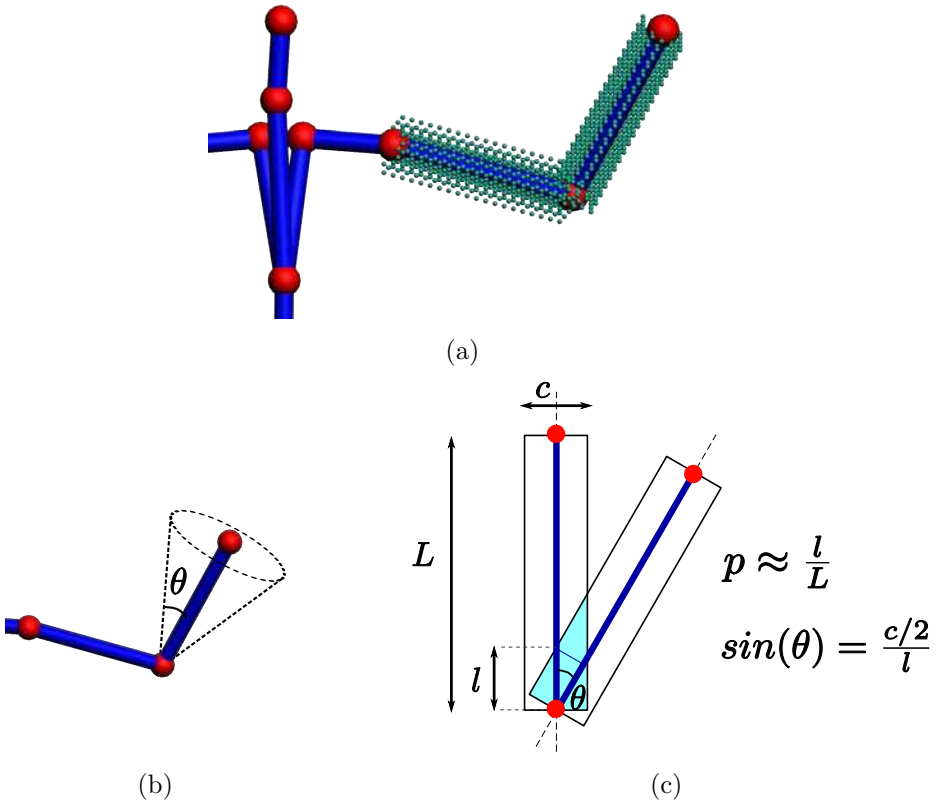


FIG. 4.16 – Raffinement de la position des membres après l'estimation par régression.
a : Les segments sont modélisés par des parallélépipèdes discrètisés. **b :** La position optimale du bras est recherchée en échantillonnant un cône d'angle θ autour de la position estimée. **c :** Evaluation de l'angle du cône en fonction du taux de remplissage initial p du parallélépipède.

4.4.3 Evaluation expérimentale du gain du raffinement

La figure 4.17 présente les résultats obtenus avec notre méthode de raffinement dans le cas de données de synthèse. La première colonne de la figure montre la posture réelle de l'avatar. La deuxième colonne présente les résultats de l'estimation par régression : le squelette estimé est superposé aux voxels de l'enveloppe. Les deux dernières colonnes montrent la pose obtenue après correction par notre méthode de raffinement, avec deux points de vue différents. Sur ces exemples, le raffinement a bien permis d'atténuer les imprécisions de l'estimation initiale et le résultat final est tout à fait satisfaisant : le squelette a bien été repositionné à l'intérieur de l'enveloppe. Nous verrons dans le chapitre 5 que de bons résultats ont également été obtenus sur des données réelles.

Dans le cas de données de synthèse, nous disposons de la vérité terrain sur la pose, ce qui nous permet d'évaluer quantitativement le gain en précision obtenu par raffinement. Le tableau 4.9 présente les erreurs moyennes sur les positions des coudes et des mains sur la séquence de 500 exemples de gestes de l'avatar 1, calculées avec les poses obtenues avant et après raffinement. On peut constater que le gain en précision est significatif puisque globalement l'erreur 3D a presque été divisée par deux.

Il reste néanmoins certains cas de figure où le raffinement n'a pas permis de corriger la pose estimée. Pour certains exemples, le raffinement a même dégradé la pose initiale. Quelques exemples sont donnés sur la figure 4.18. Sur le premier exemple, la pose initiale du bras droit était assez correcte, mais on peut penser que le raffinement du bras a décalé la position de la main, et que l'avant-bras s'est positionné sur des voxels supplémentaires de la silhouette, dont la présence est due à des artéfacts. Sur la 2^e ligne, la position de l'avant-bras a été mal estimée par la régression, et le raffinement a là aussi empiré les choses puisque l'avant-bras s'est déplacé sur les voxels du bras. Enfin, la dernière ligne de la figure présente un exemple pour lequel la posture initialement estimée est incorrecte car le bras gauche s'est placé sur le torse. Pour cet exemple, la raffinement n'a pas non plus permis de corriger la pose, car si le bras est placé au niveau du torse, le taux de remplissage du parallélépipède est tout de même correct. De tels exemples permettent d'expliquer le fait qu'il reste une erreur moyenne non négligeable sur l'ensemble des exemples de la base de tests dans le tableau 4.9.

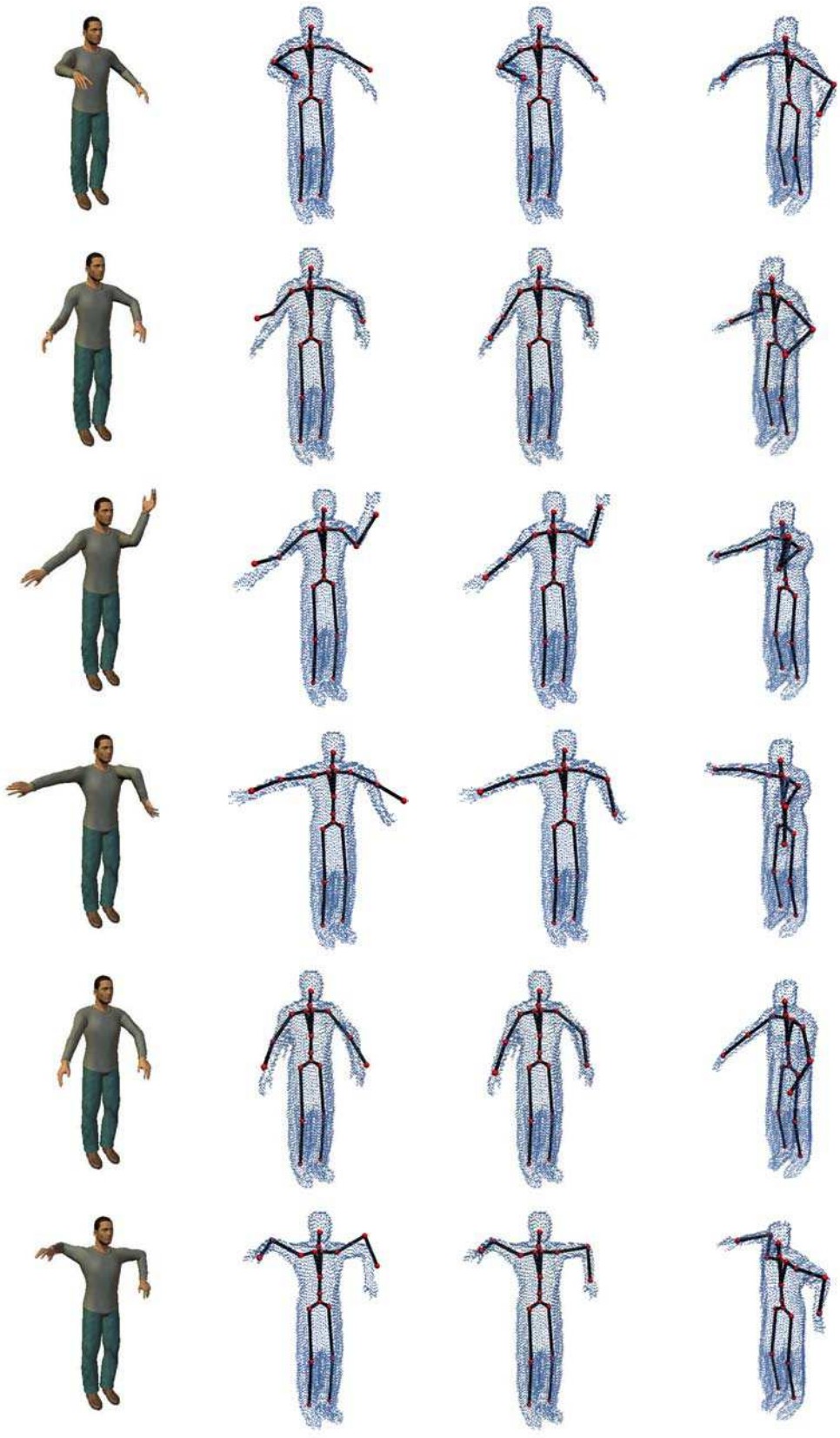


FIG. 4.17 – Exemples de poses corrigées sur des données synthétiques.

	avant raffinement	après raffinement
coude gauche	6.9	3.3
coude droit	7.4	3.9
main gauche	12.3	7.0
main droite	13.5	8.8
moyenne	10.0	5.7

TAB. 4.9 – Erreurs moyennes (en *cm*) sur les positions 3D des coudes et des mains avant et après raffinement sur une séquence de 500 exemples.

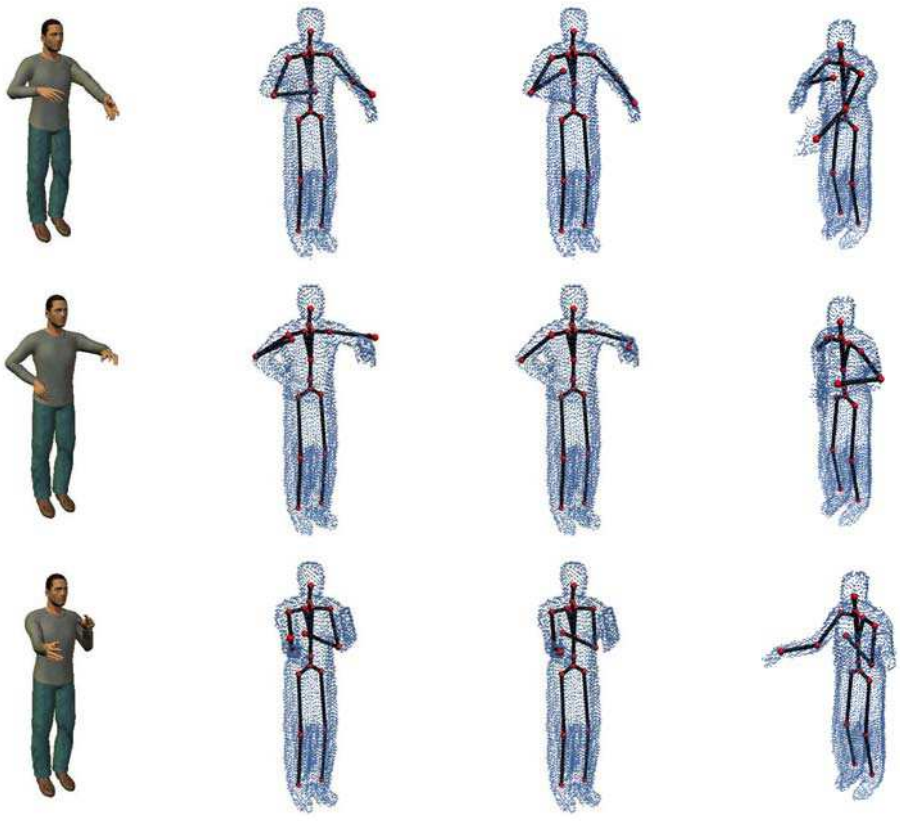


FIG. 4.18 – Exemples de poses mal estimées qui n'ont pas été corrigées par le raffinement.

4.4.4 Limites de la méthode et perspectives

Nous avons proposé une méthode simple de raffinement de la pose, qui vient compléter l'estimation par régression et affiner les résultats. Comme on vient de le voir, notre méthode de raffinement est encore très perfectible car elle ne permet de pas gérer toutes les situations. En particulier, rien n'empêche deux membres du corps de venir se placer sur les mêmes voxels de l'enveloppe, ou les segments des bras d'être ramenés vers le torse. La méthode fait l'hypothèse que la pose a été correctement estimée et fonctionne pour de petites imprécisions. Parmi les perspectives d'amélioration de la méthode, un modèle plus complet pourrait être introduit, dans lequel chaque partie du corps serait explicitement modélisée, y compris le torse (par exemple par des cylindres et des parallélépipèdes). Pour empêcher deux parties du modèle de se positionner sur les mêmes voxels, une solution pourrait être de construire une fonction de coût basée à la fois sur le taux de remplissage par des voxels des différentes parties du modèle, et sur le nombre de voxels de l'enveloppe couverts par le modèle. Ainsi, une configuration où deux membres du corps se chevauchent serait pénalisée.

4.5 Conclusion

Nous avons présenté dans cette partie notre méthode d'estimation par régression, reposant sur un apprentissage hors ligne de la relation entre le descripteur de la forme 3D et la pose du corps à partir d'une base d'entrainement. Différentes formes de régression ont été comparées de manière quantitative.

L'estimation par régression présente l'avantage de réduire les calculs nécessaires à l'évaluation de la pose du corps car elle intègre des informations a priori permettant une prédiction directe de la configuration du corps à partir des observations qu'elle a générées. Mais elle ne permet en général pas d'atteindre la même précision qu'une méthode qui ajuste un modèle aux données image. C'est pourquoi il peut être intéressant de combiner les deux types d'estimation, en complétant le résultat de la régression par l'ajustement d'un modèle. Une méthode de raffinement simple a été proposée, et testée sur des données de synthèse pour améliorer de la pose des bras. Les résultats présentés sont encourageants et suggèrent que la méthode de raffinement pourrait être prolongée pour améliorer plus globalement la pose de tous les membres du corps.

Chapitre 5

Evaluations expérimentales et discussion

Cette partie présente des résultats obtenus avec notre système d'estimation, à la fois sur des séquences de synthèse et sur des images réelles. Avec les données synthétiques, la connaissance de la vérité terrain sur les paramètres de la pose nous permet d'évaluer quantitativement l'influence de certains facteurs sur la précision du système d'estimation, et de régler les paramètres de la régression de façon optimale pour traiter au mieux les données réelles. Pour les estimations sur données réelles, l'appréciation de la qualité des résultats est plus subjective : dans notre cas, elle se fait en comparant visuellement la pose de la personne dans les images à celle d'un avatar animé avec les angles (ou les points 3D) estimés par régression, ou d'un squelette superposé aux données images ou aux voxels reconstruits. Même s'ils sont qualitatifs, ces résultats permettent de valider notre méthode, d'une part en montrant qu'elle s'applique à des personnes réelles, et donc s'étend à des morphologies plus générales et plus réalistes que les avatars d'un logiciel de synthèse, et d'autre part qu'elle est suffisamment robuste aux imperfections du système d'acquisition réel (synchronisation des caméras, calibrage imprécis, qualité des images...) et de l'algorithme d'extraction des silhouettes. Dans notre méthode, un point délicat est de parvenir à faire fonctionner l'estimation sur des données réelles à partir d'un apprentissage réalisé sur des données totalement synthétiques, encore une fois parce qu'il n'est pas garanti que nos avatars soient suffisamment variés et réalistes, et parce que les silhouettes synthétiques ne contiennent pas le bruit des images réelles. Cette difficulté est accentuée par l'utilisation de techniques de régression à noyaux comme méthode d'estimation : si le descripteur d'un exemple est trop éloigné de ceux

des exemples présents dans la base d'apprentissage, et donc des éléments sélectionnés comme vecteurs support, toutes les fonctions noyaux du modèle deviennent proches de 0 et le régresseur risque de retourner dans tous les cas une position "neutre" (celle obtenue lorsque tous les noyaux sont nuls et qu'il ne reste que les coefficients constants dans le modèle). Les paramètres du modèle de régression sont par ailleurs ajustés sur des données parfaites, et il n'est pas évident que les poids calculés s'appliqueront ensuite à des données bruitées. Le problème serait sans doute réduit si la méthode reposait sur une recherche des plus proches voisins d'un exemple dans la base, puisqu'on chercherait alors les exemples de la base les plus proches sans se préoccuper de la distance effective de l'exemple aux éléments de la base, mais les avantages de la régression (compacité des modèles, capacités de généralisation permettant d'utiliser des bases beaucoup moins importantes) seraient perdus. Il est donc important de bien prendre en compte ces difficultés dans la phase de reconstruction 3D et dans la conception du descripteur.

Ce chapitre est composé de trois parties. Le premier paragraphe présente des évaluations sur données synthétiques de différents facteurs pouvant jouer sur la qualité de l'estimation, comme le type de descripteur utilisé, la résolution de la reconstruction en voxels, le nombre et le positionnement des caméras. Une comparaison de la précision de notre méthode par rapport à celles des approches similaires proposées dans la littérature est aussi exposée. Dans le deuxième paragraphe sont présentés des résultats obtenus sur des séquences d'images réelles. La dernière partie est une discussion sur les résultats obtenus et les perspectives d'amélioration des performances de notre méthode.

5.1 Evaluations sur données synthétiques

5.1.1 Evaluations du descripteur en régression

Nous reprenons dans cette section l'évaluation des paramètres du descripteur abordée dans le paragraphe 3.4 du chapitre 3. Différentes variantes du descripteur (obtenues en changeant les valeurs du lissage, le nombre de couches verticales, de divisions angulaires ou radiales, etc.) sont testées en estimant la précision de la pose qu'elles permettent d'obtenir par régression. La base d'exemples utilisée est la base de gestes présentée au paragraphe 4.3.3, comprenant différentes postures obtenues en faisant varier les angles

des bras sur les 8 avatars. Les silhouettes des avatars sont obtenues à partir du système de 5 caméras de la figure 3.15(a). Comme dans les évaluations précédentes, chaque avatar est successivement retiré de la base d'apprentissage, et ses postures sont utilisées comme exemples tests. Les régresseurs utilisés sont des modèles linéaires dont les vecteurs support ont été choisis aléatoirement dans la base d'apprentissage. Comme précédemment, le même ensemble d'exemples est utilisé pour comparer deux méthodes, pour éviter qu'une méthode puisse être considérée comme pénalisée par le choix des vecteurs support. Les essais portant sur les gestes, seules les composantes correspondant à la moitié supérieure du descripteur sont prises en compte. Comme il a été dit au chapitre 3, il est difficile de dissocier l'influence d'un paramètre sur la précision de l'estimation de celle des autres paramètres, car ces différents facteurs (en particulier les nombres de divisions spatiales du descripteur) dépendent les uns des autres, et plus encore en régression puisque chacun d'entre eux peut jouer sur la dimension du vecteur descripteur. L'idéal serait d'étudier toutes les combinaisons possibles de paramètres, mais cela représenterait une quantité trop importante de tests. Dans nos essais, nous sommes partis d'une configuration par défaut du descripteur (avec 10 couches verticales, respectivement 4,12 et 16 divisions angulaires sur les 3 couches radiales et un lissage horizontal de $\sigma = 3$ voxels), et ces paramètres ont été ajustés au fur et à mesure des expérimentations.

Evaluation des paramètres de lissage

Les expériences du chapitre 3 n'ont pas permis de mettre en évidence de manière claire la valeur optimale de l'écart-type dans le lissage des couches horizontales du descripteur. Dans cette partie, nous reprenons donc des expériences similaires en calculant la précision obtenue en régression sur des séquences tests avec différentes valeur de lissage. Les résultats sur les 8 avatars sont donnés dans le tableau 5.1, et représentés graphiquement sur la figure 5.1. Ces tests sont effectués pour des reconstructions ayant une résolutions de 128 voxels pour 2 m ; si la résolution était différente, il conviendrait d'adapter la valeur de l'écart-type (exprimée ici en voxels) en fonction de la taille des voxels (voir paragraphe 5.1.2).

Les courbes obtenues avec les 8 avatars ne présentent pas toutes le même comportement, mais globalement on peut noter une valeur minimale de l'erreur autour de $\sigma = 7$ ou 8 voxels. Comme le temps de calcul du descripteur augmente avec l'écart-type, nous avons privilégié dans nos tests ultérieurs un lissage d'écart-type $\sigma = 7$ voxels.

écart-type σ	0	1	2	3	4	5	6	7	8	9	10
avatar 1	11.6	11.4	11.0	10.6	9.9	9.5	9.0	8.9	9.2	9.5	9.8
avatar 2	15.5	15.2	14.7	14.4	14.1	13.9	13.5	13.3	13.2	13.6	15.4
avatar 3	10.9	10.7	10.5	10.3	9.8	9.5	9.2	8.9	8.4	8.0	7.7
avatar 4	13.5	13.3	13.0	12.7	12.2	11.9	11.6	11.6	11.7	11.8	12.7
avatar 5	11.0	10.9	10.6	10.3	9.9	9.6	9.3	9.0	8.9	9.0	9.6
avatar 6	12.5	12.3	12.0	11.8	11.3	11.1	10.9	11.0	11.4	12.0	14.2
avatar 7	12.3	12.2	11.9	11.6	11.2	10.9	10.5	10.3	10.2	10.2	10.8
avatar 8	11.4	11.2	10.9	10.5	10.0	9.8	9.5	9.4	9.1	8.9	8.9
moyenne	12.3	12.1	11.8	11.5	11.0	10.8	10.4	10.3	10.2	10.4	11.1

TAB. 5.1 – Erreurs moyennes (en *cm*) sur les positions 3D des coudes et des mains pour différentes valeurs de l'écart-type dans le lissage 2D des couches horizontales du descripteur.

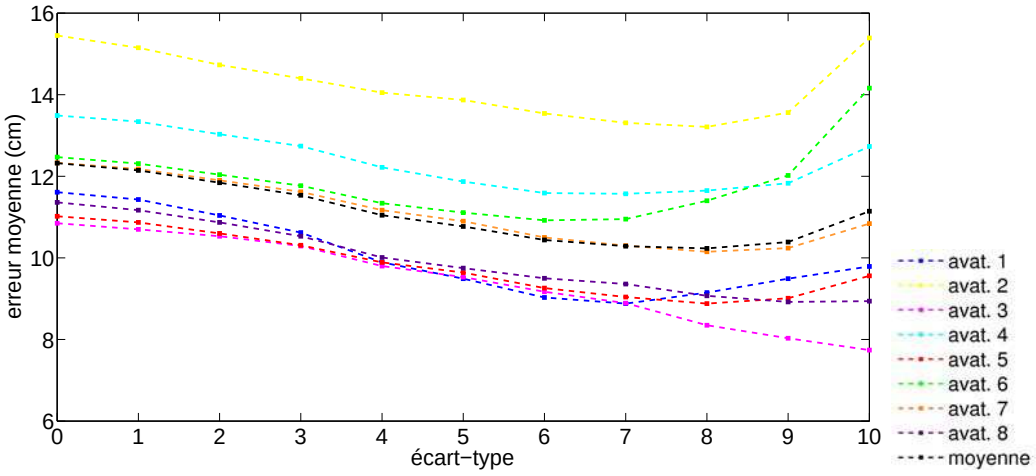


FIG. 5.1 – Erreurs moyennes sur les positions des coudes et des mains pour les 8 avatars en fonction du lissage horizontal dans le descripteur.

nombre de tranches	4	6	8	10	12	14	16
avatar 1	11.6	10.7	10.6	10.6	10.6	11.1	11.7
avatar 2	15.1	14.0	14.3	14.8	14.9	15.2	15.2
avatar 3	9.8	9.4	9.8	10.2	10.7	10.9	10.8
avatar 4	12.6	12.7	12.1	12.7	12.5	12.2	12.3
avatar 5	11.4	9.9	9.9	10.3	10.5	10.8	10.9
avatar 6	11.1	10.9	9.9	11.9	11.4	10.4	10.5
avatar 7	10.7	10.6	11.2	11.4	11.9	10.8	10.8
avatar 8	11.1	10.3	10.2	10.5	10.5	10.9	11.0
moyenne	11.7	11.1	11.0	11.5	11.6	11.5	11.7

TAB. 5.2 – Erreurs moyennes (en cm) sur les positions 3D des coudes et des mains obtenus en faisant le nombre de tranches verticales dans le descripteur.

Influence du nombre de divisions verticales

Contrairement aux courbes présentées dans le chapitre 3, les résultats des tests visant à évaluer les nombres optimaux de divisions dans le descripteur sont ici influencés par le fait qu'un découpage trop fin du cylindre produit un vecteur de dimension plus élevé et risque d'être pénalisé dans la régression. Les résultats des tests avec des nombres de couches verticales différents sont donnés dans le tableau 5.2, et les erreurs obtenues avec les 8 avatars sont représentées sur la figure 5.2 (les essais ont été réalisés avec un lissage vertical de $\sigma = 3$ voxels).

On peut remarquer que les courbes ont des allures et des valeurs de minima différentes selon les avatars, et que la valeur de l'erreur en fonction du nombre de couches ne varie pas toujours régulièrement. Ce comportement peut s'expliquer par le fait que les avatars ont des physionomies différentes (par exemple la taille des bras), et que les séparations entre les couches verticales doivent tomber à des hauteurs différentes sur le corps. La courbe moyenne semble néanmoins présenter un minimum pour $n_t = 8$ divisions verticales.

Influence du nombre de divisions angulaires

On étudie ici l'influence du nombre de divisions angulaires sur les 3 couches radiales du descripteur sur la précision de la pose des bras. Différentes combinaisons de ces trois valeurs ont été testées et les résultats de ces essais sont reportés dans le tableau 5.3 (le descripteur contient 8 divisions verticales et le lissage horizontal est de $\sigma = 3$ voxels). L'erreur minimale est

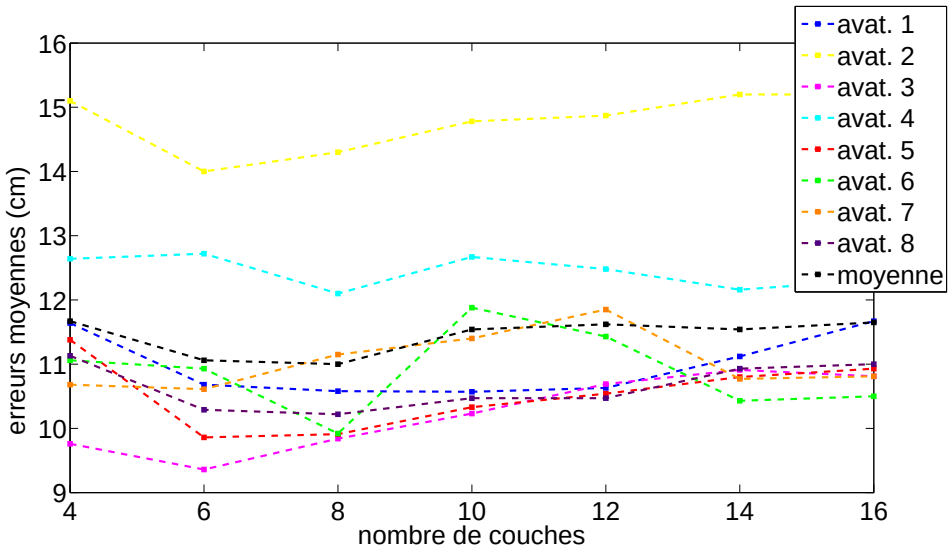


FIG. 5.2 – Erreurs moyennes sur les positions des coudes et des mains pour les 8 avatars en fonction du nombre de divisions verticales dans le descripteur.

atteinte pour la combinaison $8 - 12 - 16$.

Sélection des composantes du descripteur

L'objectif de cette section est d'étudier l'impact du choix des composantes du descripteur pour estimer un paramètre particulier. Dans le cas de la reconnaissance des gestes de deux bras, si le descripteur est recalé sur l'orientation du torse, il est en effet possible d'identifier les composantes de notre descripteur les plus utiles pour estimer la pose d'un bras. Le but de cette sélection est de ne garder que les éléments pertinents du descripteur et d'éliminer les informations superflues qui représentent un bruit pour le régresseur et peuvent perturber l'estimation. On peut également penser que le découpage du descripteur en régions correspondant à différentes parties du corps peut permettre de reconnaître une gamme de poses plus étendue, puisque les différents membres du corps sont rendus indépendants dans l'apprentissage et qu'un nombre plus important de combinaisons pourront être reconnues. L'inconvénient de cette opération est que plusieurs régresseurs doivent être entrainés pour chacune des régions du corps qui ont été isolées. Par exemple dans le cas des deux bras, deux apprentissages sont réalisés, un pour le bras gauche et un pour le bras droit.

Le descripteur utilisé pour les tests contient 8 tranches verticales, 8,12

nombre de divisions	4-4-4	4-8-12	4-12-16	8-8-8	8-12-16	12-12-12	16-16-16
avatar 1	15.3	10.9	10.6	11.6	10.4	10.7	10.4
avatar 2	18.9	14.0	14.3	14.8	14.1	14.1	14.3
avatar 3	13.0	10.0	9.8	10.7	9.9	10.2	10.1
avatar 4	18.2	12.3	12.1	13.4	12.1	12.3	12.2
avatar 5	16.1	10.4	9.9	11.2	9.9	10.4	10.0
avatar 6	12.3	10.1	9.9	10.8	9.8	10.1	9.9
avatar 7	13.9	11.3	11.2	12.2	11.1	11.3	11.1
avatar 8	16.5	10.2	10.2	11.0	10.2	10.3	10.3
moyenne	15.5	11.1	11.0	12.0	10.9	11.2	11.0

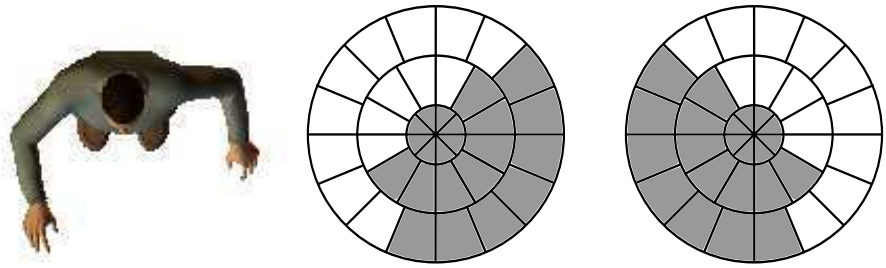
TAB. 5.3 – Erreurs moyennes (en cm) sur les positions 3D des coudes et des mains en fonction du nombre de divisions angulaires dans le descripteur. Les chiffres indiqués dans la première ligne donnent les nombres de divisions angulaires respectifs sur les trois couches radiales, du centre vers l'extérieur.

et 16 divisions angulaires et le lissage est de $\sigma = 7$ voxels (descripteur optimal des expérimentations précédentes). Deux méthodes d'estimation ont été comparées : dans la première expérience, toutes les composantes de la partie supérieure du descripteur ont été utilisées pour estimer la pose des deux bras par un unique apprentissage. Dans la seconde expérience, des composantes du descripteur ont été sélectionnées pour estimer la pose de chacun des deux bras. Ces composantes sont représentées sur la figure 5.3. Les résultats de ces essais sont donnés dans le tableau 5.4.

On peut constater que la sélection des composantes du descripteur a permis d'obtenir un gain en précision pour presque tous les avatars (excepté l'avatar 2) ; elle présente donc un intérêt pour l'estimation de la pose des bras.

Conclusion des expériences sur le descripteur

Ces expérimentations sur le descripteur confirment les résultats qui avaient été obtenus dans le chapitre 2, et permettent de mettre en évidence plus précisément les paramètres optimaux pour la régression. Chacun de ces réglages permet d'obtenir une légère augmentation de la précision sur les données de synthèse, mais on peut penser que leur combinaison permet au final d'obtenir un descripteur plus fiable pour gérer le cas difficile des données réelles.



(a) (b) (c)

FIG. 5.3 – Visualisation des composantes du descripteur sélectionnées pour évaluer la position de chacun des bras.
a : exemple de pose vue d'en haut. **b :** composantes sélectionnées pour estimer les paramètres du bras gauche. **c :** composantes sélectionnées pour estimer les paramètres du bras droit.

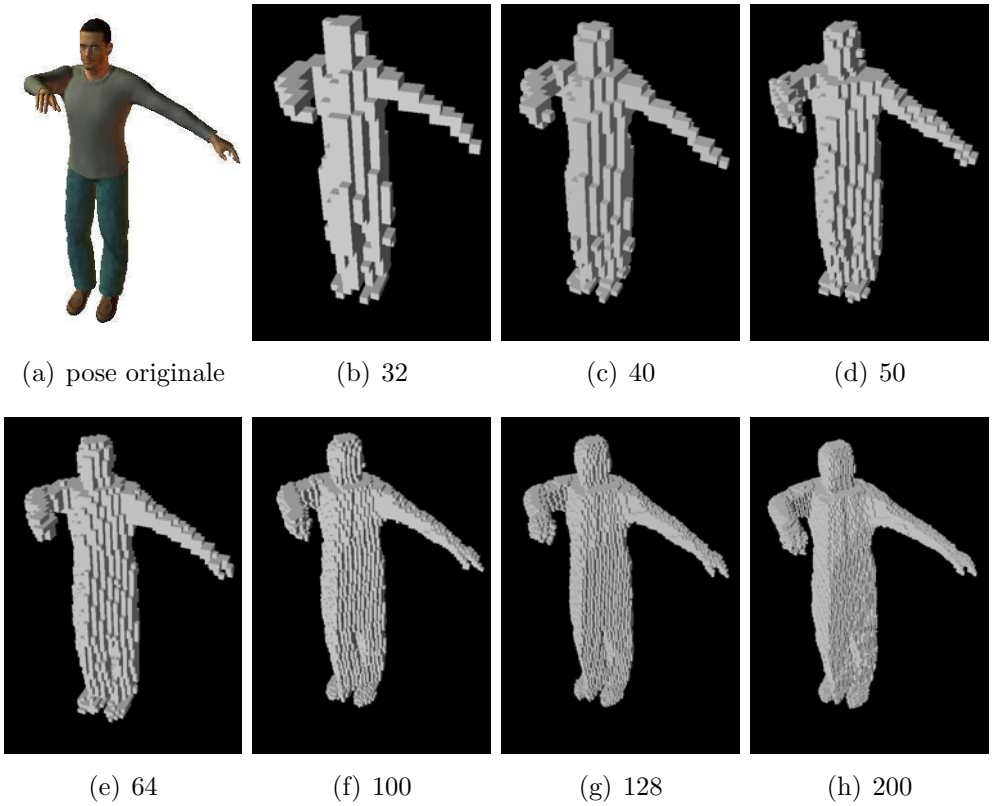
	sans sélection des composantes	avec sélection des composantes
avatar 1	8.8	8.8
avatar 2	12.9	13.9
avatar 3	8.7	6.5
avatar 4	11.1	10.1
avatar 5	8.7	8.4
avatar 6	9.0	7.9
avatar 7	10.2	8.5
avatar 8	8.9	8.1
moyenne	9.8	9.0

TAB. 5.4 – Amélioration en précision obtenue en sélectionnant les composantes du descripteur pour évaluer les positions des bras.

5.1.2 Influence de la résolution de la reconstruction

Cette section s'intéresse à l'influence de la résolution de la reconstruction, c'est-à-dire à la finesse des voxels, sur la qualité de l'estimation de la pose. L'idée naturelle est que plus la résolution est fine, et plus l'information capturée sur la forme 3D par l'algorithme de reconstruction est importante. Dans notre cas, l'appartenance d'un voxel à l'enveloppe est validée simplement par la projection de son centre dans les images, et cette approximation oblige à utiliser une résolution assez élevée. De plus, dans le cas des données réelles, on peut penser qu'une résolution fine peut permettre d'atténuer les effets des erreurs d'extraction des silhouettes : si le centre d'un voxel tombe sur un pixel mal labellisé à cause d'une erreur de segmentation (par exemple s'il est classé comme appartenant au fond alors qu'il se trouve normalement à l'intérieur de la silhouette), l'erreur induite sur la reconstruction 3D peut être compensée si les voxels voisins ont été correctement classés. Choisir une résolution trop élevée implique à l'inverse des temps de calcul importants, à la fois pour l'algorithme de reconstruction (N^3 voxels sont projetés dans les images) et pour le calcul du descripteur (chaque voxel doit être classé dans un des secteurs du cylindre), et n'est pas nécessairement pertinent pour gagner en précision, d'une part parce que l'information 3D à laquelle on peut accéder est limitée par la résolution et la qualité des images (dans le cas de données réelles), et parce que l'information contenue dans la forme 3D sera ensuite lissée dans le calcul du descripteur. Par ailleurs, la résolution de la reconstruction devrait logiquement tenir compte de l'échelle de la forme 3D reconstruite, donc de la taille de la personne dans notre cas. Ainsi, dans [106], la taille des voxels est adaptée à la taille de la personne dont l'enveloppe est reconstruite en fixant $l_{res} = \frac{h_{sub}}{60}$, où la taille du sujet h_{sub} est mesurée en traitant la première vue de la séquence avec une résolution par défaut ($l_{res} = 0.05m$). Cette technique permet en outre d'obtenir un premier niveau d'invariance à l'échelle de la forme 3D.

Pour évaluer les différences de précision de l'estimation de pose en fonction de la taille des voxels, des essais ont été réalisés avec différentes résolutions du cube (de 2 m de côté) : 32, 40, 50, 64, 100, 128 et 200 (la dimension des voxels correspondante est donnée dans le tableau 5.5). Ces tests sont réalisés sur les données de synthèse utilisées dans les paragraphes précédents et ne tiennent donc pas compte des imprécisions qui pourraient être introduites dans le cas d'un système réel. La figure 5.4 montre un exemple d'une même pose reconstruite avec ces différentes résolutions. Le descripteur utilisé dans ces tests contient 8 couches verticales, et respectivement 8, 12 et 16 divisions



(a) pose originale (b) 32 (c) 40 (d) 50

(e) 64 (f) 100 (g) 128 (h) 200

FIG. 5.4 – Reconstruction (avec 5 caméras) de la même pose à différentes résolutions.

sur les 3 couches radiales, et le lissage vertical choisi pour la résolution 128 est $\sigma = 7$ voxels. Pour les autres résolutions, l'écart-type du lissage a été réglé de façon à ce que l'étendu spatiale des votes d'un voxel soit équivalente à celle obtenue avec la résolution 128.
Les résultats obtenus en régression avec les 8 avatars sont réunis dans le tableau 5.5, et la moyenne de l'erreur obtenue est représentée graphiquement sur la figure 5.5. Il semble qu'à partir de la résolution 64 (voxels pour 2m), la résolution soit suffisante pour reconstruire la pose des bras avec une bonne précision. Ce résultat est intéressant car les temps ce calcul de la reconstruction et du descripteur avec cette résolution peuvent permettre d'envisager un fonctionnement en temps réel de la méthode. Pour traiter les données réelles, une résolution de 128 a toutefois été gardée.

résolution	32	40	50	64	100	128	200
côté d'un voxel (cm)	6.25	5	4	3.13	2	1.56	1
avatar 1	11.5	10.7	9.0	9.0	8.5	8.9	8.8
avatar 2	15.5	15.4	12.9	12.7	13.0	13.1	12.7
avatar 3	10.3	10.0	9.7	8.3	8.2	8.8	8.4
avatar 4	12.6	11.5	12.8	11.8	13.2	11.1	11.2
avatar 5	10.0	9.6	9.3	8.7	8.5	8.7	8.5
avatar 6	10.0	8.6	9.9	9.3	10.4	9.0	9.0
avatar 7	10.7	9.7	9.1	8.8	8.8	10.1	8.8
avatar 8	9.5	8.8	9.5	8.7	8.6	8.7	8.4
moyenne	11.3	10.6	10.3	9.7	9.9	9.8	9.5

TAB. 5.5 – Erreurs moyennes (en cm) sur les positions 3D des coudes et des mains pour différentes valeurs de la résolution de la reconstruction.

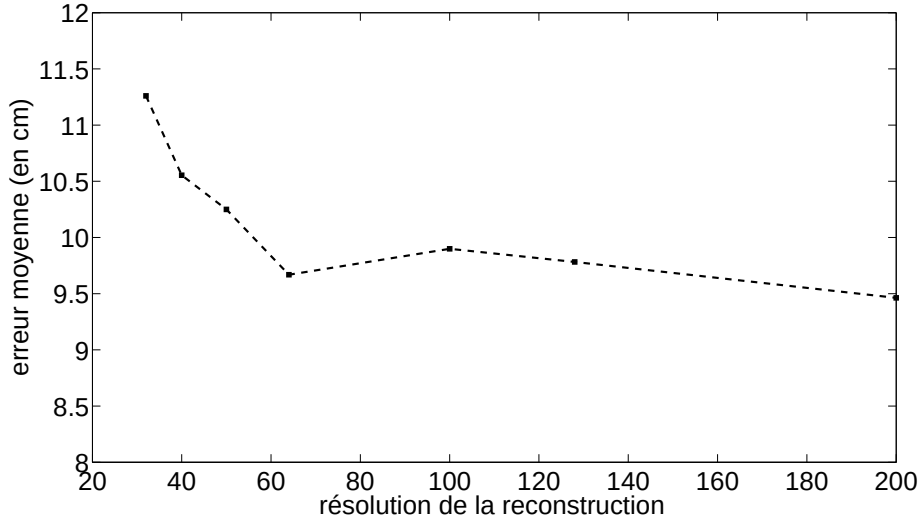


FIG. 5.5 – Erreurs moyennes sur les positions des coudes et des mains pour les 8 avatars en fonction de la résolution de la reconstruction.

5.1.3 Influence du nombre de caméras et de leur configuration spatiale

L'objectif de ce paragraphe est d'étudier l'influence du nombre des caméras et de leur positionnement sur la qualité de l'estimation. Comme il a déjà été dit dans les chapitres précédents, nous avons fait le choix de nous appuyer sur une reconstruction 3D pour combiner les différents points de vue, car celle-ci fusionne en un seul élément toutes les informations sur la géométrie de la scène et laisse espérer que le système d'estimation pourra être relativement indépendant de la configuration des caméras. Cependant, nous avons vu qu'un nombre insuffisant des caméras, ou des caméras mal positionnées, engendrent sur la silhouette 3D des artéfacts qui varient en fonction du placement des caméras.

Nous proposons dans cette partie deux types d'expériences : la première vise à évaluer la précision de notre méthode en fonction du nombre de caméras utilisées pour l'apprentissage et les tests, et la deuxième à étudier l'impact d'un changement de configuration des caméras entre l'apprentissage et l'estimation. Pour ces tests, une base d'apprentissage a été réalisée avec l'avatar par défaut de Poser 6, et des mouvements de bras analogues à ceux des essais des paragraphes précédents. La base d'apprentissage contient 2000 exemples, et la base de tests 500 exemples. L'apprentissage est effectué via un modèle linéaire comprenant 700 vecteurs support.

Influence du nombre de caméras

Pour ces expériences, un système comprenant un grand nombre de caméras (13 caméras, voir figures 5.6(a) et 5.6(b)) a été construit, dans l'idée que l'enveloppe 3D qu'il permet de reconstruire doit être proche de l'enveloppe "idéale" du corps. Des sous-ensembles de ces 13 caméras ont ensuite été sélectionnés pour constituer différents systèmes d'acquisition :

- un système de 3 caméras (figure 5.6(c)),
- un système de 4 caméras au sol (figure 5.6(d)),
- deux systèmes de 5 caméras, avec chacun 4 caméras au sol et une caméra au plafond : le système de la figure 5.6(e) et la configuration régulière de la figure 5.8,
- un système de 7 caméras : 6 caméras au sol et une caméra au plafond (figure 5.6(f)),
- le système de 13 caméras (figures 5.6(a) et 5.6(b)).

système de caméras	erreur 3D moyenne (cm)
3 caméras	8.6
4 caméras	7.2
5 caméras (figure 5.6(e))	7.8
5 caméras (figure 5.8)	7.7
6 caméras	8.1
7 caméras	7.3
12 caméras	7.7
13 caméras	7.2

TAB. 5.6 – Erreurs moyennes (en cm) sur les positions 3D des coudes et des mains en fonction du nombre de caméras utilisé pour l'apprentissage et les tests.

Deux systèmes de 5 caméras ont été utilisés dans les tests : le premier (figure 5.6(e)) est une configuration proche de celle que nous avons utilisée pour les séquences réelles (les caméras au sol sont aux 4 coins d'un rectangle), et le deuxième (figure 5.8) est une configuration régulière (les caméras au sol sont aux 4 coins d'un carré). Ces deux systèmes nous permettent de déterminer si le fait de ne pas choisir une configuration régulière est pénalisant pour les résultats. La figure 5.7 montre un exemple des reconstructions 3D obtenues à partir d'une même posture avec les différents systèmes de caméras. Les silhouettes reconstruites avec les systèmes privés de la caméra au plafond contiennent de gros artéfacts au niveau des bras.

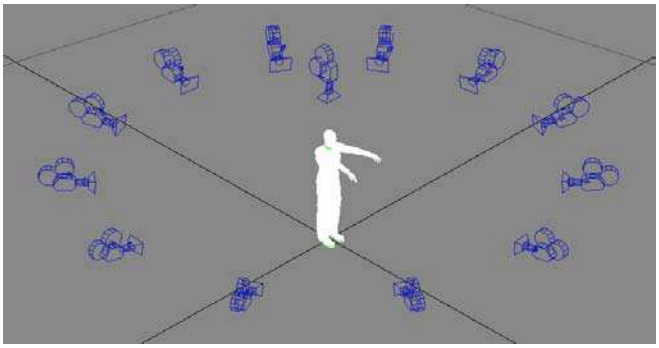
Le tableau 5.6 donne les précisions obtenues sur la base de tests pour l'estimation des positions des coudes et des mains de l'avatar.

Influence d'un changement de configuration des caméras entre l'apprentissage et les tests

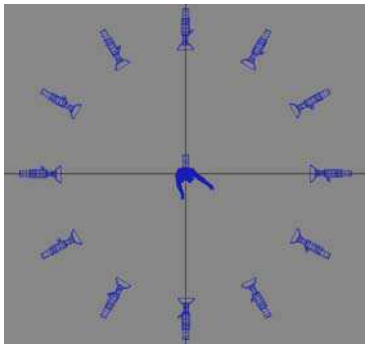
A partir du système de 5 caméras de configuration régulière (4 caméras au sol aux sommets d'un carré), différents systèmes de caméras "bruités" sont créés : pour chaque prise de vue, la position de chacune des caméras au sol est écartée d'un angle aléatoire compris entre $-30°$ et $+30°$ (voir figure 5.7) par rapport à la position de base.

Pour déterminer l'influence d'un changement de configuration des caméras entre l'apprentissage et les tests, différentes situations ont été comparées :

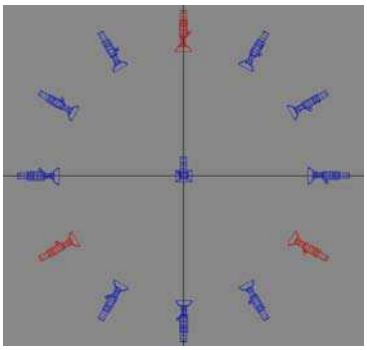
– le premier cas, traité dans le paragraphe précédent, est celui où l'apprentissage et l'estimation sont effectuées avec le système de 5 caméras



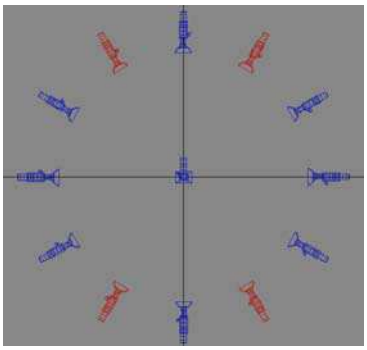
(a) Système de 13 caméras (vue perspective)



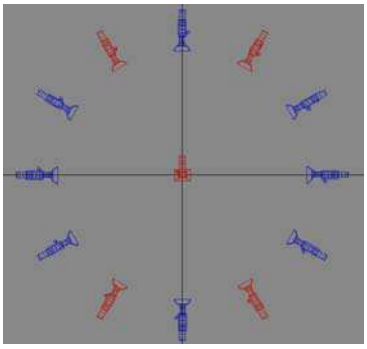
(b) 13 caméras (vue d'en haut)



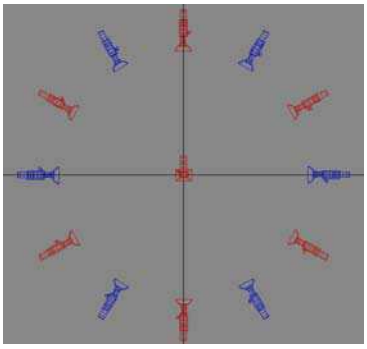
(c) 3 caméras (en rouge)



(d) 4 caméras (en rouge)



(e) 5 caméras (en rouge)

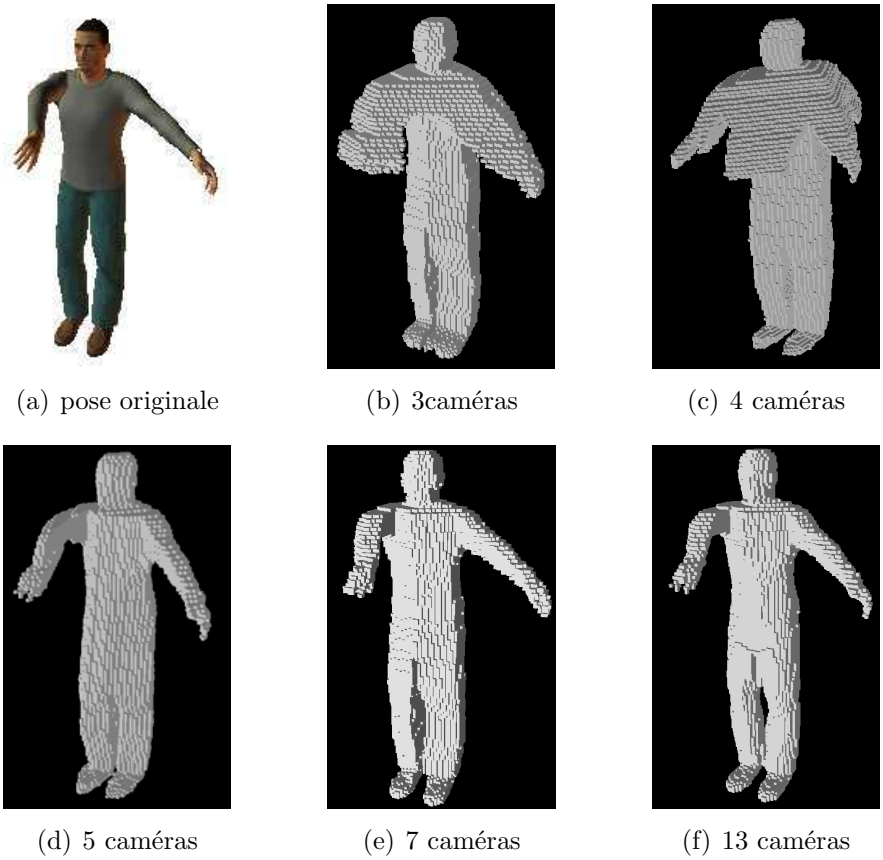


(f) 7 caméras (en rouge)

FIG. 5.6 – Les différents systèmes de caméras testés.

fixes,

– dans la deuxième expérience, l'apprentissage est réalisé avec le système de caméras fixes, mais le régresseur appris est testé avec des silhouettes



(a) pose originale (b) 3caméras (c) 4 caméras

(d) 5 caméras (e) 7 caméras (f) 13 caméras

FIG. 5.7 – Reconstructions de la même posture obtenues avec les différents systèmes de caméras.

reconstruites par les configurations bruitées,
- dans le troisième cas, l'apprentissage et les tests sont effectués avec les configurations bruitées, pour déterminer si un apprentissage avec des configurations variées permet de compenser le fait que le système n'est pas fixe entre l'apprentissage et l'estimation,
- dans la dernière expérience, l'apprentissage est effectué avec le système de 13 caméras, et les tests sur des configurations bruitées, l'idée étant de savoir si une silhouette 3D proche de l'enveloppe idéale peut permettre de fournir des estimations correctes à partir d'enveloppes contenant des artéfacts variables.

Les résultats de ces tests sont donnés dans le tableau 5.7. La même série d'expérience a également été réalisée en supprimant la caméra au plafond. Les résultats sont reportés dans le tableau 5.8.

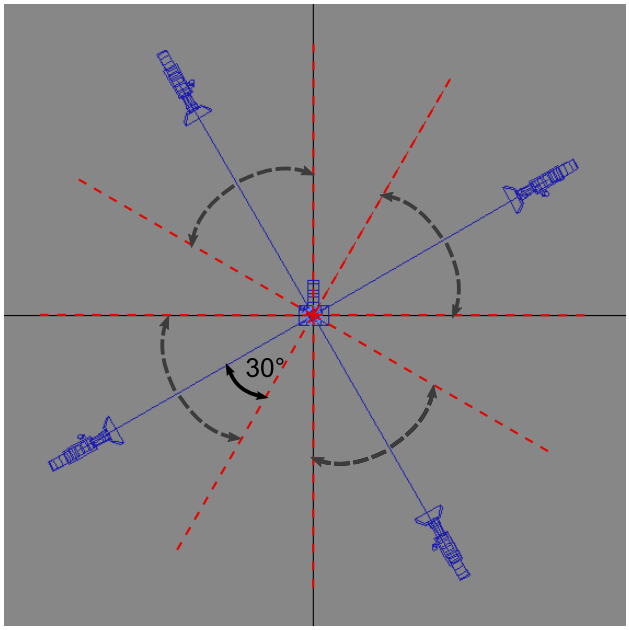


FIG. 5.8 – Positions possibles des 4 caméras au sol.

système caméras apprentissage	système caméras tests	erreur 3D moyenne
4 au sol fixes + 1 plafond	4 au sol fixes + 1 plafond	7.7
4 au sol fixes + 1 plafond	4 au sol avec bruit + 1 plafond	7.9
4 au sol avec bruit + 1 plafond	4 au sol avec bruit + 1 plafond	7.7
12 au sol fixes + 1 plafond	4 au sol avec bruit + 1 plafond	8.0

TAB. 5.7 – Erreurs moyennes (en *cm*) sur les positions 3D des coudes et des mains en fonction du système de caméras utilisé pour l'apprentissage et les tests.

système caméras apprentissage	système caméras tests	erreur 3D moyenne
4 au sol fixes	4 au sol fixes	7.9
4 au sol fixes	4 au sol avec bruit	16.3
4 au sol avec bruit	4 au sol avec bruit	10.6
12 au sol fixes	4 au sol avec bruit	12.1

TAB. 5.8 – Erreurs moyennes (en *cm*) sur les positions 3D des coudes et des mains en fonction du système de caméras utilisé pour l'apprentissage et les tests.

Conclusions

Les résultats du tableau 5.6 peuvent de prime abord sembler paradoxaux, puisqu'il ne semble y avoir que très peu de différence entre la précision atteinte avec un petit nombre de caméras et celle obtenue avec un système plus complet permettant de reconstruire une silhouette quasi-parfaite. Dans certains cas, la précision est même un peu meilleure avec un système comprenant moins de caméras. En particulier, la caméra au plafond, qui élimine une grosse partie des artéfacts, aurait plutôt tendance à gêner l'estimation. On s'attendrait au contraire à ce qu'un système aboutissant à une silhouette proche de l'enveloppe réelle supprime des ambiguïtés handicapantes pour la robustesse et la précision de l'estimation. Il est en fait très difficile d'évaluer le niveau réel d'ambiguïté d'une silhouette 3D. En examinant des reconstructions comme celles des figures 5.7(b) et 5.7(c), l'idée qui vient naturellement à l'esprit est que le système d'estimation n'arrivera pas à évaluer la position réelle des bras dans la masse des voxels reconstruits, et que n'importe quelle position d'un bras au milieu d'un artéfact qui lui correspond peut convenir au regard de la silhouette 3D. Ce n'est pourtant pas le cas, puisqu'une position du bras différente n'aurait pas généré les mêmes artéfacts. Il faut souligner que dans ces expériences les caméras ont été placées exactement de la même manière dans la phase d'apprentissage et dans les tests : les artéfacts ont donc été appris, modélisés lors de la phase d'entrainement. Au lieu d'être un handicap, ils peuvent au contraire constituer un élément supplémentaire sur lequel la régression va s'appuyer, car la matière en plus peut rendre la silhouette 3D plus discriminante. Au final, deux phénomènes doivent vraisemblablement se compenser : utiliser plus de caméras doit certainement limiter les ambiguïtés visuelles, mais les artéfacts peuvent aussi représenter une indication supplémentaire pour le régresseur.

Il semblerait alors logique que des méthodes s'appuyant sur les artéfacts des silhouettes soient fortement pénalisées par un changement de configuration des caméras entre l'apprentissage et les tests. C'est ce que nous avons cherché à mettre en évidence dans la 2^e série d'expériences. Comme la caméra située au plafond joue un rôle essentiel dans la présence ou non d'artéfacts sur la silhouette, la comparaison les deux séries d'expériences (tableau 5.7 avec la caméra au plafond et tableau 5.8 sans cette caméra) est assez instructive. Les résultats du premier tableau suggèrent que l'estimation n'a pas été réellement perturbée par la différence de positionnement des caméras au sol, dans le cas où la caméra au plafond a été prise en compte. En revanche, le deuxième tableau montre que, privée de cette caméra, l'estimation a beau-

coup perdu en précision lorsque les caméras au sol ont changé de position (l'erreur est deux fois plus élevée). L'analyse de ces résultats montre aussi que le fait d'intégrer l'incertitude sur la position des caméras dans l'apprentissage permet d'atténuer les imprécisions : on "apprend" ainsi au système à être robuste à des variations de calibrage.

En conclusion, les performances de notre système d'estimation sont satisfaisantes, soit dans le cas où les caméras sont calibrées de la même façon lors que la phase d'apprentissage que dans les tests, et même avec un nombre de caméras réduit, soit, si le positionnement des caméras varie, lorsque les silhouettes présentent peu d'artéfacts. Il faut toutefois rappeler que les expériences ont été menées ici avec un avatar statique : ni l'orientation, ni la position dans l'espace de capture ne varient. Si l'orientation de l'avatar changeait, les artéfacts générés pour une même pose avec différentes orientations varieraient aussi beaucoup. Après recalage du descripteur sur l'orientation du corps, une méthode qui s'appuie sur les artéfacts seraient sans doute pénalisée.

Plus généralement, la question du nombre et du positionnement optimal des caméras nécessiterait d'être approfondie. Pour un nombre de caméras donné, il existe certainement un positionnement des caméras idéal pour évaluer la posture. Il doit également exister un nombre minimal de caméras qui permet, si les caméras sont correctement réparties les unes par rapport aux autres, d'être à peu près indépendant du calibrage. Toutes ces données représenteraient des informations précieuses pour la mise en oeuvre pratique de notre système. Pour répondre à ces questions, il faudrait sans doute d'une part mesurer le rapport entre le placement des caméras et la qualité de la silhouette reconstruite (i.e. sa fidélité par rapport à l'enveloppe réelle), par exemple en quantifiant les artéfacts des silhouettes, et d'autre part déterminer dans quelle mesure la présence d'artéfacts sur la silhouette pénalise l'estimation par régression.

5.1.4 Comparaison de notre méthode à l'état de l'art

Nous reprenons dans cette section les données utilisées pour les tests du paragraphe 4.3.2. Ces données ont été produites à partir de la capture des mouvements de 3 personnes qui marchent en spirale, et sont disponibles en ligne [4]. Elles nous permettent de comparer les performances de notre système d'estimation à celles des méthodes présentées dans [7] et [106], puisque leurs auteurs utilisent ces mêmes données pour évaluer la précision de leurs approches. Ces deux références sont intéressantes car elles adoptent des ap-

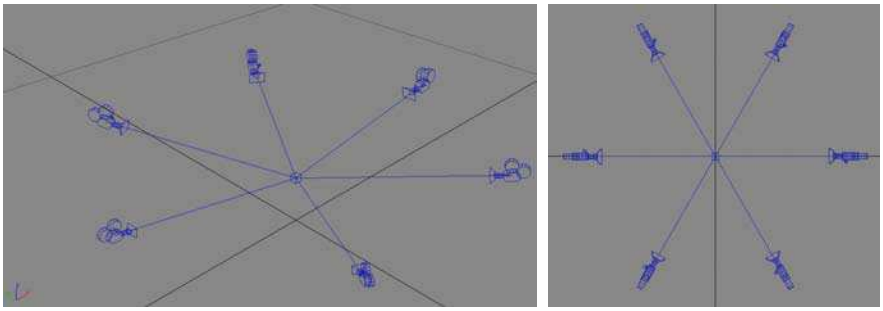


FIG. 5.9 – Système de 6 caméras utilisé pour les tests.

proches similaires à la nôtre : elles sont toutes les deux basées sur l'utilisation des silhouettes (en monoculaire dans [7] et via une reconstruction 3D en voxels pour [106]) et une régression. La comparaison avec [7] permet de mettre en avant les bénéfices d'une méthode d'estimation multi-vues par rapport au cas monoculaire, et avec [106] de mesurer les performances de notre descripteur par rapport aux histogrammes de Shape Context 3D. Les résultats de ces deux publications sont résumés dans le tableau 1 de [106]. Pour mieux nous comparer à [106], un système de 6 caméras circulaires, similaire à celui qui est utilisé pour l'apprentissage et les tests de l'article, a été employé. Ce système est représenté sur la figure 5.9.

Pour nos tests, l'avatar par défaut de POSER 6 a été animé avec les fichiers BVH de [4]. Comme dans les deux articles, l'une des séquences, comprenant 418 exemples, est utilisée comme séquence de test. La base d'apprentissage contient 2537 exemples. Dans notre cas, des SVM sont utilisés pour la régression.

Le tableau 5.9 présente les résultats obtenus avec les différentes méthodes. Comme on pouvait s'y attendre, les méthodes utilisant le système de 6 caméras atteignent une plus grande précision. Notre système semble aussi être plus précis que celui qui est présenté dans [106]. Cette comparaison doit cependant être considérée avec prudence, car les résultats peuvent dépendre de certains facteurs qui ne sont pas toujours précisés dans les différents travaux, comme le niveau de détail de la paramétrisation du corps, le placement des caméras et le déplacement du sujet dans leur champ de vision, l'avatar qui est animé (nous avons utilisé l'avatar par défaut de Poser 6, une version antérieure du logiciel semble être utilisée dans [7], et les auteurs de [106] animent un modèle du corps composé de sphères et d'ellipsoïdes)... Nous avons de plus utilisé des SVM alors que des MVRVM sont employés dans [106], et nos tests du chapitre précédent ont montré que les SVM sont un peu plus précis.

	corps entier	orientation	épaule gauche	jambe droite
[7]	6.0	17	7.5	4.2
[106]	5.2	8.8	6.3	3.2
notre approche	3.0	4.6	3.7	2.8
avec recalage	2.5	-	3.4	2.1

TAB. 5.9 – Comparaison des RMS des erreurs entre différentes approches sur une séquence synthétique de 418 exemples de marche en spirale. **1ère ligne :** résultats de l'approche monoculaire présentée par Agarwal et Triggs dans [7]. **2e ligne :** résultats présentés dans [106] avec 6 caméras circulaires. **3e ligne :** résultats obtenus avec notre méthode avec 6 caméras, sans recaler le descripteur sur l'orientation. **4e ligne :** même chose avec l'estimation en deux temps (recalage du descripteur sur l'orientation estimée pour les angles internes).

La figure 5.10 présente des courbes représentant les valeurs des angles estimés et de la vérité terrain le long de la séquence test, pour l'un des angles des jambes et pour l'orientation du torse. Nos résultats sont comparés aux courbes reportées dans [7], dans le cas de l'estimation monoculaire. On peut observer que dans notre cas, l'utilisation de plusieurs caméras a permis de lever certaines ambiguïtés qui apparaissent en monoculaire dans l'orientation du corps et des jambes. Comme on l'a vu déjà vu, une silhouette unique ne permet pas toujours de distinguer dans une posture de marche quelle jambe est placée devant l'autre, et le régresseur peut dans ce cas choisir la mauvaise solution ou retourner une solution moyenne représentant une compromis entre deux les possibilités. Ce phénomène se manifeste par exemple dans les courbes de la figure 5.10 par une inversion des pics dans l'estimation de l'angle de la jambe (aux alentours de la vue 80).

5.2 Résultats sur données réelles

Cette partie présente des résultats obtenus sur des séquences réelles tournées au laboratoire. Comme il a été dit en introduction, la difficulté a été de faire fonctionner l'estimation à partir d'un apprentissage réalisé sur des données synthétiques. D'un autre coté, le fait de pouvoir utiliser des bases artificielles est un gros atout sur le plan pratique, car il permet de réaliser facilement des bases très riches, contenant des mouvements très variés, d'ajuster et de compléter ces bases en fonction des poses que l'on souhaite

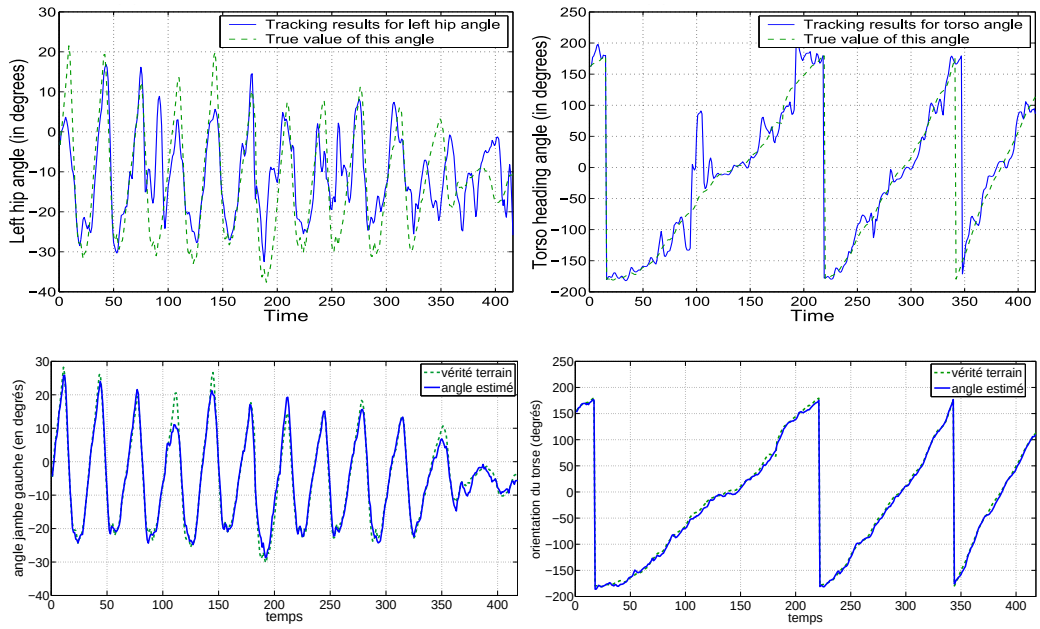


FIG. 5.10 – Angles estimés pour l'orientation du torse et la jambe gauche le long d'une séquence de marche synthétique, comparée à la vérité terrain. **haut :** résultats présentés dans [7]. **bas :** résultats obtenus avec notre méthode.

reconnaître, sans avoir à tourner de nouvelles séquences.

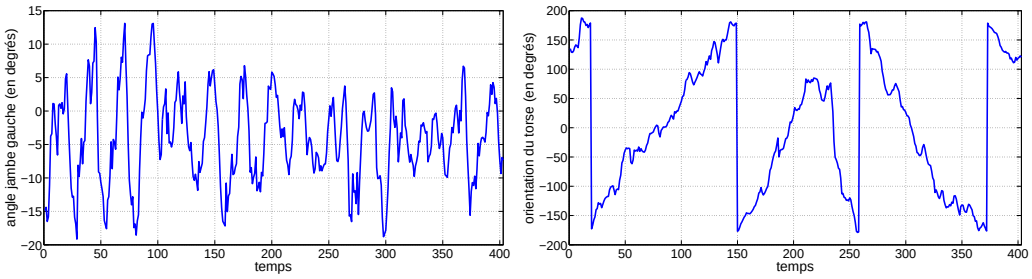
Le système de caméras utilisé pour ces acquisitions est analogue à celui qui est représenté sur la figure 3.15(a). Des essais ont été réalisés sur des séquences de marche et de gestes. Dans le premier cas, seules les 4 caméras au sol ont été prises en compte, et dans le second cas, la personne est fixe au centre de l'espace de capture, et les images des 5 caméras ont été employées.

5.2.1 Résultats sur des séquences de marche

Pour ces tests, une base d'apprentissage a été réalisée à partir des données de capture de mouvement de [4], qui contiennent les mouvements des différentes personnes qui marchent en spirale. Les 8 avatars ont été utilisés, et la base contient au total 2952 exemples. Cette base d'apprentissage contient des mouvements de marche simple et régulière, elle ne comporte pas par exemple de pose où des mouvements sont effectués avec les bras pendant la marche. Les mouvements ont été paramétrés par des angles, et l'estimation a été réalisée en deux temps : l'orientation du torse est d'abord estimée,

puis le descripteur est recalé sur cette orientation pour estimer les angles internes. Comme dans les essais du chapitre 4, un bruit de $\pm 10°$ est ajouté sur l'orientation du descripteur dans la base d'apprentissage pour gérer dans la 2^e étape les imprécisions dans l'estimation de l'orientation.

Des résultats sont présentés sur deux séquences de respectivement 403 et 241 images. Les courbes de la figure 5.11 montrent les angles estimés le long de la première séquence pour l'orientation du torse et l'angle de la jambe gauche. Ces courbes sont à rapprocher de celles de la figure 5.10 dans le cas de données de synthèse. Pour ces essais, on ne dispose pas de la vérité terrain pour évaluer la précision des résultats, mais on peut tout de même constater sur la figure 5.11(b) que les angles estimés pour l'orientation sont cohérents avec ce que l'on peut voir dans les vidéos : la personne observée marche le long d'une trajectoire circulaire, effectue un tour dans un sens, puis 3/4 de tour, puis se retourne et marche dans l'autre direction. On peut aussi retrouver sur la figure 5.11(a) le mouvement périodique de balancement de la jambe gauche, même si cette courbe présente bien plus de bruits et d'irrégularités que celles de la figure 5.10. Ces irrégularités peuvent être attribuées aux imprécisions du système d'estimation mais aussi au fait que la personne marche de manière bien plus irrégulière que dans les données de synthèse !



(a) (b)

FIG. 5.11 – Angles estimés pour l'orientation du torse et la jambe gauche le long d'une séquence de marche réelle de 403 vues.
a : Angle de la jambe gauche (avant-arrière). **b :** orientation du torse dans l'espace.

Les figures 5.12, 5.13 et 5.14 présentent des résultats obtenus pour des positions régulièrement réparties le long de la trajectoire circulaire de la personne. La première ligne de la figure montre la posture réelle de la personne observée par les caméras, la 2^e ligne la reconstruction 3D en voxels, et la dernière ligne un avatar animé avec les angles estimés. Le rendu de l'avatar est effectué avec une caméra virtuelle ayant le même point de vue que la

caméra réelle. La reconstruction en voxels est vue de la même caméra, mais est placée au centre du repère, ce qui peut expliquer la petite différence de points de vue sur certains exemples.

On peut constater sur quelques images de légères imprécisions sur l'orientation de l'avatar, et une mauvaise estimation sur la position des jambes dans une certaine zone de la pièce (dernière colonne de la figure 5.13). En dehors de ces exemples, les résultats sont globalement satisfaisants, et même si les poses estimées ne sont pas exactement les mêmes que les poses réelles, l'orientation est correcte et la phase de la marche est respectée (les jambes sont placées correctement l'une devant l'autre). La régression estime une pose analogue à celles des données d'apprentissage qui explique au mieux l'image courante, et le résultat ne correspond pas forcément à un modèle du corps que l'on pourrait superposer aux données image. Les résultats pourraient sans doute être améliorés avec une base d'apprentissage plus complexe et plus réaliste.

La figure 5.15 montre de quelle manière peuvent être gérés par la régression les cas de figure où la pose que l'on cherche à estimer diffère légèrement du type de poses contenues dans la base d'apprentissage. Sur cet exemple, le mouvement du bras droit sort du cadre d'une marche "régulière", et ce genre de pose n'est pas représenté dans la base. On peut constater que la régression a estimé une pose similaire à celles de la base qui explique au mieux l'exemple, sans doute en se basant sur la position des jambes. La pose des jambes est correcte mais le bras droit a été mal estimé : il a été placé le long du corps conformément aux exemples d'apprentissage.

Les courbes de la figure 5.16 présentent les angles estimés le long de la 2^e séquence pour l'orientation du torse et le balancement avant-arrière de la jambe gauche. L'évolution de l'orientation du torse sur la figure 5.16(b) est cohérente avec les données de la vidéo (marche le long d'une trajectoire circulaire toujours dans la même direction), à l'exception d'une des images (image 74) pour laquelle l'orientation est incorrecte, avec une erreur de 180°. Ce type d'erreur s'explique par le fait que pour des données 3D bruitées, il peut exister une ambiguïté sur l'orientation du corps dans l'espace (la direction de la marche) : une silhouette 3D peut facilement être confondue avec la même silhouette retournée de 180°.

Les figures 5.17 et 5.18 montrent des exemples de poses estimées le long de la trajectoire de la personne. Ici encore, sans être extrêmement précises, les poses reconstruites sont globalement satisfaisantes.

FIG. 5.12 – Résultats sur une séquence de marche.

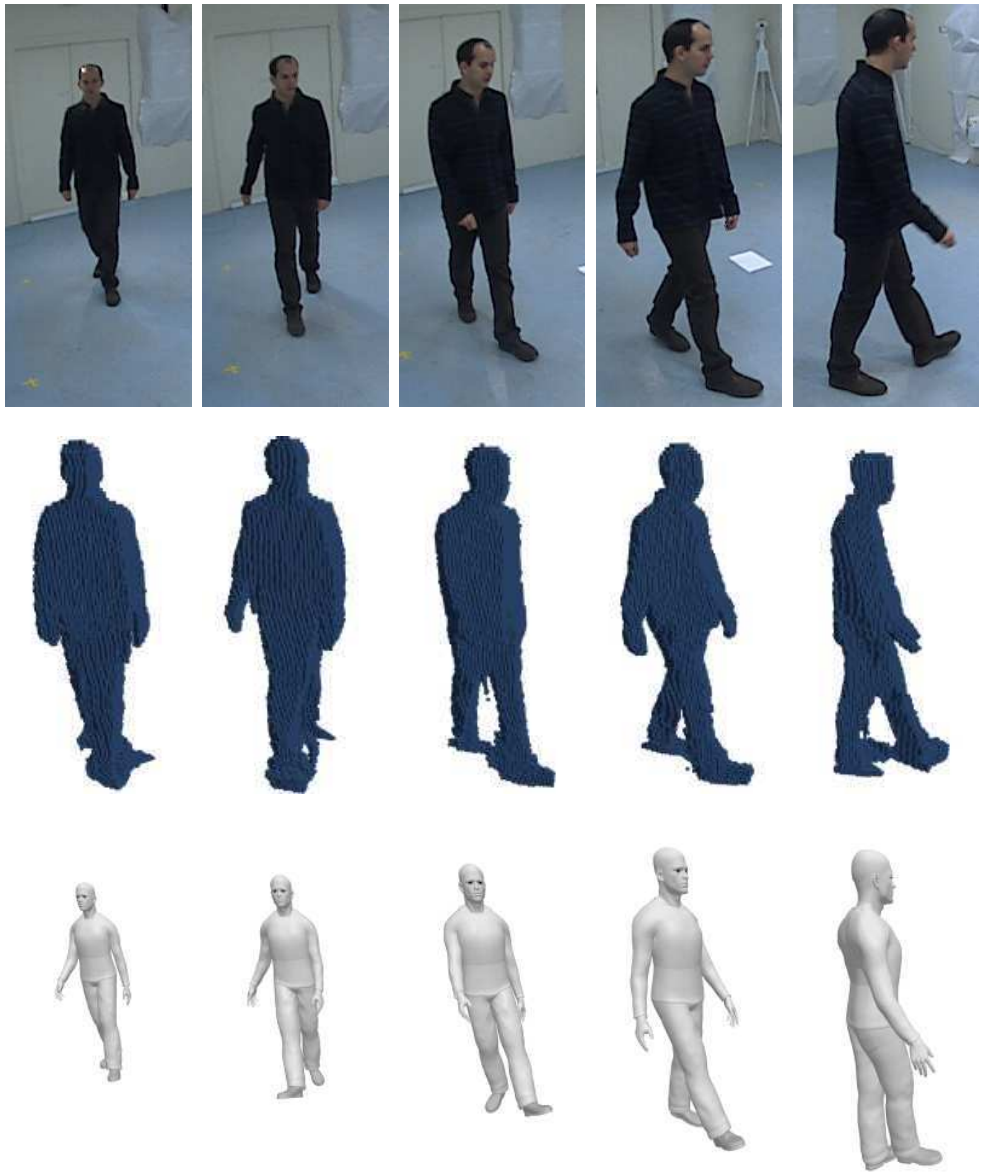


FIG. 5.13 – Résultats sur une séquence de marche.

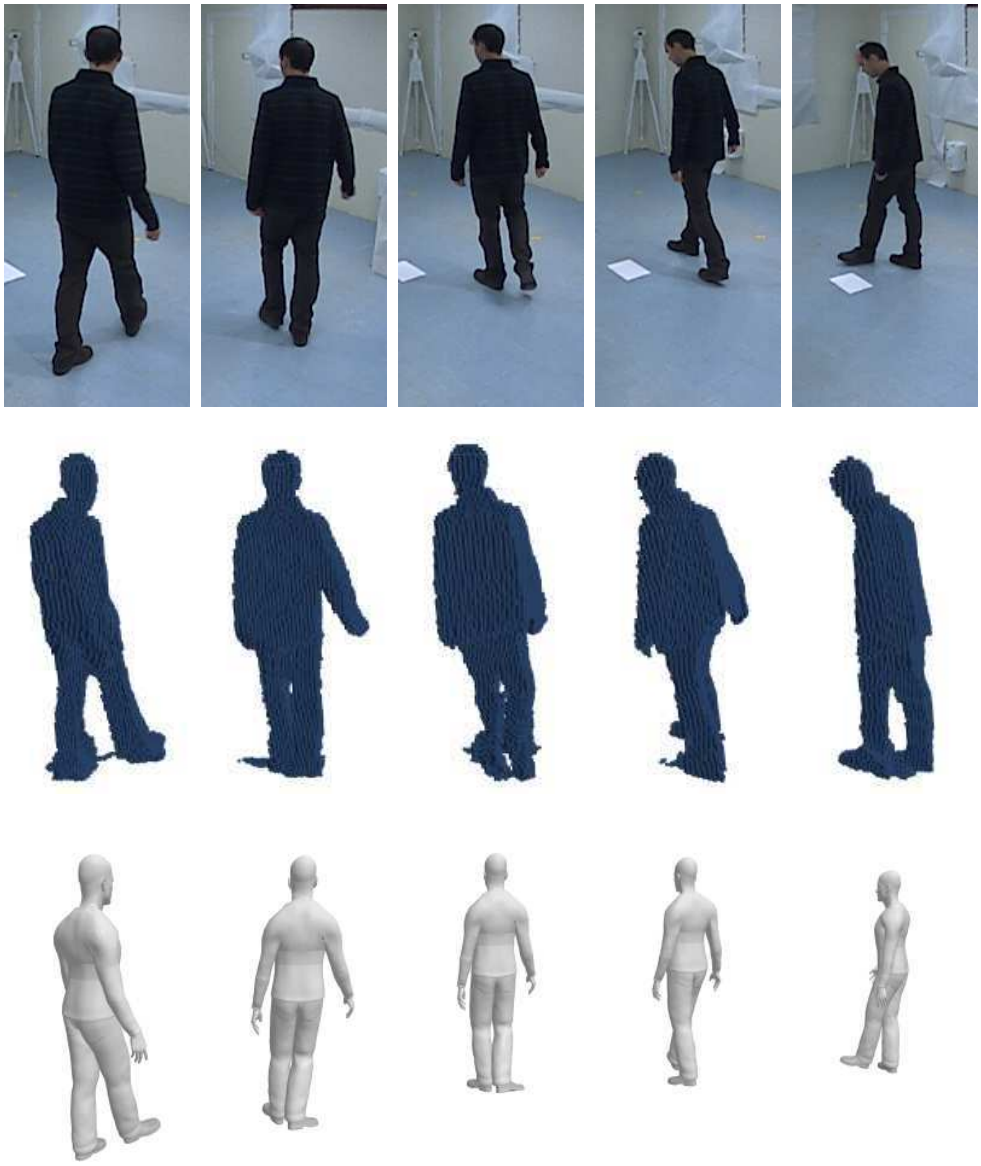
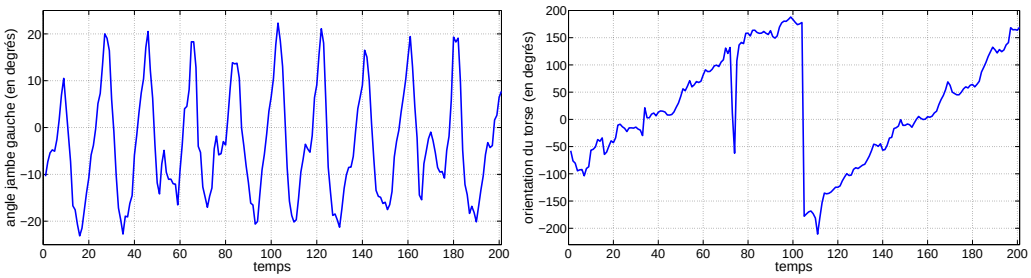


FIG. 5.14 – Résultats sur une séquence de marche.

(a) (b)

FIG. 5.15 – Exemple d'estimation sur une pose éloignée des exemples de la base.



(a) (b)

FIG. 5.16 – Angles estimés pour l'orientation du torse et la jambe gauche le long d'une séquence de marche réelle de 241 vues.
a : Angle de la jambe gauche (avant-arrière). **b :** orientation du torse dans l'espace.

5.2.2 Résultats sur des gestes

Pour ces essais, les exemples des tests réalisés en synthèse en début de chapitre et dans le chapitre précédent, avec les 8 avatars, ont été repris. Nous avons constaté expérimentalement que certains gestes fréquemment effectués dans les vidéos n'étaient pas couverts par ces exemples (le cas où les mains sont proches de la tête), cette base a donc été complétée avec de nouveaux exemples. Les nouveaux intervalles des valeurs possibles pour les différents angles des bras sont donnés dans le tableau 5.10. La base contient au total

FIG. 5.17 – Résultats sur une séquence de marche.

FIG. 5.18 – Résultats sur une séquence de marche.

articulation	axe de rotation	intervalle des valeurs possibles
épaule gauche	X (autour de l'axe du bras)	[-90 , 80]
épaule gauche	Y (avant-arrière)	[-90 , 40]
épaule gauche	Z (haut-bas)	[-80 , 30]
épaule droite	X (autour de l'axe du bras)	[-90 , 80]
épaule droite	Y (avant-arrière)	[-40 , 90]
épaule droite	Z (haut-bas)	[-30 , 80]
coude gauche	Y (plié-tendu)	[-120 , 20]
coude droit	Y (plié-tendu)	[-20 , 120]

TAB. 5.10 – Degrés de liberté et intervalles angulaires (en degrés) utilisés pour générer la base d'apprentissage.

5000 exemples d'apprentissage (les 4000 exemples de l'ancienne base auxquels ont été ajoutés 1000 nouveaux exemples). Pour la reconnaissance de gestes, la paramétrisation par les points 3D a été utilisée (les résultats du chapitre précédent ont montré que cette paramétrisation est plus précise dans le cas des gestes).

Ce type de mouvement est plus complexe à gérer qu'une marche simple, car les possibilités de gestes qui peuvent être effectués avec les bras sont beaucoup plus nombreuses. Il est aussi très difficile de construire des bases d'apprentissage suffisamment complètes pour couvrir des gestes quelconques.

Les figures 5.19, 5.21 et 5.23, montrent quelques exemples de poses estimées par régression, et les figures 5.20, 5.22 et 5.24 présentent les résultats obtenus sur ces mêmes images après raffinement. Sur chaque figure, le squelette estimé a été superposé aux images de 3 des 5 caméras (3 premières colonnes), et les deux dernières colonnes montrent l'enveloppe 3D reconstruite ainsi que le squelette estimé de deux points de vue différents.

Sur ces exemples, les poses estimées sont globalement satisfaisantes, mais certains cas problématiques ont été identifiés :

- les exemples où la personne s'écarte des postures de la base d'apprentissage, par exemple en se penchant en avant (première ligne de la figure 5.21) en tournant trop fortement les épaules (2^e ligne de la figure 5.21), ou en levant les bras au dessus de la tête (ligne de la figure 5.23),
- les cas où les bras sont tendus le long du corps (première ligne de la figure 5.19) : ce type de postures est peut être sous-représenté dans la base d'apprentissage,
- les cas où les bras sont pliés contre le torse (5^e ligne de la figure 5.19),

car l'enveloppe 3D générée est trop ambigüe pour déterminer la position des bras de manière exacte.

Pour certains de ces exemples, les résultats pourraient sans doute être améliorés en ajoutant des degrés de liberté de rotation pour les épaules ou le torse.

Effets du raffinement

Concernant le raffinement, les résultats sur données réelles confirment les conclusions des expériences chapitre précédent. Le raffinement permet d'affiner la précision lorsque l'estimation initiale est correcte, et peut dans certains cas corriger des erreurs assez grossières (3^e ligne des figures 5.23 et 5.24). Il ne permet cependant pas d'améliorer la pose reconstruite lorsque l'enveloppe 3D est trop ambigüe (5^e ligne de la figure 5.20) ou lorsque la solution initialement trouvée était trop éloignée de la pose réelle (2^e ligne de la figure 5.22). Sur ces exemples, l'estimation initiale a été dégradée par le raffinement, car le modèle s'est déplacé sur des voxels ne correspondant pas au membre du corps qu'il représente : le bras s'est recalé sur des voxels du torse dans le premier cas, et l'avant-bras sur le voxels des bras dans le second cas.

5.3 Discussion

Les tests sur des séquences réelles ont mis en évidence l'importance des postures contenues dans la base d'apprentissage sur la qualité des estimations : même s'il se généralise bien à des données non-apprises, le régresseur ne peut pas estimer correctement une pose trop éloignée de celles qui sont présentes dans la base. Sur les données de synthèse, une possibilité pour améliorer les bases est de mesurer l'erreur sur la pose reconstruite en fonction de la zone de l'intervalle angulaire dans laquelle sont situés les exemples, et de compléter les bases en ajoutant des exemples dans les zones faibles.

Pour évaluer de manière quantitative la précision de notre système sur les données réelles, il serait intéressant de construire une base de tests en localisant manuellement dans les images les positions de points clés du corps, par exemple les mains, les coudes, les épaules... et de comparer leurs positions à celles des points correspondant du squelette estimé. Une mesure de l'erreur sur la pose reconstruite pourrait être ainsi définie, soit en 2D, en projetant les points 3D estimés dans les images, soit en 3D, en remontant par triangulation

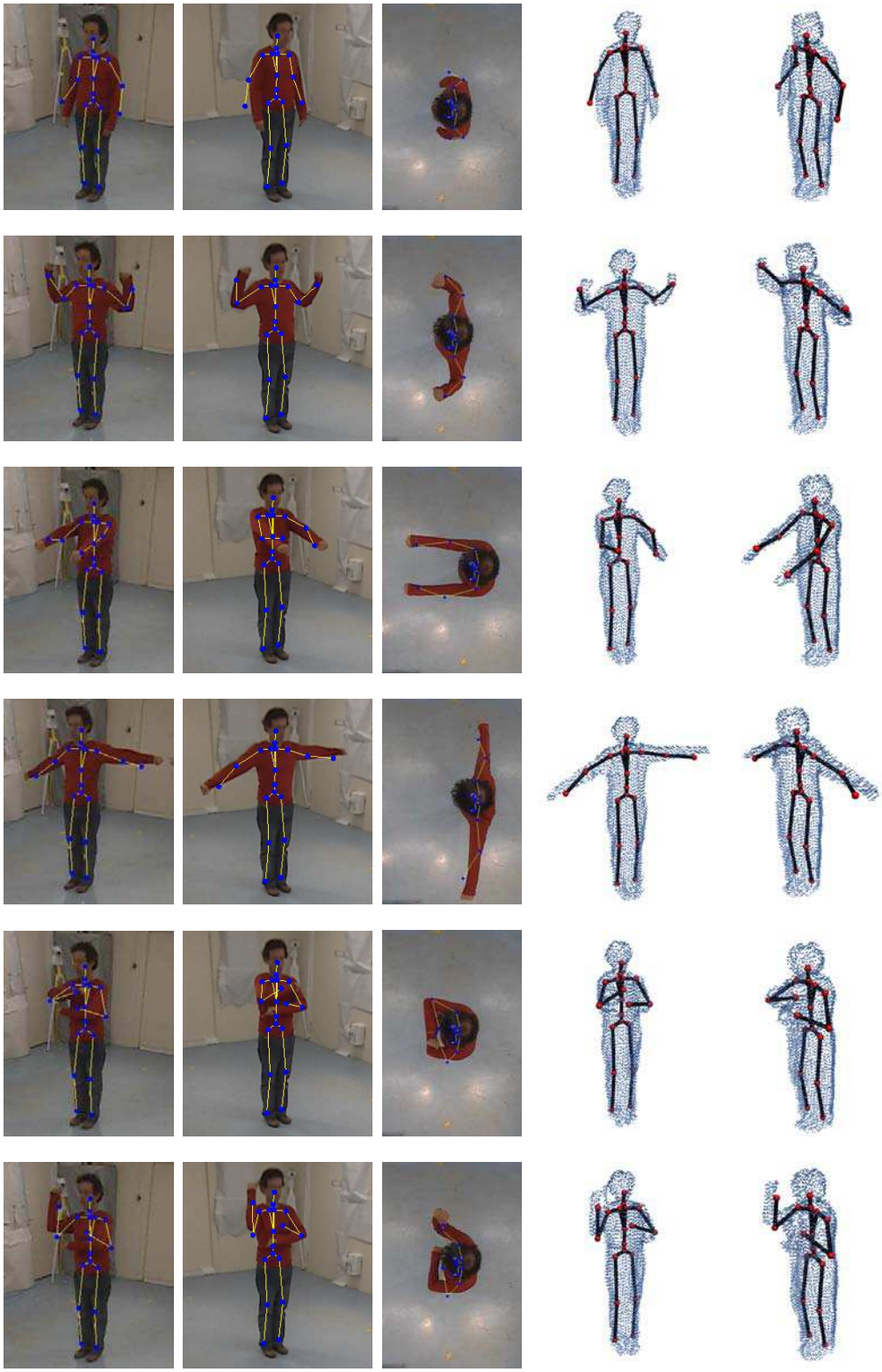


FIG. 5.19 – Exemples de poses estimées par régression.

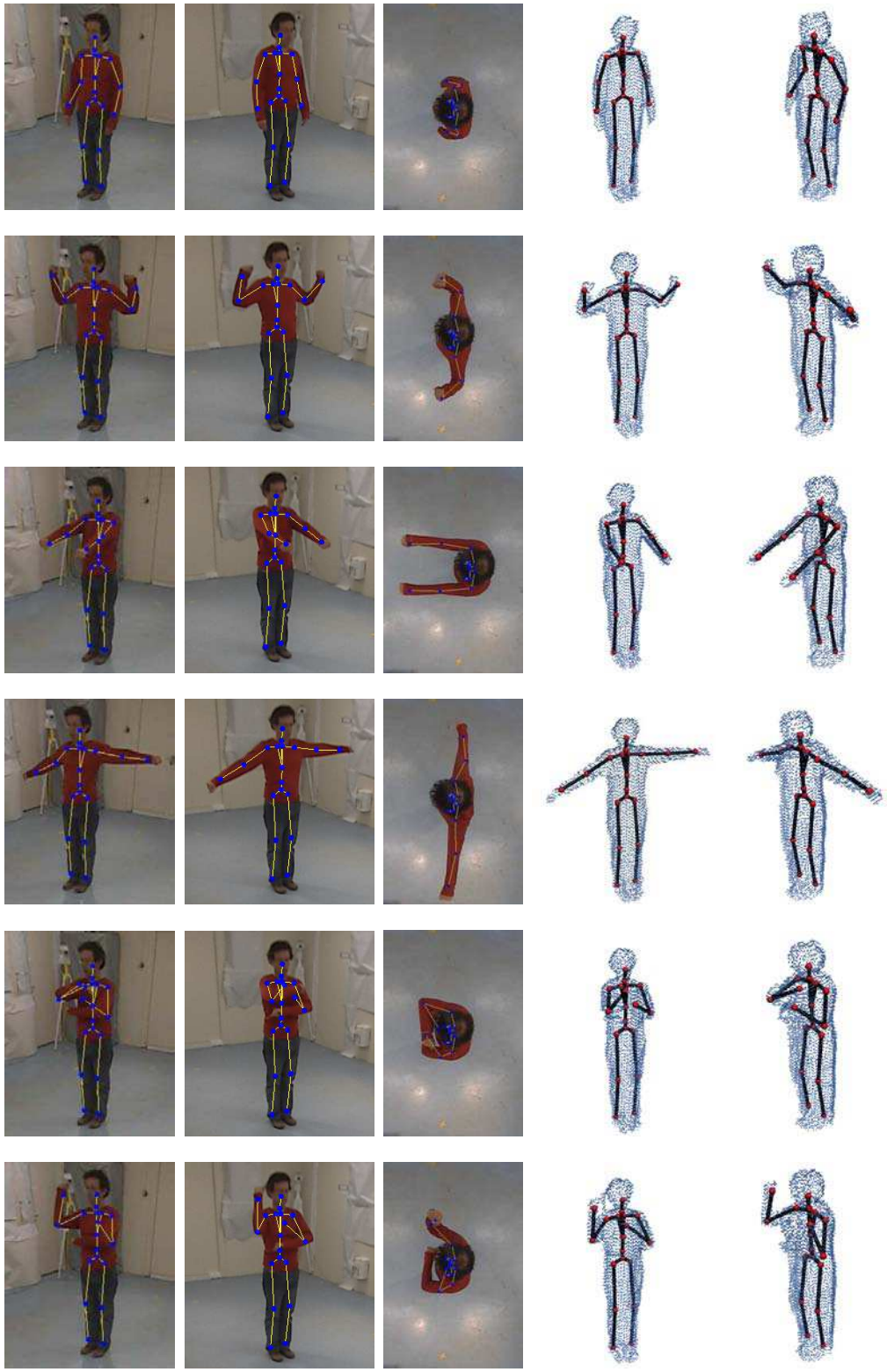


FIG. 5.20 – Poses de la figure précédente corrigées par raffinement.

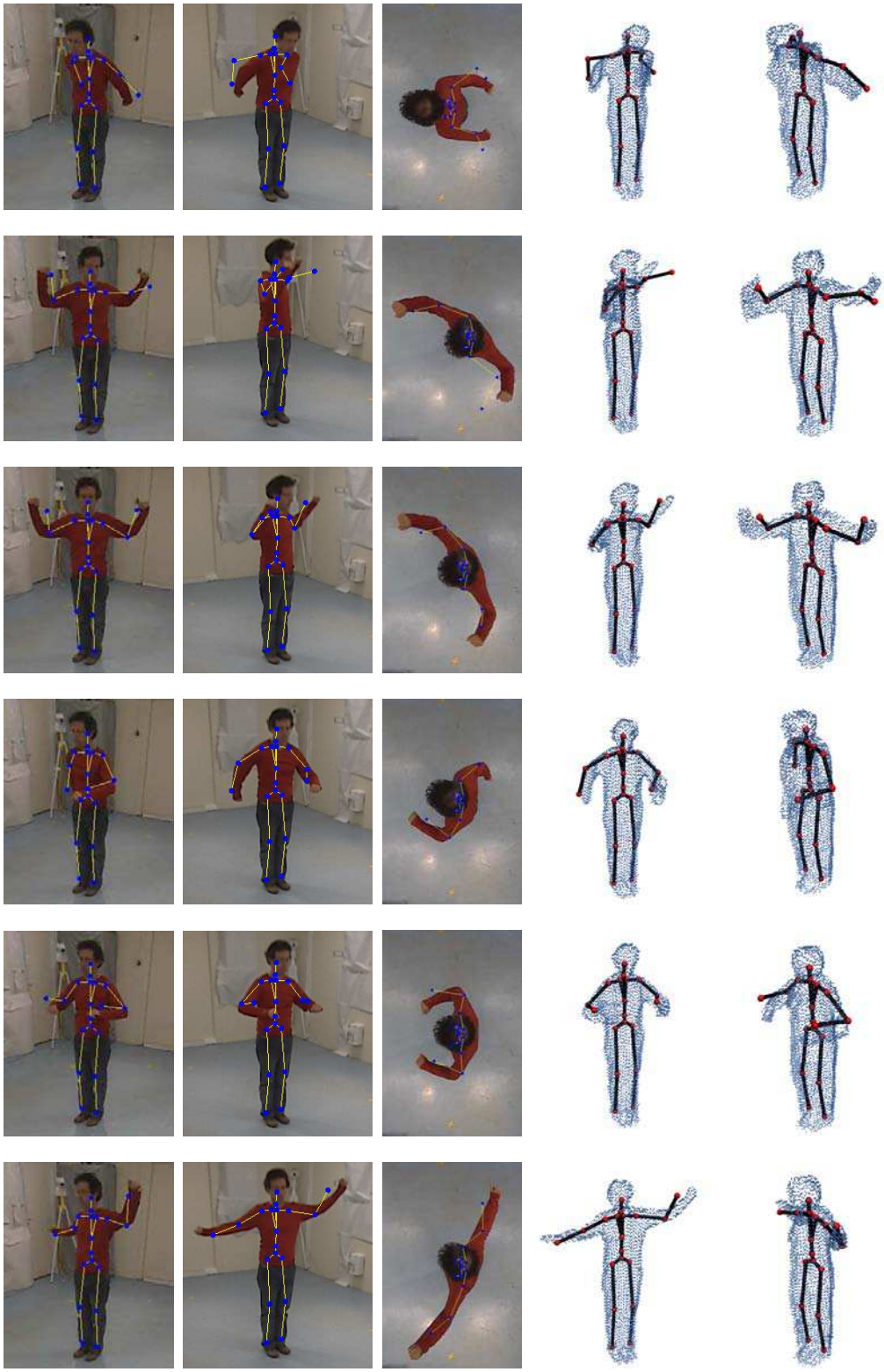


FIG. 5.21 – Exemples de poses estimées par régression.

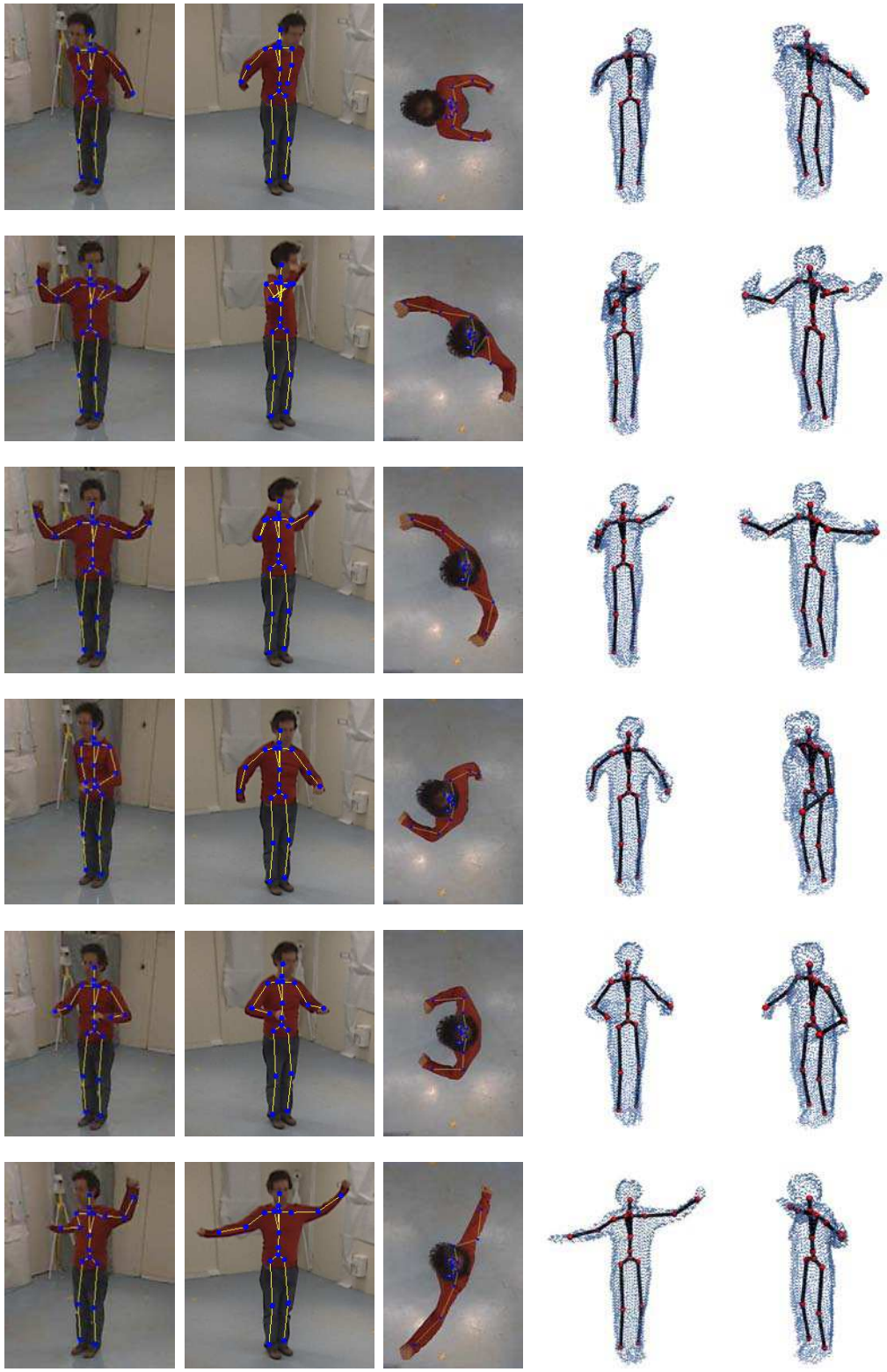


FIG. 5.22 – Poses de la figure précédente corrigées par raffinement.

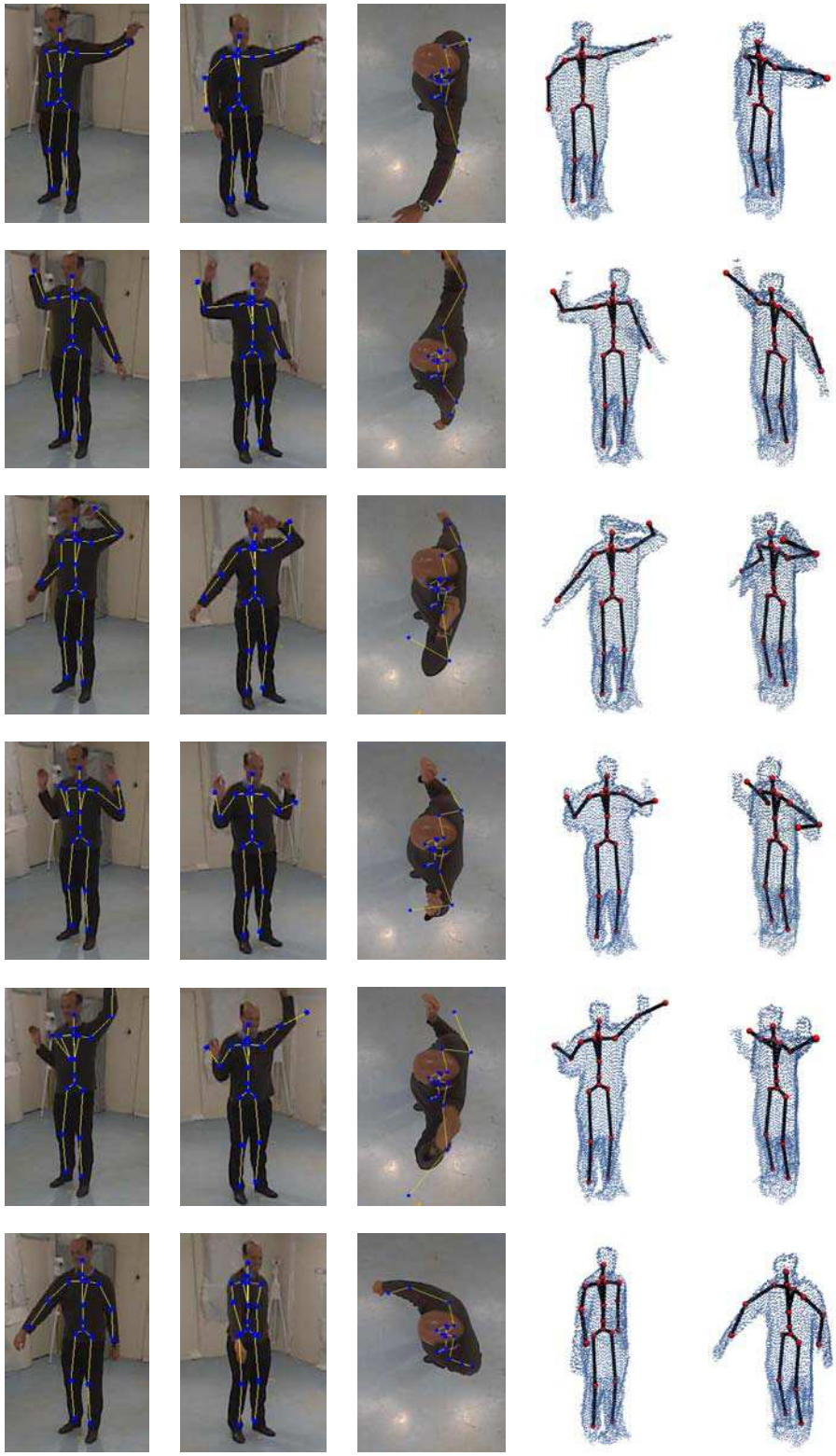


FIG. 5.23 – Exemples de poses estimées par régression.

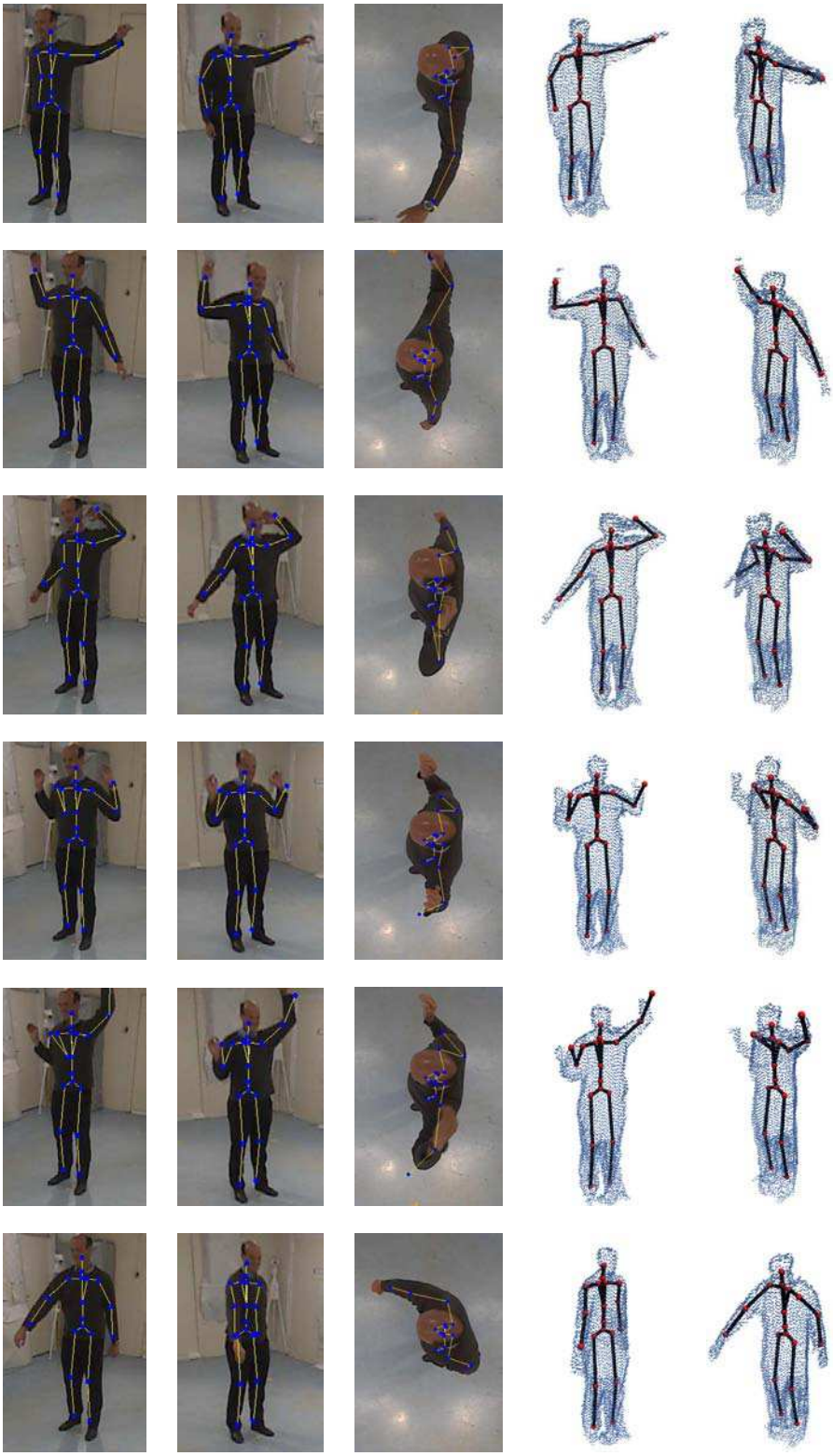


FIG. 5.24 – Poses de la figure précédente corrigées par raffinement.

à une position dans le repère du monde des points du corps identifiés dans les images.

Les expériences du paragraphe 5.1.1 ont montré en synthèse l'intérêt de la sélection des composantes du descripteur pour estimer les paramètres correspondant à une région du corps particulière. Sur les données réelles, des résultats ont été présentés sur deux groupes de mouvements, des postures de marche et des gestes. Les expériences menées en synthèse suggèrent que les différentes bases d'apprentissage pourraient être associées pour reconnaître une gamme de postures plus étendue, constituée de la combinaison de ces deux types de mouvements, en recalant le descripteur sur l'orientation du corps puis en séparant l'estimation de la pose des jambes et des bras.

Conclusion et Perspectives

Le travail présenté dans cette thèse a permis de développer un système complet d'estimation de la pose d'une personne à partir des images acquises par plusieurs caméras calibrées. Durant cette thèse ont été développés :

- un algorithme de reconstruction 3D inspiré dans la méthode proposée dans [25] ; l'enveloppe visuelle est reconstruite à partir des silhouettes 2D obtenues grâce aux outils de soustraction de fond du laboratoire (présentés en annexe A).
- un nouveau descripteur 3D permettant d'encoder de manière compacte la géométrie de l'enveloppe visuelle. La configuration de ce descripteur a été ajustée au travers de nombreux tests.
- une méthode originale d'estimation par régression en deux temps : l'orientation du corps dans l'espace est d'abord estimée, puis les angles internes sont calculés grâce à un descripteur aligné sur l'orientation du torse.
- une méthode complémentaire de raffinement de la pose d'un modèle du corps, dont la position est initialisée avec le résultat de l'estimation par régression.

De nombreuses expérimentations sur données de synthèse ont permis de régler de manière optimale les différents outils mis en oeuvre dans le système d'estimation : descripteur, paramétrisation des mouvements, régression... Des résultats encourageants ont été obtenus sur des données réelles, sur des séquences de marche d'une part, et sur des séquences de gestes réalisés par une personne statique d'autre part. Le fait d'utiliser une méthode basée sur un apprentissage laisse espérer que le système pourrait à terme fonctionner en temps-réel. Nous avons vu que les temps de calcul pourraient aussi être optimisés en adaptant l'algorithme de reconstruction 3D (notamment en utilisant une table de correspondance 3D/2D) et en choisissant la résolution des voxels judicieusement pour obtenir un bon compromis temps de calcul/précision. Par ailleurs, l'utilisation d'un modèle du corps en complément de la régression semble représenter un bon moyen d'améliorer la précision de

l'estimation, même si la méthode que nous avons proposée doit encore être complétée. Elle laisse envisager la possibilité de mettre en place un système combinant les avantages des deux types d'approches : rapidité et absence d'initialisation d'une méthode basée sur la régression, et précision d'une méthode basée sur l'ajustement d'un modèle.

Concernant les outils bas niveau impliqués dans l'estimation, un effort doit être fait au niveau de la méthode d'extraction des silhouettes. La qualité des silhouettes 2D représente en effet un élément décisif dans notre système. Les résultats ont été présentés sur des séquences réelles relativement favorables, c'est-à-dire présentant un fond simple dont la couleur se distingue clairement de celle des vêtements, mais il est probable que les estimations seraient fortement dégradées sur des données plus réalistes, plus représentatives des systèmes de vidéo surveillance. Différents moyens d'améliorer les silhouettes pourraient être envisagés, par exemple en prenant en compte des contraintes temporelles ou géométriques (i.e. entre les images des différentes caméras), ou bien des modèles sur l'apparence des silhouettes humaines. Dans cet ordre d'idées, on peut aussi envisager de coupler l'extraction des silhouettes 2D avec l'algorithme de reconstruction 3D, ou même de mélanger l'estimation de la pose et l'extraction de primitives visuelles (des méthodes basées sur cette idée d'estimation conjointe de la segmentation des images et de la pose sont par exemple proposées [21] et [42]). Par ailleurs, notre méthode s'appuie uniquement sur les silhouettes, mais nous avons vu dans l'état de l'art que de nombreuses autres primitives peuvent être mises à profit : on pourrait ainsi imaginer prendre en compte des informations sur les contours internes des silhouettes, ou sur la position des mains et du visage obtenue grâce à des algorithmes de détection.

Comme il a été dit à plusieurs reprises, un des points forts de la méthode est qu'elle fonctionne sur une seule image (plus précisément sur l'ensemble des images acquises simultanément par les caméras) et ne se sert pas comme point de départ de l'estimation de la vue précédente. Cependant, dans le cas d'une séquence vidéo, et si la fréquence d'acquisition des images le permet, il est bien évident que la prise en compte de l'information de cohérence temporelle entre des images consécutives représente une perspective intéressante pour améliorer la robustesse des estimations. Nous avons envisagé plusieurs possibilités, mais ce problème demeure ouvert. Une solution proposée par Agarwal et Triggs dans [7] est d'intégrer dans le processus de régression des données sur les poses précédemment estimées, par le biais d'un modèle de prédiction dynamique ajusté sur les données d'apprentissage. Des expériences avec une

approche similaire nous ont permis de constater que ce modèle de prédiction contraint fortement le type de mouvements qui peuvent être suivis : le mouvement (de la marche dans le cas de ces travaux) doit être effectué de la même manière et à la même vitesse que dans les données d'apprentissage. Une autre possibilité est d'intégrer l'estimation dans le modèle d'observation d'un filtre de Kalman. Mais son utilisation impose de fixer judicieusement les covariances, à la fois dans le modèle de prédiction et celui de mise à jour, ce qui suppose d'avoir une idée sur l'incertitude de la pose que nous estimons. Si un ordre de grandeur de cette incertitude peut être obtenu sur des données de synthèse, elle est extrêmement difficile à évaluer sur des données réelles puisque nous ne disposons pas de la vérité terrain.

Notre analyse pourrait sans doute bénéficier de la possibilité d'obtenir des vérités terrain sur des séquences réelles. Ces données pourraient être obtenues en utilisant un système de capture de mouvement conjointement au système d'acquisition des vidéos. Le fait de disposer des paramètres de la pose sur des données réelles pourrait par exemple permettre :

- d'avoir une mesure de l'erreur sur les données réelles. On pourrait ainsi envisager de reproduire en réel les expériences qui ont été présentées tout au long du manuscrit sur des données de synthèse, pour évaluer l'influence des différents facteurs de la méthode sur la robustesse vis-à-vis du bruit présent dans les données réelles. Ceci permettrait par exemple d'évaluer la résolution minimale de la reconstruction 3D.
- d'intégrer des exemples réels dans les bases d'apprentissage, et de rendre ainsi le système de régression plus robuste par rapport au bruit des données réelles.
- de déterminer automatiquement quelles sont les postures mal reconnues par notre système. Comme l'ont montré nos essais sur des séquences réelles, les performances du système sont fortement influencées par la qualité de la base d'apprentissage, car des poses trop éloignées de celles de la base d'entrainement sont souvent mal évaluées. Dans le cas de mouvements de marche régulière, il est relativement aisé de construire une base d'apprentissage représentative des postures qui seront observées, car le mouvement est très répétitif, et assez constant suivant les personnes. En revanche, pour reconnaître les mouvements de bras, la construction des bases est plus complexe à gérer. Il est difficile de prévoir et de bien couvrir les mouvements qui vont être réalisés par un humain. La mesure de l'erreur d'estimation sur des données réelles pourrait ainsi permettre de détecter automatiquement les faiblesses de notre système, et de compléter la base en rajoutant des exemples dans

les zones mal reconnues.

D'un point de vue applicatif, notre méthode d'estimation des paramètres de la pose peut être considérée comme un élément d'un système plus global de surveillance et d'interprétation de scènes. Nous nous sommes intéressés à l'analyse des mouvements de personnes debout, et notre analyse pourrait être complétée d'une part par une méthode de classification de postures (assis, debout, couché...), et d'autre part par une phase d'interprétation de l'évolution au cours du temps des paramètres de pose estimés par notre système. Le mouvement cyclique des angles des jambes peut par exemple permettre d'identifier des phases de marche, l'estimation de la position des mains peut servir à une détection de gestes, etc.

Annexe A

Soustraction de fond

Cette section présente la méthode utilisée pour extraire les silhouettes d'une séquence vidéo. Elle s'appuie sur la soustraction du fond par différence d'intensité entre l'image courante et l'image de fond de référence. Cette approche, fréquemment employée, est sujette à des nombreuses difficultés telles que :

- les variations colorimétriques de l'arrière-plan au cours du temps : température de couleur, intensité lumineuse, déplacement d'objets, etc. ;
- les ambiguités de couleur entre le fond et la personne/l'objet à extraire ;
- le bruit dans les images.

L'ajout de contraintes spatiales, présentées ci-après, permet de limiter considérablement ces difficultés.

A.1 Formulation du problème

On s'intéresse d'abord au terme d'attache aux données avant de détailler les contraintes spatiales. Les notations introduites ici reprennent celles de l'article [19] sur les *graph cuts*.

A.1.1 Attache aux données

Nous calculons la différence pour chaque pixel entre l'image courante I et l'image de fond I_{fond} préalablement enregistrée, et un seuillage attribue à chaque pixel sa classe, soit celle du fond, soit celle de la personne. On peut écrire le problème comme suit : soit $L(p)$, notée l_p, la fonction qui associe à chaque pixel p sa classe Fond ou sa classe Personne. On note $\delta(p) = |I_{fond}(p) - I(p)|$ la différence d'intensité entre le fond et l'image courante

pour le pixel p et S le seuil à partir du lequel on considère qu'une différence d'intensité entre le fond et l'image courante correspond à un objet distinct du fond. On veut alors estimer L pour minimiser l'erreur de classification D_p pour chaque pixel p comme suit :

$$D_p(L) = \begin{cases} \delta(p) & \text{si } l_p = Fond \text{ et } \delta(p) > S \\ \delta(p) & \text{si } l_p = Personne \text{ et } \delta(p) < S \\ 0 & \text{sinon} \end{cases} \tag{A.1}$$

Sur les séquences étudiées, le seuil S a été fixé à 30.

A.1.2 Contraintes spatiales

Pour rendre la méthode plus robuste au bruit et aux ambiguités de couleur, nous considérons la fonction de pénalité $\mathcal{V}(p, q)$ entre deux pixels voisins p et q définit comme suit :

- un premier terme qui pénalise 2 pixels voisins p et q s'ils sont de classes différentes. On s'appuie sur le modèle de Potts [89] :

$$\mathcal{V}_{pq}(l_p, l_q) = \quad (l_p \neq l_q) \tag{A.2}$$

  où $\quad(x)$ est la fonction indicatrice qui vaut 1 si x est vraie et 0 sinon.
- on ajoute un terme de gradient : on pénalise un changement de classes entre 2 pixels voisins dans les zones de faible gradient. On incite ainsi les contours de la segmentation à suivre les zones de fort gradient, en général les contours naturels de l'image [18]. La nouvelle contrainte spatiale s'écrit :

$$\mathcal{V}_{pq}(l_p, l_q) = \quad (l_p \neq l_q) \cdot \exp\left(-\frac{\|I(p) - I(q)\|^2}{2\sigma^2}\right) \tag{A.3}$$

  σ est ici l'écart type des normes des gradients de l'image. Pour éviter qu'un trop fort gradient annule complètement la contrainte de lissage, on considère finalement le critère suivant :

$$\mathcal{V}_{pq}(l_p, l_q) = \quad (l_p \neq l_q) \cdot \max\left(\gamma, \exp\left(-\frac{\|I(p) - I(q)\|^2}{2\sigma^2}\right)\right) \tag{A.4}$$

  où $\gamma \in [0, 1]$ est sélectionné par l'utilisateur (ici, $\gamma = 0.1$) .

A.1.3 Energie à optimiser

L'énergie de classification à minimiser s'écrit finalement :

$$E(L) = \underbrace{\sum_{p} D_p(L)}_{\text{difference d'intensité}} + \lambda \underbrace{\sum_{p} \sum_{q \in \mathcal{N}_p} \mathcal{V}_{pq}(l_p, l_q)}_{\text{contrainte spatiale}} \tag{A.5}$$

où $\mathcal{N}_p$ est le voisinage immédiat de p et λ le paramètre définissant le poids donné aux contraintes spatiales. On pose ici $\lambda = 30$. L'énergie E minimum correspond alors à la classification optimale pour le problème considéré.

A.2 Optimisation de cette énergie par Graph Cuts

Dans le cadre de la vision par ordinateur, les *graph cuts* [19] sont fréquemment utilisés pour résoudre de façon efficace une large variété de problèmes bas-niveau de vision par ordinateur, tels que la segmentation/classification, la mise en correspondance stéréoscopique, le *stitching* (collage d'images pour la génération de panorama), la génération de texture, ainsi que de nombreux problèmes qui peuvent être formulés sous la forme d'une minimisation d'énergie de type MRF (*Markov Random Field*). En particulier, les problèmes dits "binaires" (classification en 2 classes par exemple) peuvent se ramener au problème du flot maximal dans un graphe, qui est résolu de façon optimale via un algorithme de coupe minimale dans un graphe (Ford-Fulkerson, 1987). La configuration du flot associée à la coupe minimale est aussi la solution optimale au problème de vision considéré. Le lecteur intéressé est invité à lire l'article [83] et le chapitre *Graph Cuts in Vision and Graphics : Theories and Applications* du livre [82] qui offrent une bonne introduction aux *graph cuts*, tant d'un point de vue théorique que pour les applications potentielles.

On construit le graphe comme suit (Figure A.1) :

1. 2 noeuds spéciaux s et t, dits terminaux, qui représentent le puit et la source,
2. n noeuds pour les n pixels $p \in \mathcal{P}$,
3. $2n$ arcs orientés dits *t-links* qui représentent les termes d'attache aux données (en rouge et bleu sur la figure A.1)
4. et des arcs non-orientés dits *n-links* qui représentent les contraintes de voisinage (en jaune sur la figure).

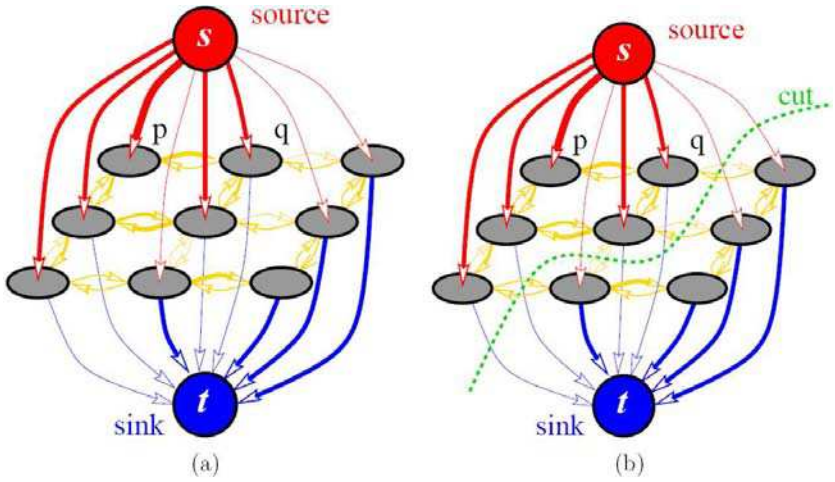


FIG. A.1 – Exemple de graphe et de coupe.
a : exemple d'un graphe avec 9 noeuds reliés au puit et à la source par des $t-links$ (en rouge et en bleu) et entre eux par des $n-links$ (en jaune). **b :** exemple d'une coupe (en vert) qui partitionne le graphe en 2 parties distinctes de sorte que chaque noeud appartienne soit au puit, soit à la source. Notons qu'ici, la coupe est schématisée par une courbe alors qu'elle est en fait une hypersurface : elle coupe aussi certains $t-links$ rouges et bleus.

Chaque pixel p est ainsi relié :

1. à la source par un *t-link* $t_{s \to p}$ de valeur $D_p(Fond)$,
2. au puit par un *t-lintk* $t_{p \to t}$ de valeur $D_p(Personne)$,
3. et aux quatre pixels voisins q_i par des *n-links* non orientés $t_{p \leftrightarrow q_i}$ de poids λ (pour des raisons de clarté, le terme de gradient est ignoré par la suite).

La coupe de ce graphe (en vert sur la figure A.1) est une séparation du graphe en deux parties disjointes S (partie contenant la source) et T (contenant le puit) assignant de ce fait chaque noeud/pixel soit à T (classe $Personne$), soit à S (classe $Fond$). Le coût de la coupe est la somme des poids des arcs coupés :

- ceux des *t-links* (un et un seul *t-link* coupé par noeud) : si le noeud considéré $p \in S$ alors nécessairement le *t-link* $t_{p \to t}$ de valeur $D_p(Personne)$ est tranché, sinon c'est le *t-link* $t_{s \to p}$ de valeur $D_p(Fond)$ qui est tranché,
- et ceux des *n-links* : ici de poids λ.

On peut vérifier que la coupe minimale du graphe décrit ci-dessus (Figure A.2) donne la solution optimale à l'énergie associée (A.5) (la figure A.2 en donne un exemple simplifié).

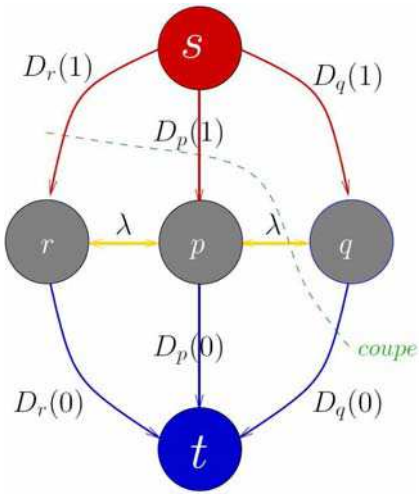


FIG. A.2 – Exemple de graphe et de coupe sur 3 pixels voisins. La classe *Fond* est notée ici 0 et la classe *Personne* est notée 1. $D_p(0)$ et $D_p(1)$ représentent resp. le coût de la classification du pixel p resp. au Fond et à Personne. Idem pour r et q. λ est le coût de la contrainte spatiale si deux pixels voisins ne sont pas de même classe (Fond ou Personne). La valeur de la coupe ici vaut $D_r(1) + D_p(1) + D_q(0) + \lambda$ correspondant à la classification $p, r \in$ Personne et $q \in$ Fond.

Annexe B

Imagerie infrarouge pour la surveillance de foules

Cette section présente notre système de surveillance des foules à partir d'une caméra infrarouge, développé dans le cadre du projet ISCAPS. Ces travaux ont été présentés au 16^e congrès francophone sur la Reconnaissance de Formes et l'Intelligence Artificielle (RFIA 2008).

Imagerie infrarouge pour la surveillance de foules

Crowd monitoring using Thermal Imaging

L. Gond Q.C. Pham J.Begard N.Allezard P.Sayd

CEA, LIST,
Laboratoire Systèmes de Vision Embarqués,
Boîte Courrier 94, Gif-sur-Yvette, F-91191 France ;

laetitia.gond@cea.fr
quoc-cuong.pham@cea.fr

Résumé

Cet article présente un système de vidéosurveillance développé dans le cadre du projet ISCAPS. L'imagerie infrarouge lointain représente un moyen robuste de gérer les changements de visibilité pouvant intervenir lors des acquisitions (luminosité, fumée), et de distinguer plus facilement des humains dans des scènes complexes. Dans cet article, nous démontrons en particulier son efficacité pour l'analyse des postures dans un groupe compact de personnes. Notre objectif est de détecter automatiquement la chute de plusieurs personnes dans une foule dense. La méthode présentée ici est basée sur la détection et la segmentation d'individus dans le groupe de personnes, grâce à l'utilisation d'une combinaison de plusieurs classifieurs faibles. L'analyse des silhouettes ainsi extraites permet de détecter les situations anormales. Notre approche a été appliquée avec succès au contexte de détection d'attaques chimiques sur un quai de gare et validée expérimentalement dans le projet. Des résultats expérimentaux sont présentés dans cet article.

Mots Clef

analyse de posture, foule, imagerie infrarouge, videosurveillance.

Abstract

This article describes a video-surveillance system developed within the ISCAPS project. Thermal imaging provides a robust solution to visibility change (illumination, smoke) and is a relevant technology for discriminating humans in complex scenes. In this article, we demonstrate its efficiency for posture analysis in dense groups of people. The objective is to automatically detect several persons lying down in a very crowded area. The presented method is based on the detection and segmentation of individuals within groups of people using a combination of several weak classifiers. The classification of extracted silhouettes enables to detect abnormal situations. This approach was successfully applied to the detection of terrorist gas attacks on railway platform and experimentally validated in the project. Some of the results are presented here.

Keywords

posture analysis, crowd, thermal imaging, video-surveillance.

1 Introduction

L'objectif général du projet ISCAPS est de proposer des réponses technologiques au risque d'attaques terroristes dans les lieux publics afin de les éviter ou de limiter leurs conséquences. Ces réponses passent par la mise en place de moyens de surveillance à la fois simples d'utilisation et efficaces, fonctionnant de façon automatique et en temps réel. Cet article décrit l'un des systèmes développés dans ce projet, répondant aux spécifications d'un opérateur de transport public (SNCF). Le scénario est le suivant : sur le quai d'une gare, le système doit détecter au plus tôt une éventuelle attaque chimique, simplement grâce à l'analyse du comportement des personnes présentes dans la scène. Il semble en effet clair que dans un pareil cas, une détection précoce peut permettre de réduire considérablement les dégâts engendrés [15]. Deux cas de figures importants se distinguent. Premièrement, à la suite d'un incendie ou d'une attaque au gaz, la zone peut se retrouver complètement enfumée, engendrant probablement un mouvement de panique collective. La plupart des personnes parviendront à s'enfuir, mais d'autres se retrouveront bloquées dans un environnement particulièrement dangereux. Deuxièmement, en l'absence de détection des gaz, un incident peut tout de même être détecté par l'observation des réactions des personnes présentes, comme le vacillement, des quintes de toux ou des chutes. Parmi les contraintes du scénario fi-

gurent la capacité du système à fonctionner malgré la fumée ou l'obscurité, et l'interprétation automatique du comportement de personnes en cas de malaise, qui représentent deux tâches difficiles. Un capteur infrarouge non-refroidi (technologie micro-bolomètre, $8 - 12\mu m$) est utilisé pour apporter de la robustesse vis-à-vis des conditions de visibilité difficiles (figure 1). Notre système doit être à la fois capable d'estimer le nombre de personnes présentes dans un lieu envahi par la fumée, mais aussi de détecter d'éventuelles chutes. Jusqu'à présent, en raison de leur coût et de leur faible durée de vie, l'utilisation de capteurs infrarouges restait confinée au domaine militaire, pour des applications comme la détection ou le suivi de véhicules. Avec l'apparition de la nouvelle génération de capteurs IR non-refroidis, moins coûteux et plus robustes, un nouveau champ d'application s'est ouvert. Leurs bonnes performances dans des conditions de visibilité difficiles et leur faculté à détecter aisément des personnes (grâce à l'émission IR naturelle des êtres vivants), en font un outil prometteur pour des applications comme la surveillance de sites [4] ou l'assistance à la conduite [14, 17]. Dans le cas d'ISCAPS, c'est la robustesse de ces capteurs vis-à-vis de la présence de fumée qui nous a conduit à choisir cette technologie. Les images infrarouges sont toutefois monochromatiques, et leur texture reste relativement pauvre si on les compare à celle des images du spectre visible.

FIG. 1 – Influence de la fumée dans des images couleur et infrarouges. Première ligne : pas de fumée, seconde ligne : zone envahie par la fumée

Dans notre cas, l'analyse de la scène est rendue complexe d'une part par la densité de la foule faisant face à la caméra et d'autre part par la présence possible de bagages sur le quai de la gare.
Indépendamment de la technologie d'imagerie utilisée, la plupart des techniques de vidéosurveillance se sont concentrées sur l'analyse de scènes dans lesquelles les personnes ne se chevauchent pas ou peu dans les images. La complexité et la variabilité des scènes de foule (nombre de personnes, occultations, postures) requièrent l'utilisation d'outils bien spécifiques. Concernant la détection,[19] propose une méthode capable de gérer le cas de personnes partiellement occultées, mais ne considère pas réellement le problème des groupes très denses. Dans [6], une méthode basée sur le flot optique et l'extraction de contours permet d'estimer la densité et le mouvement d'une foule. Cette approche n'est toutefois pas fiable dans le cas de foules denses. Les auteurs de [12] parviennent à effectuer le suivi global d'un groupe et à évaluer sa densité. Dans [13], la densité d'une foule est estimée grâce au calcul de la dimension fractale des contours. [1] présente un détecteur de situations d'urgence dans des foules, s'appuyant sur des statistiques du flot optique extraites de données vidéo. Toutefois, aucune de ces approches n'a été conçue pour détecter le comportement anormal de certains individus dans la foule. Dans notre application, cette analyse représente pourtant une étape essentielle. Certaines approches ont abordé ce sujet. Dans [20], un algorithme de segmentation bayésien a été proposé pour dénombrer les personnes dans la foule grâce à des modèles de forme. Mais cette méthode est trop lente dans le cas d'une grande foule. [16] présente un algorithme de détection dans une foule basé sur l'analyse spatio-temporelle d'une séquence vidéo. Une segmentation des régions en mouvement est combinée avec une classification des piétons, de la foule et des véhicules. Cette approche est intéressante pour compter les personnes plus que pour extraire certains individus et elle ne permet pas d'analyser des personnes statiques. Notre approche consiste à traiter les images infrarouges dans le but d'extraire des individus d'un groupe dense et propose une solution innovante pour détecter des personnes à terre.

2 Résumé de la méthode proposée

La détection des personnes à terre est rendue difficile par la grande variabilité de leurs apparences (allongées, roulées en boule...). Notre méthode consiste à extraire les objets d'intérêt de la scène. Parmi ces objets, les personnes debouts sont segmentées et supprimées afin de permettre une meilleure analyse des objets restant et d'en extraire les personnes couchées ou agenouillées.

Comme notre caméra IR est fixe, nous avons choisi d'effectuer une modélisation du fond pour extraire les objets d'intérêt dans un module de prétraitement. Le fond est donc appris grâce à une méthode statistique adaptative. En raison de la grande variabilité des vêtements et des postures, et de la densité potentiellement forte des personnes présentes dans l'image, la forme et l'apparence de la tête nous semble être une caractéristique visuelle plus stable que l'individu tout entier. Dans la deuxième phase de notre algorithme, les hypothèses sur la présence d'individus sont donc générées par un détecteur de têtes, qui combine trois techniques complémentaires i) une détection des pics sur l'ensemble des blobs obtenus après soustraction de fond, ii) une détection des formes elliptiques dans l'image infrarouge, et iii) une détection de la forme représentée par l'ensemble tête-épaules. Ces hypothèses de détection sont ensuite classées en deux groupes, en fonction de leur position dans la scène par rapport à un seuil sur la hauteur : les

têtes situées au dessus du seuil permettent d'initialiser un modèle de personne debout. Les paramètres de ce modèle sont ajustés lors d'une étape de segmentation, grâce à une méthode de Monte Carlo par chaînes de Markov (MCMC). Ce raffinement permet d'obtenir une meilleure localisation des personnes debout, pour améliorer par la suite l'analyse des composantes restantes. Les blobs restants sont examinés et classés comme étant une personne couchée ou un autre objet. Les têtes détectées situées en dessous du seuil de hauteur constituent quant à elles des hypothèses de personnes couchées. Ces données sont enfin fournies en entrée d'un module de détection de menace, qui estime le risque de manière probabiliste, et déclenche différents niveaux d'alarme. Une vue d'ensemble de notre algorithme est présentée sur la figure 2.

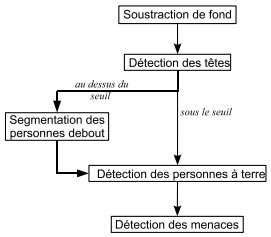


FIG. 2 – Les différentes étapes de l'algorithme.

3 Segmentation des humains dans des images infrarouges

3.1 Modélisation du fond

Dans les images thermiques, les personnes se distinguent plus facilement du fond que dans les images couleur. Mais cette technologie nécessite tout de même des outils d'analyse spécifiques. Par exemple, les intensités peuvent varier d'un individu à l'autre ou dépendre fortement des conditions extérieures [7]. D'autres effets, comme les changements de polarité thermique des objets ou l'inhomogénéité des corps humains, représentent des difficultés supplémentaires dans la segmentation des personnes. De plus, même si elles sont moins sensibles aux conditions d'illumination que les images couleurs, les émissions infrarouges restent dépendantes de certains facteurs comme la lumière du soleil sur les objets.

L'approche SKDA (Sequentiel Kernel Density Approximation) a déjà prouvé son efficacité pour la modélisation du fond [9] ou pour des algorithmes de suivi basés sur l'apparence [10]. Cette méthode tire sa robustesse de sa capacité à encoder plusieurs modes, et à s'adapter à des variations lentes au cours du temps, en intégrant de nouveaux échantillons et en délaissant les plus anciens. L'algorithme SKDA permet en outre d'obtenir une représentation compacte de l'information, puisque les modes proches les uns des autres peuvent être fusionnés grâce à une procédure de mean-shift, avec une complexité temporelle linéaire [10].

Dans le cas de notre capteur IR, il se peut qu'un léger déplacement des intensités du fond se produise, en raison par exemple de la présence de sources chaudes dans le champ de vision de la caméra. Cet effet indésirable peut conduire à de médiocres performances de l'algorithme de soustraction de fond. Pour surmonter cette difficulté, nous proposons une soustraction de fond en deux temps : après une première soustraction de fond, nous calculons l'offset entre la moyenne des modes du modèle SKDA, et la moyenne des pixels classés comme appartenant au fond, et nous appliquons cet offset lors d'une seconde soustraction de fond. Dans de nombreux cas difficiles, l'image après soustraction s'en trouve nettement améliorée.

3.2 Configuration du système d'acquisition

Le scénario qui nous intéresse ici se déroule sur le quai d'une gare où des personnes attendent un train. Pour le type de capteur utilisé, le choix sur les objectifs disponibles reste encore restreint et nous ne disposons que d'une focale de $25\,mm$. Par conséquent, pour couvrir une vaste étendue et prendre en compte les différentes contraintes du site, le capteur est placé à $10\,m$ en face du quai. Dans cette configuration, la profondeur du quai peut être considérée comme faible devant sa largeur. Nous l'approximons donc par un plan vertical dans l'espace 3-D, délimité verticalement par une ligne située à hauteur du sol ($z = 0$), et une autre située à $z = 2\,m$ (voir figure 3). Une seconde approximation nous permet de définir une correspondance directe entre les coordonnées dans ce plan 3-D et la région d'intérêt 2-D correspondante dans l'image : pour une position horizontale donnée dans l'image, la distance en pixels entre les deux lignes correspond à une hauteur réelle de $2\,m$.

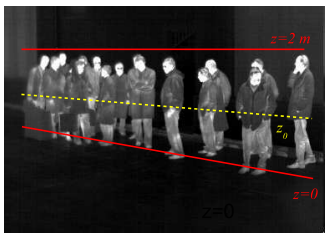


FIG. 3 – Le quai est modélisé par un plan 3-D du point de vue du capteur. La région d'intérêt est située entre les deux lignes $z = 0$ et $z = 2\,m$. La ligne en pointillés représente le seuil de hauteur z_0 utilisé pour détecter les personnes à terre.

3.3 Modélisation du corps humain

Pour représenter les personnes debout, nous utilisons un modèle géométrique 2-D. La tête est modélisée par une ellipse, et le torse et les jambes par deux rectangles verticaux (voir figure 4). Si un tel modèle peut paraître simpliste, il

présente l'avantage de nous éviter des processus complexes de projection et d'évaluations dans l'image. Des rectangles 2-D nous permettent par exemple d'utiliser des images intégrales [18]. Rappelons que notre objectif ici est d'avoir une bonne approximation de l'espace occupé par une personne debout, pour la distinguer des autres composantes présentes dans l'image (comme des personnes couchées au sol), et non de segmenter précisément toutes les parties du corps pour retrouver son attitude exacte, ce qui n'est pas l'objet de cet article.

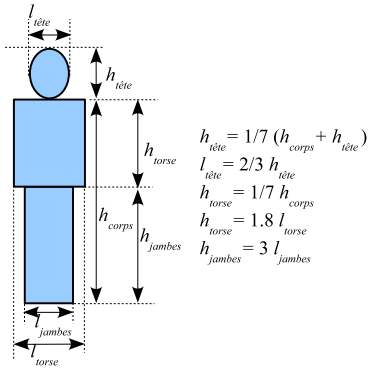


FIG. 4 – modèle 2-D d'être humain composé d'une ellipse pour la tête et de deux rectangles pour le torse et les jambes.

3.4 Détection de la tête

A l'exemple de [20], des hypothèses sur les positions des têtes dans l'image sont générées par deux méthodes. La première est une détection des pics dans l'image obtenue après soustraction de fond. Un pic correspond à un maximum local vertical sur l'ensemble des blobs extraits. Nous définissons une région d'intérêt par une procédure de calibrage manuelle, qui nous donne une estimation de la taille d'une personne en fonction de sa position horizontale $\mathbf{x}$ dans l'image. Ce calibrage permet aussi de déterminer la taille de la zone de recherche des maxima locaux. La seconde méthode est basée sur une détection des têtes dans l'image infrarouge grâce à un modèle elliptique, comme décrit dans [3]. L'idée est que dans des images thermiques, les gradients les plus hauts sont observés autour des parties du corps les plus exposées comme le visage. Pour chaque position $\mathbf{x}$ du centre du modèle elliptique Γ, un score de concordance est calculé :

$$S_{\Gamma(\mathbf{x})} = \frac{1}{N_{\mathbf{x}_i}} \sum_{\mathbf{x}_i \in \Gamma(\mathbf{x})} \nabla I(\mathbf{x}_i) \cdot \mathbf{n}(\mathbf{x}_i) \qquad (1)$$

où les $\mathbf{x}_i$ sont des points distribués sur l'ellipse, ∇I le gradient de l'image, et $\mathbf{n}(\mathbf{x}_i)$ la normale à $\Gamma(\mathbf{x})$ au point $\mathbf{x}_i$. Les têtes sont recherchées uniquement à l'intérieur de la zone d'intérêt définie dans 3.2, et l'échelle du modèle elliptique est adaptée selon la position horizontale dans l'image.

Les hypothèses sur les positions des têtes sont en outre validées en calculant l'intersection entre le modèle 2-D de forme humaine correspondant à ces positions et la carte F obtenue par soustraction de fond. Pour accélérer le calcul de cette intersection, nous utilisons l'image intégrale de F pour les parties rectangulaires du modèle (le torse et les jambes). Toutes ces détections sont enfin fusionnées grâce à un algorithme de clustering séquentiel.

3.5 Détection de l'ensemble tête-épaules grâce à une cascade de classifieurs

Les gradients significatifs le long des contours de la silhouette des individus, et en particulier la forme de l'ensemble tête-épaules, représentent des informations pertinentes pour la détection de personnes. En complément de la méthode précédente, nous utilisons donc le résultat d'un détecteur "tête-épaules" basé sur des descripteurs locaux combinés dans une cascade de classifieurs [18]. En raison de la grande variabilité des apparences et des postures humaines, un descripteur robuste est nécessaire pour représenter les caractéristiques pertinentes. Nous avons utilisé des histogrammes de gradient (comme [5] et [2]) constitués de n cellules d'orientation et d'une cellule additionnelle représentant la quantité d'information contenue dans le support de l'histogramme. Après des normalisations en luminance, ces histogrammes sont calculés sur une grille dense (en position et en échelle), pour capturer de la manière la plus fine possible les caractéristiques de la forme "tête et épaules" que nous souhaitons reconnaître. Nos paramètres par défaut donnent des histogrammes de taille 900. L'utilisation d'une image intégrale permet de faciliter le calcul des valeurs du gradient et des votes. Nous avons observé que notre descripteur est plus performant avec 9 subdivisions sur l'orientation non-signée (tous les 20˚entre 0˚et 180˚). Nous obtenons ainsi des vecteurs à 9000 composantes pour chaque forme représentée. L'utilisation d'orientations non-signées implique qu'il n'y a pas de distinction entre les différences zone claire/zone sombre et zone sombre/zone claire (ce qui est raisonnable si on considère la variabilité des apparences humaines : cheveux, peau, vêtements...). Pour réduire l'aliasing, nous effectuons un lissage des composantes de l'histogramme en attribuant une fraction x du vote à la cellule correspondante et une fraction $1 - x$ à la cellule la plus proche, où $x \in [x_{min}, 1]$. x_{min} dépend de l'angle seuil α_T au dessus duquel on considère qu'un vote se fait uniquement sur une cellule.

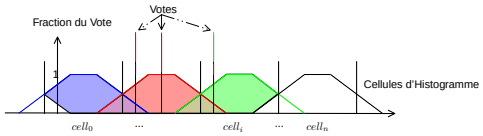


FIG. 5 – histogramme des votes *lissés*

L'apprentissage est réalisé par une cascade de classifieurs

[8, 18], où les classifieurs faibles sont de simples arbres de décision à un niveau (decision stump) sur les cellules de l'histogramme. L'objectif de cette approche et de réduire au maximum le nombre de zones d'intérêt candidates au fur à mesure de la cascade, de manière que la première couche de la cascade élimine la majorité des zones d'intérêt et que la dernière couche n'ait que quelques régions à évaluer. La figure 6 présente une vue d'ensemble de cette méthode. L'évolution de l'erreur et du taux de détection au cours des différentes étapes de la cascade permet au détecteur d'obtenir au final de bons résultats. Une cascade à 10 étages permet en effet de parvenir à un taux de détection de 0.9 lorsque chaque étage possède un taux de 0.99 (puisque $0.99^{10} \approx 0.904$). De la même façon, un taux de faux positifs de 10^{-4} est quasiment atteint avec 10 étages ayant un taux de faux positifs de 40% ($0.40^{10} \approx 1.0 \times 10^{-4}$). Nous avons testé différents jeux de paramètres d'initialisation conduisant à des cascades de 9 à 12 étages. Les résultats de ces détecteurs diffèrent principalement sur le nombre de faux positifs, les taux de détection étant en revanche assez similaires.

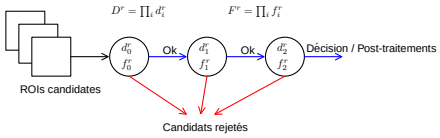


FIG. 6 – cascade de classifieurs, où D^r est le taux de détection et F^r le taux de faux positifs.

Lors de la procédure d'apprentissage, nous avons utilisé $n_+ = 3000$ exemples positifs et $n_- = 10000$ exemples négatifs *difficiles*. Les exemples difficiles sont obtenus sur une séquence à l'aide d'un classifieur simple (quelques étages), entrainé sur des exemples négatifs choisis aléatoirement. La procédure de boosting sélectionne les composantes pertinentes (correspondant à des classifieurs faibles) pour construire un classifieur fort, lui-même utilisé pour choisir $n-$ exemples négatifs pour le prochain étage de la cascade. La dernière couche entrainée est évaluée sur un ensemble de validation, et si le résultat n'est pas satisfaisant, une nouvelle couche est ajoutée à la cascade. Pour améliorer la procédure et la rendre plus robuste, une fois que les classifieurs faibles ont été sélectionnés pour l'étage courant de l'algorithme, une autre boucle permet d'ajuster les poids des différents classifieurs.

3.6 Raffinement de la segmentation par un échantillonnage MCMC

Une fois que les têtes ont été localisées précisément par notre détecteur, nous procédons à une étape de segmentation dont le but est d'ajuster la position et la forme du modèle de corps humain pour les personnes debout. Cette procédure permet par la suite une meilleure analyse des blobs restants. Le problème de segmentation peut être formulé sous la forme d'une estimation d'un maximum a posteriori (MAP) :

$$\Theta^* = \arg\max_{\Theta} p(\Theta|F) \tag{2}$$

où $\Theta = \{\theta_i\}$ est l'ensemble des paramètres des différents modèles humains, et F est la carte obtenue après soustraction du fond. D'après la règle de Bayes, la probabilité a posteriori peut être décomposée en un terme de vraisemblance et une probabilité a priori :

$$p(\Theta|F) \propto p(F|\Theta)\, p(\Theta) \tag{3}$$

Les paramètres de chaque individu i sont $\theta_i = \{\Delta x_i, h_i, f_i\}$, où Δx est la translation horizontale du corps par rapport à sa position initiale, h sa hauteur et f sa corpulence (rapport de la largeur sur la hauteur).
Comme les paramètres des modèles des différents individus sont indépendants les uns des autres, on peut supposer que la probabilité a priori jointe est le produit des probabilités a priori pour chaque individu :

$$p(\Theta) = \prod_{i=1}^{N} p(\theta_i) \tag{4}$$

où N est le nombre de personnes debout détectées.
Pour un individu i, la probabilité a priori est :

$$p(\theta_i) = p(\Delta x_i)\, p(h_i)\, p(f_i) \tag{5}$$

où $p(\Delta x_i)$ est une distribution gaussienne $\mathcal{N}(0, \sigma_{\Delta x})$ tronquée sur l'intervalle $[-0.4, 0.4]$, $p(h_i)$ est une distribution uniforme sur l'intervalle $[h_{i0} - 0.3, h_{i0} + 0.3]$ (où h_{i0} est la taille initiale du modèle), et $p(f_i)$ est une distribution uniforme sur l'intervalle $[0.9, 2.2]$.
Comme plusieurs personnes peuvent s'occulter les unes les autres, la vraisemblance jointe ne peut pas s'exprimer comme le produit des vraisemblances de chaque hypothèse d'individu. Nous utilisons donc une vraisemblance basée sur le nombre de pixels "mal classés", c'est-à-dire N_{01}, le nombre de pixels qui appartiennent à la carte F obtenue après soustraction du fond mais qui ne sont dans aucun modèle humain, et N_{10}, le nombre de pixels qui sont dans un modèle de personne mais qui ne sont pas dans F :

$$p(F|\Theta) = \sigma(\lambda_{01} \frac{\Delta N_{01}}{N}) \cdot \sigma(\lambda_{10} \frac{\Delta N_{10}}{N}) \tag{6}$$

où $\sigma(x) = 1/(1 + e^{-x})$ est la fonction sigmoïde, ΔN_{01} (resp. ΔN_{10}) est la différence entre la valeur courante et la valeur initiale de N_{01} (resp. N_{10}), et λ_{01} et λ_{10} sont deux coefficients de pondération dépendant de la taille des êtres humains dans l'image.
Pour maximiser une telle fonction, les méthodes d'échantillonnage nous fournissent un moyen simple d'explorer l'espace des états possibles et d'évaluer la solution optimale avec une grande robustesse vis-à-vis des maxima locaux. Les paramètres optimaux Θ^* sont calculés par une approche Monte Carlo par chaînes de Markov (MCMC) à l'exemple de [20] pour la segmentation de personnes, et de

[11] dans le contexte du suivi multi-cibles. L'algorithme de Metropolis-Hastings est une technique efficace pour échantillonner une distribution quelconque, en construisant séquentiellement une chaine de Markov qui converge vers cette distribution. Nous avons utilisé comme distribution instrumentale une distribution gaussienne. Les principales étapes de l'algorithme sont les suivantes :

– Initialiser les modèles de personnes d'après les résultats de la détection de tête, avec les paramètres définis en 3.3 et une échelle définie par la position horizontale x dans l'image
– Pour chaque échantillon :
  1. choisir un individu i au hasard ,
  2. à partir de l'état courant θ_i^t, prédire une nouvel état θ_i^{t+1} avec la distribution instrumentale
  3. estimer la nouvelle probabilité a posteriori $p^{t+1}(\Theta|F)$,
  4. calculer le taux d'acceptation $r = \frac{p^{t+1}(\Theta|F)}{p^t(\Theta|F)}$
  5. si $r > 1$ le nouvel état θ^{t+1} est accepté, sinon il est accepté avec la probabilité r

Une fois que les échantillons de la distribution postérieure sont générés, une estimation de l'état est obtenue en calculant la moyenne pondérée des paramètres des différents échantillons.

4 Détection d'une menace

Pour chaque image de la vidéo, notre système de détection des menaces peut produire quatre sorties possibles : scène *vide* lorsqu'aucune personne n'a été détectée, *normal* si des individus debout ont été détectés et personne n'est à terre, *avertissement* ou *alarme*, selon le niveau de confiance, dans le cas où des personnes allongées par terre ont été détectées. La décision est prise en calculant une probabilité de menace associée à l'événement *personne à terre*, $p_t(LD)$ et en la comparant à deux seuils, un seuil d'avertissement τ_W et un seuil d'alarme τ_A tels que $0 < \tau_W < \tau_A \leq 1$.

4.1 Détection des personnes allongées par terre

Pour déterminer si des personnes sont à terre, notre algorithme se base sur :

– la hauteur des têtes détectées : en dessous de $z_0 = 1\,m$, la personne est classée comme couchée (voir 6 pour les personnes de petite taille),
– l'analyse des blobs restants, une fois les personnes debout supprimées.

Les personnes debout qui ont été segmentées sont donc supprimées de la carte obtenue après soustraction de fond. Des opérations de morphologie mathématique sont ensuite appliquées à cette image binaire pour éliminer les régions les plus fines. Dans les images infrarouges, les corps humains ont généralement une texture plus hétérogène que les objets inertes, comme les valises par exemple. Les blobs restants sont finalement analysés et classés comme des personnes allongées ou d'autres objets, en utilisant un critère sur la distance au sol et la texture, caractérisée par la variance locale calculée sur un voisinage de 3x3 pixels.

4.2 Probabilité d'une menace

On note N_l le nombre d'hypothèses sur les personnes à terre à un instant t donné. La probabilité de menace associée à l'événement *personne à terre*, $p_t(LD)$ peut s'exprimer comme le produit de trois probabilités. Le premier terme $p_t^{N_l}(LD)$ est lié au nombre d'hypothèses sur les personnes à terre détectées : plus le nombre d'hypothèses est grand, plus la probabilité de menace est élevée. La deuxième probabilité $p_t^{X_l}(LD)$ dépend des positions estimées de ces hypothèses par rapport au sol : le niveau de confiance augmente lorsque la distance moyenne au sol diminue. Le troisième terme $p_t^{f_l}(LD)$ exprime la fréquence de détection dans une fenêtre temporelle. Un historique de la détection d'événements sur une fenêtre temporelle de largeur h_w est conservé, et on compte le nombre d'occurrences n_l de l'événement *personne à terre* détectées. La probabilité résultante peut s'écrire :

$$p_t(LD) = \underbrace{\frac{1}{1+e^{-\lambda_1 N_l}}}_{p_t^{N_l}(LD)} \cdot \underbrace{\frac{1}{1+e^{-\frac{\lambda_2}{N_l}\sum_{i=1}^{N_l}\frac{z_i}{z_0}}}}_{p_t^{X_l}(LD)} \cdot \underbrace{\frac{n_l}{h_w}}_{p_t^{f_l}(LD)} \quad (7)$$

où λ_1 et λ_2 sont deux coefficients de pondération, fixés empiriquement à $\lambda_1 = 1$ et $\lambda_2 = 2$, et où z_i représente la distance des hypothèses au seuil de hauteur z_0.

5 Résultats expérimentaux et discussion

Le détecteur de menace a été testé à de nombreuses reprises dans le cadre du projet. Nous présentons ici les résultats obtenus sur deux séquences longues représentatives (3351 et 7342 images respectivement). La dimension des images est de 384x272 pixels. Dans ces séquences, un groupe de personnes pénètrent dans la zone vide, et restent debout sur le quai. A un instant donné, certaines d'entre elles tombent par terre, d'autres restent debout. Les résultats obtenus lors des différentes étapes de l'algorithme sont illustrés dans les figures qui suivent. Dans la figure 7 sont présentés les résultats de soustraction de fond obtenus avec la méthode SKDA. Sur toutes les images traitées, aucun des individus présents n'a été supprimé lors de la soustraction de fond. En revanche, quelques fausses détections ont été observées, mais celles-ci ont pu être filtrées lors des étapes suivantes.
La figure 8 présente les résultats de la détection de têtes. Le détecteur s'est révélé très robuste étant donné la complexité des scènes à traiter en terme de densité humaine, d'occultations et de variabilité des postures. Le détecteur de formes elliptiques et la cascade de classifieurs parviennent à détecter des têtes même lorsqu'une autre personne placée en arrière plan vient altérer les contours de la tête ou lorsque la

FIG. 7 – Résultats de la soustraction de fond.

personne s'est penchée. On peut aussi souligner la complémentarité des différents détecteurs dans les cas difficiles. Des fausses détections en dessous du seuil sur la hauteur ont été observées dans seulement 14 des 10639 images.

Le nombre d'hypothèses sur les têtes situées au-dessus du seuil est une estimation du nombre de personnes debout dans la scène, la précision de l'estimation dépendant naturellement de la densité et du niveau de chevauchement. Nous avons représenté sur un graphique le nombre d'hypothèses générées pour les têtes sur une séquence de 2555 vues. Au début de cette séquence, les 15 personnes présentes sont debout. Après environ 1150 vues, 9 personnes s'en vont et durant les 600 autres vues, 6 individus restent debout sur le quai (voir figure 9). Les trois principales phases de la séquence sont visibles sur le graphique. Comme on pouvait s'y attendre, l'erreur d'estimation augmente avec la densité des personnes présentes, mais les résultats restent cohérents avec la vérité terrain.

La figure 10 présente le résultat de la segmentation de personnes debout avec l'approche bayésienne et le modèle 2-D du corps humain. Les paramètres optimaux obtenus après l'échantillonnage MCMC permettent au modèle de mieux s'ajuster à l'apparence des individus. En particulier, la hauteur et la corpulence de la personne restent correctement estimées et l'inclinaison du corps d'une personne peut être compensée dans l'image par une translation du corps par rapport à la tête.

Une fois que les corps des personnes debout ont été supprimés, les blobs restants et proches du sol sont classés comme étant des personnes debout ou d'autres objets, comme indiqué sur la figure 11. Les cas de non détection sont dus à la forte densité locale et à un chevauchement extrême.

La table 1 donne les résultats de la détection des menaces : les différentes vues sont classées comme vide, normale ou avertissement/alarme. L'algorithme donne des résultats satisfaisants puisque l'estimation est cohérente avec les nombres de vues de la vérité terrain.

FIG. 8 – Résultats de la détection de têtes. Les croix indiquent les têtes détectées par la détection des pics et le détecteur de formes elliptiques (en jaune au dessus du seuil sur la hauteur, et en magenta en dessous), les boîtes représentent les résulats de détection de la cascade de classifieurs. La dernière image montre un exemple de fausse détection sur les jambes, qui présentent un fort gradient.

D'un point de vue temporel, si nous représentons les niveaux d'alarme au cours du temps (figure 12), on observe une bonne concordance entre la sortie de l'algorithme et la vérité terrain sur les séquences prétraitées. On peut noter un léger décalage de quelques images entre le début de la menace dans la vérité terrain et l'activation de l'alarme par notre système. Ceci est dû à la largeur de la fenêtre temporelle utilisée pour calculer la probabilité de fréquence. Pour cette application, un retard de quelques secondes pour le déclenchement de l'alarme est tout à fait acceptable surtout s'il permet de réduire les risques de fausses alarmes. Il reste néanmoins certains aspects à améliorer. Au début de la séquence 2, de fausses alarmes apparaissent brièvement. De plus, l'alarme n'est pas activée de façon continue du-

Seq 1	*Vide*	*Normal*	*Avertissements/Alarmes*
Estimation	0	2214	1137
Vérité terrain	0	2161	1190
Seq 2	*Vide*	*Normal*	*Avertissements/Alarmes*
Estimation	0	4784	2557
Vérité terrain	0	4394	2948

TAB. 1 – Résultats de la détection de menaces (en nombre de vues).

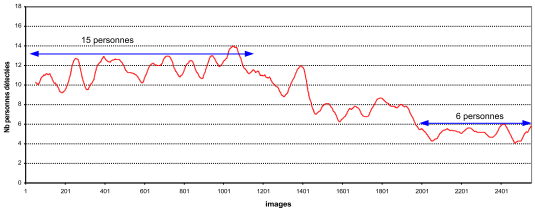


FIG. 9 – Estimation du nombre de personnes debout. Au début de la séquence, 15 personnes sont debout. 9 quittent la zone d'intérêt, et 6 restent.

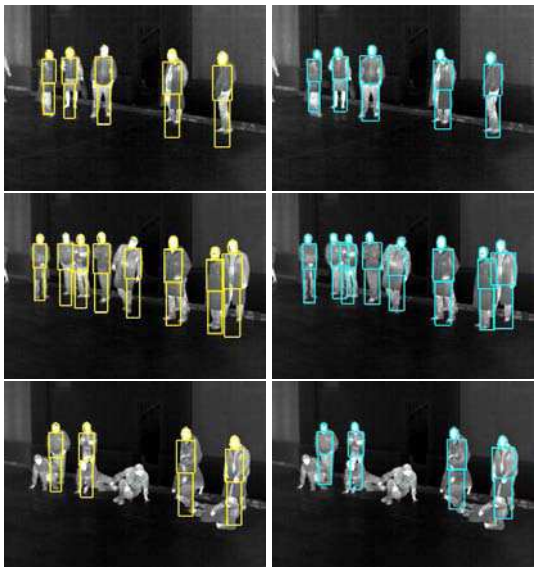


FIG. 10 – Résultats de la segmentation des personnes debout. Colonne de gauche : position initiale, colonne de droite : segmentation finale après échantillonage MCMC.

rant la période critique à cause des cas intermittents où les personnes à terre ne sont pas détectées. Le lissage temporel de la détection de menace pourrait être amélioré grâce à un filtrage à plus long terme.
En termes de rapidité, avec un code C++ pouvant être encore serieusement optimisé, notre algorithme traite approximativement 2-3 images par secondes sur un PC conventionnel Pentium IV 3Ghz, 1.5Gb RAM. Cette fréquence de traitement est largement suffisante pour l'application visée.

6 Conclusion et perspectives

Dans cet article, nous avons démontré les capacités de notre système à analyser des scénarios complexes de détection de menaces dans des images infrarouges, comme la détection des personnes tombées à terre dans une foule. Les

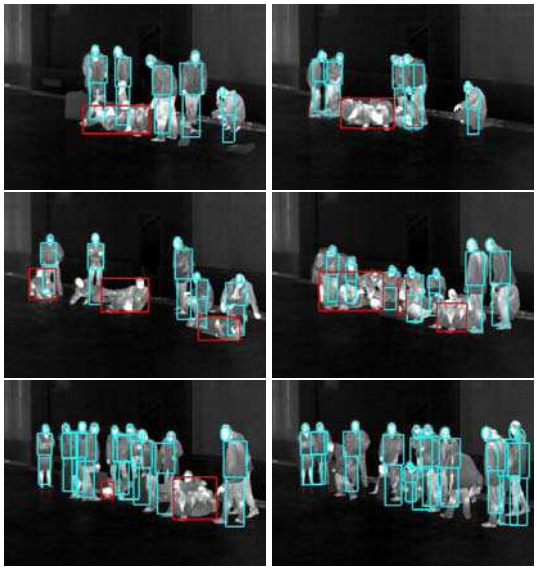


FIG. 11 – Détections des chutes. La dernière image illustre un cas de non détection.

résultats expérimentaux montrent la robustesse de la méthode, puisque le taux de fausses alarmes est bas, et peu d'alarmes attendues ont été manquées. Les performances de la détection pourraient être améliorées en intégrant un lissage temporel, à la fois dans le processus de segmentation (suivi visuel) et dans la sortie du module de détection de menace (long terme). Une autre amélioration possible serait l'enrichissement du modèle de silhouette pour augmenter la précision de la segmentation à un faible coût calculatoire. Un meilleur modèle permettrait de distinguer les personnes de petites tailles des personnes agenouillées. Un modèle multi-couches des groupes de personnes pourrait également permettre de mieux gérer les occultations. De plus, les résultats obtenus dans cette étude dépendent largement de la position du capteur infrarouge et de son champ de vision. Idéalement, une vue de dessus et un champ de vision plus large diniueraient le chevauchement entre les individus.

Références

[1] Ernesto L. Andrade, Scott Blunsden, and Robert B. Fisher. Hidden markov models for optical flow analysis in crowds. In *Proc. IEEE Int. Conf. on Pattern Recognition*, volume 1, pages 460–463, Los Alamitos, CA, USA, 2006.

[2] J Begard, N Allezard, and P Sayd. Real-time humans detection in urban scenes. In *BMVC*, 2007.

[3] Stan Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *Proc. IEEE*

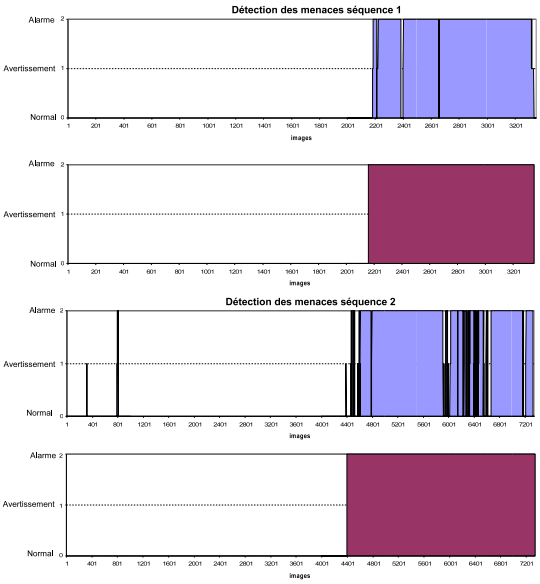


FIG. 12 – Résultats de la détection de menaces. Première ligne : sortie de l'algorithme pour la séquence 1, deuxième ligne : vérité terrain pour la séquence 1, troisième ligne : sortie de l'algorithme pour la séquence 2, dernière ligne : vértié terrain pour la séquence 2.

Conf. on Computer Vision and Pattern Recognition, pages 232–237, 1998.

[4] C. O Conaire, E. Cooke, N. O'Connor, N. Murphy, and A. Smearson. Background modelling in infrared and visible spectrum video for people tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition Workshop*, page 20, Washington, DC, USA, 2005.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, volume II, pages 886–893, 2005.

[6] A.C. Davies, Jia Hong Yin, and S.A. Velastin. Crowd monitoring using image processing. *Electronics and Communications Engineering Journal*, 7(1) :37–47, 1995.

[7] James W. Davis and Vinay Sharma. Robust background-subtraction for person detection in thermal imagery. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition Workshop*, volume 8, page 128, Washington, DC, USA, 2004.

[8] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression : a statistical view of boosting. Technical report, Dept. of Statistics, Stanford University, August 1998.

[9] Bohyung Han, Dorin Comaniciu, and Larry Davis. Sequential kernel density approximation through mode propagation : applications to background modeling. In *Proc. of the 2004 Asian Conference on Computer Vision*, 2004.

[10] Bohyung Han and Larry Davis. On-line density-based appearance modeling for object tracking. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 1492–1499, Washington, DC, USA, 2005.

[11] Zia Khan, Tucker Balch, and Frank Dellaert. Mcmc-based particle filtering for tracking a variable number of interacting targets. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(11) :1805–1918, 2005.

[12] P. Kilambi, O. Masoud, and N. Papanikolopoulos. Crowd analysis at mass transit sites. In *ITSC '06, Intelligent Transportation Systems Conference*, pages 753 – 758, Toronto, Canada, September 2006.

[13] A. Marana, S. Velastin, L. Costa, and R. Lotufo. Automatic estimation of crowd occupancy using texture and nn classification. *Safety Science*, 28(3) :165–175, 1998.

[14] Davis L. Nanda H. Probabilistic template based pedestrian detection in infrared videos. In *IEEE Intelligent Vehicle Symposium*, pages 18–20, June 2002.

[15] A.J. Policastro and S.P Gordon. The use of technology in preparing subway systems for chemical/biological terrorism. In *Commuter Rail/Rapid Transit Conference Proceedings*, 1999.

[16] P. Reisman, O. Mano, S. Avidan, and A. Shashua. Crowd detection in video sequences. In *IEEE Intelligent Vehicles Symposium*, pages 66–71, June 2004.

[17] Gandhi T. and Trivedi M.M. Pedestrian collision avoidance systems : A survey of computer vision based recent studies. In *ITSC '06, Intelligent Transportation Systems Conference*, pages 976–981, Toronto, Canada, September 2006.

[18] Paul Viola and Michael Jones. Robust real-time object detection. In *International Workshop on Statistical and Computational Theories of Vision Modeling, Learning, Computing and Sampling*, July 2001.

[19] B. Wu and R. Neviata. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *ICCV*, pages 90–97, 2005.

[20] Tao Zhao and Ramakant Nevatia. Bayesian human segmentation in crowded situations. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 459–466, June 2003.

Bibliographie

[1] http ://mocap.cs.cmu.edu/.

[2] http ://svr-www.eng.cam.ac.uk/ at315/mvrvm.htm.

[3] http ://www.mocapdata.com/.

[4] www.ict.usc.edu/graphics/animweb/humanoid.

[5] A. Agarwal et B. Triggs. Monocular human motion capture with a mixture of regressors. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2005.

[6] A. Agarwal et B. Triggs. A local basis representation for estimating human pose from cluttered images. In *Proceedings of the Asian Conference on Computer Vision*, 2006.

[7] A. Agarwal et B. Triggs. Recovering 3d human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1) :44–58, January 2006.

[8] M. Ankerst, G. Kastenmüller, H.-P. Kriegel, et T. Seidl. 3d shape histograms for similarity search and classification in spatial databases. In *Proceeding of the 6th International Symposium on Advances in Spatial Databases*, 1999.

[9] J. Bandouch, F. Engstler, et M. Beetz. Accurate human motion capture using an ergonomics-based anthropometric human model. In *Proceedings of the International Conference on Articulated Motion and Deformable Objects*, 2008.

[10] C. Barron et I.A. Kakadiaris. Estimating anthropometry and pose from a single uncalibrated image. *Computer Vision and Image Understanding*, 81(3) :269–284, 2001.

[11] S. Belongie, J. Malik, et J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24 :509–522, 2002.

[12] C. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[13] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[14] A Bottino et A Laurentini. A silhouette based technique for the reconstruction of human movement. *Computer Vision and Image Understanding*, 83 :79–95, 2001.

[15] B. Boulay, F. Bremond, et M. Thonnat. Human posture recognition in video sequence. In *Proceedings of the Workshop on Video Surveillance and Performance Evaluation of Tracking and Surveillance, at the International Conference on Computer Vision*, 2003.

[16] B. Boulay, F. Brémond, et M. Thonnat. Applying 3d human model in a posture recognition system. *Pattern Recognition Letters*, 27(15) :1788–1796, 2006.

[17] R. Bowden, T. Mitchell, et M. Sahardi. Non-linear statistical models for the 3d reconstruction of human pose and motion from monocular image sequences. *Image and Vision Computing*, 18 (9) :729–737, 2000.

[18] Y. Boykov et M.-P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *Proceedings of the International Conference on Computer Vision*, pages 105–112, 2001.

[19] Y. Boykov, O. Veksler, et R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11) :1222–1239, 2001.

[20] M. Brand. Shadow puppetry. In *Proceedings of the International Conference on Computer Vision*, pages 1237–1244, 1999.

[21] M. Bray, P. Kohli, et P. Torr. Posecut : Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In *Proceedings of the European Conference on Computer Vision*, 2006.

[22] C. Bregler et J. Malik. Tracking people with twists and exponential maps. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 8–15, 1998.

[23] N. Canterakis. Fast 3d zernike moments and- invariants. Technical report, Institute of Informatics, University of Freiburg, Germany, 1997.

[24] T.J. Cham et J.M. Rehg. A multiple hypothesis approach to figure tracking. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages II : 239–245, 1999.

[25] K. M. Cheung, T. Kanade, J.-Y. Bouguet, et M. Holler. A real time system for robust 3d voxel reconstruction of human motions. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2000.

[26] I. Cohen et H. Li. Inference of human postures by classification of 3d human body shape. In *Proceedings of the IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, 2003.

[27] R. Collobert et S. Bengio. Svmtorch : Support vector machines for large-scale regression problems. *Journal of Machine Learning Research*, 1 :143–160, 2001.

[28] R. Cucchiara, C. Grana, A. Prati, et R. Vezzani. Probabilistic posture classification for human-behavior analysis. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, 35(1) :42–54, 2005.

[29] T. E. de Campos et D. W. Murray. Regression-based hand pose estimation from multiple cameras. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2006.

[30] Q. Delamarre et O. Faugeras. 3d articulated models and multiview tracking with physical forces. *Computer Vision and Image Understanding*, 81(3) :328–357, 2001.

[31] J. Deutscher, A. Blake, et I. Reid. Articulated body motion capture by annealed particle filtering. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2000.

[32] A. Elgammal et C.S. Lee. Inferring 3d body pose from silhouettes using activity manifold learning. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2004.

[33] P. Felzenszwalb et D. Huttenlocher. Efficient matching of pictorial structures. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2000.

[34] M. Fischler et R. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, 22 :67–92, 1973.

[35] J.-S. Franco et E. Boyer. Fusion of multi-view silhouette cues using a space occupancy grid. In *Proceedings of the International Conference on Computer Vision*, 2005.

[36] D. M. Gavrila. The visual analysis of human movement : A survey. *Computer Vision and Image Understanding*, 73 :82–98, 1999.

[37] D. M. Gavrila et L. S. Davis. 3-d model-based tracking of human upper body movement : a multi-view approach. In *Proceedings of the International Symposium on Computer Vision*, 1995.

[38] D. M. Gavrila et L. S. Davis. 3d model-based tracking of humans in action : a multi-view approach. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 1996.

[39] L. Gond, P. Sayd, T. Chateau, et M. Dhome. A 3d shape descriptor for human pose recovery. In *Proceedings of the International Conference on Articulated Motion and Deformable Objects*, 2008.

[40] L. Gond, P. Sayd, T. Chateau, et M. Dhome. A regression-based approach to recover human pose from voxel data. In *Proceedings of the 2nd IEEE International Workshop on Tracking Humans for the Evaluation of their Motion in Image Sequences, at the International Conference on Computer Vision*, 2009.

[41] K. Grauman, G. Shakhnarovich, et T. Darrell. A bayesian approach to image-based visual hull reconstruction. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2003.

[42] K. Grauman, G. Shakhnarovich, et T. Darrell. Inferring 3d structure with a statistical image-based shape model. In *Proceedings of the International Conference on Computer Vision*, 2003.

[43] R. Gross et J. Shi. The cmu motion of body (mobo) database. Technical report, Robotics Institute, Carnegie Mellon University, 2001.

[44] L. Guan, J.-S. Franco, et M. Pollefeys. 3d occlusion inference from silhouette cues. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2007.

[45] F. Guo et G. Qian. Dance posture recognition using wide-baseline orthogonal stereo cameras. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2006.

[46] F. Guo et G. Qian. Learning and inference of 3d human poses from gaussian mixture modeled silhouettes. In *Proceedings of the International Conference on Pattern Recognition*, 2006.

[47] I. Haritaoglu, D. Harwood, et L. S. Davis. W4 : Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 :809–830, 2000.

[48] I. Haritaoglu, D. Harwood, et L.S. Davis. Ghost : a human body part labeling system using silhouettes. In *Proceedings of the International Conference on Pattern Recognition*, 1998.

[49] I. Haritaoglu, D. Harwood, et L.S. Davis. W4s :a real time system for detecting and tracking people in $2\frac{1}{2}$d. In *Proceedings of the European Conference on Computer Vision*, 1998.

[50] L. Herda, R. Urtasun, et P.Fua. Hierarchical implicit surface joint limits for human body tracking. *Computer Vision and Image Understanding*, 99(2) :189–209, 2005.

[51] N. Howe, M. Leventon, et W. Freeman. Bayesian reconstruction of 3d human motion from single-camera video. In *Neural Information Processing Systems*, 2000.

[52] S. Ioffe et D.A Forsyth. Probabilistic methods for finding people. *International Journal of Computer Vision*, 43 (1) :45–68, 2001.

[53] M. Isard et A. Blake. Condensation - conditional density propagation forvisual tracking. *International Journal of Computer Vision*, 29(1) :5–28, 1998.

[54] L. B. Jack et A. K. Nandi. Fault detection using support vector machines and artificial neural networks, augmented by genetic algorithms. *Mechanical systems and signal processing*, 16(2-3) :373–390, 2002.

[55] S.X. Ju, M.J. Black, et Y. Yacoob. Cardboard people : a parameterized model of articulated image motion. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, pages 38–44, 1996.

[56] I.A. Kakadiaris et D. Metaxas. Three-dimensional human body model acquisition from multiple views. *International Journal of Computer Vision*, 30(3) :192–218, 1998.

[57] M. Körtgen, G.J. Parl, M. Novotni, et R. Klein. 3d shape matching with 3d shape contexts. In *Proceedings of the 7th Central European Seminar on Computer Graphics*, 2003.

[58] M.P. Kumar, P.H.S. Torr, et A. Zisserman. Learning layered motion segmentations of video. In *Proceedings of the International Conference on Computer Vision*, 2005.

[59] X. Lan et D.P. Huttenlocher. Beyond trees : common-factor models for 2d human pose recovery. In *Proceedings of the International Conference on Computer Vision*, 2005.

[60] A. Lanza, L. Di Stefano, J. Berclaz, F. Fleuret, et P. Fua. Robust multi-view change detection. In *Proceedings of the British Machine Vision Conference*, 2007.

[61] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2) :150–162, 1994.

[62] M. W. Lee et I. Cohen. Human upper body pose estimation in static images. In *Proceedings of the European Conference on Computer Vision*, 2004.

[63] M.W. Lee et I. Cohen. A model-based approach for estimating human 3d poses in static images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(6) :905–916, 2006.

[64] W. Lee, W. Woo, et E. Boyer. Identifying foreground from multiple images. In *Proceedings of the Asian Conference on Computer Vision*, 2007.

[65] F. Lerasle, G. Rives, et M. Dhome. Tracking of human limbs by multiocular vision. *Computer Vision and Image Understanding*, 75(3) :229–246, 1999.

[66] C.-H. Lo et H.-S. Don. 3-d moment forms : their construction and application to object identification and positioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(10) :1053–1064, 1989.

[67] J. MacCormick et M. Isard. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *Proceedings of the European Conference on Computer Vision*, 2000.

[68] W. Matusik, C. Buehler, et L. McMillan. Polyhedral visual hulls for real-time rendering. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, 2001.

[69] W. Matusik, C. Buehler, R. Raskar, S. J. Gortler, et L. McMillan. Image-based visual hulls. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques*, 2000.

[70] A. S. Micilotta, E. J. Ong, et R. Bowden. Detection and tracking of humans by probabilistic body part assembly. In *Proceedings of the British Machine Vision Conference*, 2005.

[71] I. Mikic, M. Trivedi, E. Hunter, et P. Cosman. Articulated body posture estimation from multi-camera voxel data. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2001.

[72] C. Ménier, E. Boyer, et B. Raffin. 3d skeleton-based body pose recovery. In *3rd International Symposium on 3D Data Processing, Visualization and Transmission*, 2006.

[73] T. Moeslund et E. Granum. A comprehensive survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3) :231–268, 2001.

[74] T. Moeslund, A. Hilton, et V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 103(2-3) :90–126, 2006.

[75] G. Mori et J. Malik. Recovering 3d human body configurations using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28 (7) :1052–1062, 2006.

[76] G. Mori, X. Ren, A. Efros, et J. Malik. Recovering human body configurations : Combining segmentation and recognition. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2004.

[77] D.D. Morris et J.M. Rehg. Singularity analysis for articulated object tracking. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 289–296, 1998.

[78] E. J. Ong et S. Gong. A dynamic human model using hybrid 2d-3d representations in hierarchical pca space. In *Proceedings of the British Machine Vision Conference*, 1999.

[79] E.-J. Ong, A. Micilotta, R. Bowden, et A. Hilton. Viewpoint invariant exemplar-based 3d human tracking. *Computer Vision and Image Understanding*, 104(2) :178–189, 2006.

[80] E.J. Ong et S. Gong. The dynamics of linear combinations : tracking 3d skeletons of human subjects. *Image and Vision Computing*, 20 :397 – 414, 2002.

[81] L. Panini et R. Cucchiara. A machine learning approach for human posture detection in domotics application. In *Proceedings of the 12th International Conference on Image Analysis and Processing*, 2003.

[82] N. Paragios, Y. Chen, et O. Faugeras. *Mathematical Models in Computer Vision : The Handbook*. Springler, 2005.

[83] M. Pechaud. Introduction aux graphcuts en vision par ordinateur. *ENS Paris*, 2007.

[84] Q.-C. Pham, Y. Dhome, L. Gond, et P. Sayd. Video monitoring of vulnerable people in home environment. In *Proceedings of the 6th International Conference On Smart homes and health Telematics, Ames, IOWA, June 28-July 2*, 2008.

[85] Q.-C. Pham, L. Gond, J. Begard, N. Allezard, et P. Sayd. Real-time posture analysis in a crowd using thermal imaging. In *Proceedings of the 7th IEEE International Workshop on Visual Surveillance, at the International Conference on Computer Vision and Pattern Recognition*, 2007.

[86] R. Poppe. Vision-based human motion analysis : An overview. *Computer Vision and Image Understanding*, 108(1-2) :4–18, 2007.

[87] R.W. Poppe. Evaluating example-based pose estimation : Experiments on the humaneva sets. In *Proceedings of the Workshop on Evaluation of Articulated Human Motion and Pose Estimation (EHuM), at the International Conference on Computer Vision and Pattern Recognition*, 2007.

[88] R.W. Poppe et M. Poel. Comparison of silhouette shape descriptors for example-based human pose recovery. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2006.

[89] R. B. Potts. Some generalized order-disorder transformations. *Proceedings of the Cambridge Philosophical Society*, 48 :106–109, 1945.

[90] D. Ramanan et D. Forsyth. Finding and tracking people from the bottom up. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2003.

[91] T. J. Roberts, Stephen J. McKenna, et Ian W. Ricketts. Human pose estimation using learnt probabilistic region similarities and partial configurations. In *Proceedings of the European Conference on Computer Vision*, 2004.

[92] R. Ronfard, C. Schmid, et B. Triggs. Learning to parse pictures of people. In *Proceedings of the European Conference on Computer Vision*, 2002.

[93] R. Rosales et S. Sclaroff. Inferring body pose without tracking body parts. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages II : 721–727, 2000.

[94] R. Rosales et S. Sclaroff. Specialized mappings and the estimation of human body pose from a single image. In *Proceedings of the Workshop on Human Motion*, pages 19–24, 2000.

[95] R. Rosales, M. Siddiqui, J. Alon, et S. Sclaroff. Estimating 3d body pose using uncalibrated cameras. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages I :821–827, 2001.

[96] R.Plänkers et P.Fua. Articulated soft objects for multiview shape and motion capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9) :1182–1887, 2003.

[97] A. Shahrokni, V. Lepetit, et P. Fua. Bundle adjustment for markerless body tracking in monocular video sequences. In *Proceedings of the ISPRS Workshop on Visualization and Animation of Reality-based 3D Models, Vulpera, Switzerland*, 2003.

[98] G Shakhnarovich, P. Viola, et T. Darrell. Fast pose estimation with parameter-sensitive hashing. In *Proceedings of the International Conference on Computer Vision*, 2003.

[99] D. G. Shen et Horace H S Ip. Discriminative wavelet shape descriptors for recognition of 2-d patterns. *Pattern Recognition*, 32(2) :151–165, 1999.

[100] L. Sigal, S. Bhatia, S. Roth, M. Black, et M. Isard. Tracking loose-limbed people. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2004.

[101] L. Sigal et M. J. Black. Humaneva : Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical report, Brown University, Department of Computer Science, 2006.

[102] L. Sigal, M. Isard, B. Sigelman, et M. Black. Attractive people : Assembling loose-limbed models using non-parametric belief propagation. *Neural Information Processing Systems*, 16 :1539–1546, 2003.

[103] C. Sminchisescu, A. Kanaujia, Z. Li, et D. Metaxas. Learning to reconstruct 3d human motion from bayesian mixtures of experts. a probabilistic discriminative approach. Technical report, CSRG-502, University of Toronto, 2004.

[104] C. Sminchisescu, A. Kanaujia, Z. Li, et D. Metaxas. Discriminative density propagation for 3d human motion estimation. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2005.

[105] C. Sminchisescu et B. Triggs. Estimating articulated human motion with covariance scaled sampling. *International Journal of Robotics Research*, 22(6) :371–391, 2003.

[106] Y. Sun, M. Bray, A. Thayananthan, B. Yuan, et P. H. S. Torr. Regression-based human motion capture from voxel data. In *Proceedings of the British Machine Vision Conference*, 2006.

[107] R. Szeliski. Rapid octree construction from image sequences. *CVGIP : Image Understanding*, 58(1) :23–32, 1993.

[108] T. Tangkuampien et D. Suter. Real-time human pose inference using kernel principal component pre-image approximations. In *Proceedings of the British Machine Vision Conference*, 2006.

[109] C. J. Taylor. Reconstruction of articulated objects from point correspondences in a single uncalibrated image. *Computer Vision and Image Understanding*, 80 :349–363, 2000.

[110] M. Teague. Image analysis via the general theory of moments. *Journal of the Optical Society of America*, 70(8) :920–930, 1980.

[111] A. Thayananthan. *Template-based Pose Estimation and Tracking of 3D Hand Motion.* PhD thèse, Department of Engineering, University of Cambridge, 2005.

[112] A. Thayananthan, R. Navaratnam, B. Stenger, P. Torr, et R. Cipolla. Multivariate relevance vector machine for tracking. In *Proceedings of the European Conference on Computer Vision*, 2006.

[113] M. Tipping. The relevance vector machine. In *Advances in Neural Information Processing Systems, San Mateo, CA*, 2000.

[114] M. Tipping. Bayesian inference : an introduction to principles and practice in machine learning. *Advanced Lectures on Machine Learning*, 3176 :41–62, 2004.

[115] M. Tipping et C. Bishop. Mixtures of probabilistic principal component analysers. *Neural Computation*, 11(2) :443–482, 1999.

[116] M. Tipping et A. Faul. Fast marginal likelihood maximisation for sparse bayesian models. In *Proceedings of the 9th International Workshop on Artificial Intelligence and Statistics*, 2003.

[117] P. Tresadern. *Visual analysis of articulated motion.* PhD thèse, Robotics Research Group, Department of Engineering Science, University of Oxford, 2006.

[118] P. A. Tresadern et I. D. Reid. An evaluation of shape descriptors for image retrieval in human pose estimation. In *Proceedings of the British Machine Vision Conference*, 2007.

[119] S. Wachter et H.-H. Nagel. Tracking persons in monocular image sequences. *Computer Vision and Image Understanding*, 74(3) :174–192, 1999.

[120] J. J. Wang et S. Singh. Video analysis of human dynamics - a survey. *Real Time Imaging*, 9 :321–346, 2003.

[121] N. Werghi. A discriminative 3d wavelet-based descriptors : Application to the recognition of human body postures. *Pattern recognition letters*, 26(5) :663–677, 2005.

[122] O. Williams. A sparse probabilistic learning algorithm for real-time tracking. In *Proceedings of the International Conference on Computer Vision*, 2003.

[123] C.R. Wren, Azarbayejani A., T. Darrell, et A.P. Pentland. Pfinder : real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7) :780–785, 1997.

[124] T. Zhao et R. Nevatia. Bayesian human segmentation in crowded situations. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2003.

[125] T. Zhao et R. Nevatia. Tracking multiple humans in complex situations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9) :1208–1221, 2004.