Segmentation multi-échelle d'objets en haute résolution par approches CNN : Application au contrôle d'assemblage de pièces de carrosserie

Yoann Boussit

N°d'ordre NNT : 2022LYSES023

THESE de DOCTORAT DE L'UNIVERSITE DE LYON
opérée au sein du
Laboratoire Hubert Curien

Ecole Doctorale N° 488
Sciences Ingénierie Santé

Discipline de doctorat: Image, Vision et Signal

Soutenue publiquement le 27/06/2022, par :
Yoann BOUSSIT

Segmentation multi-échelle d'objets en haute résolution par approches CNN : Application au contrôle d'assemblage de pièces de carrosserie

Devant le jury composé de :

BOURENNANE, El-bay	Professeur d'université	Laboratoire ImViA	Président
VOISIN, Yvon	Professeur d'université	LICB Bourgogne	Rapporteur
CHAUSSE, Frédéric	Professeur d'université	Institut Pascal	Rapporteur
MORAND, Karynn	Invité	Segula Technologies	Examinatrice
KONIK, Hubert	Maître de conférences	LaHC – UJM	Co-directeur de thèse
FRESSE, Virginie	Maître de conférences	LaHC - UJM	Directrice de thèse

REMERCIEMENTS

Mes premiers remerciements vont à Mme Fresse qui m'a accompagné tout au long de ces trois ans. Merci beaucoup pour la direction et l'encadrement quotidien de mes travaux de thèse qui m'ont permis de réaliser ces travaux de façon sereine jusqu'à la fin.

Je remercie vivement M Konik d'avoir accepté le rôle de co-directeur de thèse. Vos compétences en image et vision m'ont offert une formidable opportunité d'approfondir mes connaissances et de parfaire ma réflexion scientifique.

Je remercie aussi Mme Morand pour m'avoir donné l'opportunité de réaliser une thèse au sein de l'entreprise SEGULA. De plus, je la remercie pour sa patience et sa résilience lors des relectures du manuscrit et des différents rapports.

J'aimerais vous remercier tous les trois pour l'écoute et la bienveillance que vous avez eu envers moi tout au long de ces trois ans et notamment dans les moments plus difficiles.

J'adresse aussi mes remerciements à Yvon Voisin et Frédéric Chausse d'avoir accepté le rôle de rapporteur et d'avoir évalué ce travail.

Un grand merci à mes différents collègues de bureau et de couloir : Cédric Maron, Ana Florencia Juarez Saborio, Lucas Grativol Ribeiro, An Hoang Thi Ngoc, Erwan De Lavergne De Cerval, Mathilde Prudent, Anthony Makhoul et Yannick Bleu. Merci pour tous ces bons moments passés ensemble notamment pour la pause douceur du vendredi matin ou encore le foot. Un immense merci à Cédric Maron pour ta présence, que ce soit dans les différents moments de réflexion, de partage de connaissances ou juste les bons moments durant cette dernière année. Je te remercie aussi pour ton implication dans ce travail qui m'a été d'une grande aide.

Je remercie Nadège Ollier et Yves Bringer pour avoir accepté d'être dans le jury de comité de suivi de thèse et de m'avoir suivi tout au long de ces trois années.

Je remercie mes amis et ma famille pour tout le soutien que vous m'avez apporté.

Enfin je remercie mon épouse, Marine Boussit, pour sa patience, son aide, sa bienveillance, sa résistance à la fatigue et son dévouement. Merci aussi pour toutes les fois où tu as dû expliquer ma thèse, où tu t'es cassé la tête pour comprendre mes travaux et pour ta relecture. Merci enfin pour tout ce que tu m'apportes au quotidien : l'amour, la joie, la complicité, la tendresse, la stabilité , le réconfort...

RÉSUMÉ

Le contrôle qualité des produits dans l'industrie, et notamment le contrôle du montage de véhicules industriels de type autobus et autocars, nécessite l'utilisation de systèmes de vision composés de matériels et d'algorithmes sophistiqués pour répondre à des exigences toujours accrues. Dans l'exemple des bus, l'acquisition et la segmentation des différentes pièces de carrosserie doivent avoir des erreurs de mesure et de segmentation très faible pour pouvoir contrôler le positionnement des pièces de carrosserie de l'arrière d'un bus et ainsi assurer l'esthétique du véhicule (ou qualité perçue). La solution proposée pour cela est composée de trois blocs : Le premier permet d'acquérir avec une faible erreur de mesure, c'est-à-dire en dessous du millimètre, la profondeur de la scène grâce à une combinaison de deux algorithmes de lumière structurée. Le premier algorithme est utilisé pour régler au millimètre et au degré près le système lors de la prise d'image des plans de référence et le deuxième permet d'utiliser ces plans pour estimer la profondeur de façon contrôlée avec une faible erreur. Le deuxième bloc permet d'améliorer la segmentation multi-échelle des pièces de carrosserie grâce à des images de couleur et de profondeur (RGB-D). Pour cela, une méthode hybride est proposée. Elle utilise un CNN pour segmenter une première fois la scène en faible résolution, puis les contours actifs pour améliorer la segmentation issue du CNN. Une exploration des forces inhérentes aux contours actifs est réalisée pour pouvoir les automatiser. Le troisième bloc, non exploré durant cette thèse, devra permettre d'estimer l'écart entre les pièces de carrosserie et ainsi garantir un bon aspect extérieur du véhicule. Enfin, un outil de génération d'images de synthèse en haute résolution est créé pour expérimenter et valider la solution proposée sur l'application des bus, que ce soit pour l'acquisition de la profondeur ou pour la segmentation des pièces.

Mots clés : segmentation, multi-échelle, RGB-D, CNN, contours actifs, grande dimension, méthode hybride, lumière structurée, faible erreur de mesure, haute résolution, bus.

ABSTRACT

The quality control of products in industry, and in particular the control of the assembly of industrial vehicles such as buses and coaches, requires the use of vision systems composed of sophisticated hardware and algorithms to meet ever increasing requirements. In the example of buses, the acquisition and segmentation of the various body parts must have very low measurement and segmentation errors in order to be able to control the positioning of the body parts at the rear of a bus and thus ensure the aesthetics of the vehicle (or perceived quality). The proposed solution consists of three blocks : The first one allows to acquire with a low measurement error, i.e. below the millimetre, the depth of the scene thanks to a combination of two structured light algorithms. The first algorithm is used to set the system to millimetre and degree accuracy when imaging the reference planes and the second allows these planes to be used to estimate depth in a controlled manner with low error. The second block is used to improve the multi-scale segmentation of body parts using colour and depth images (RGB-D). For this purpose, a hybrid method is proposed. It uses a CNN to first segment the scene in low resolution, and then the active contours to improve the segmentation from the CNN. An exploration of the forces inherent in the active contours is carried out in order to automate them. The third block, not explored during this thesis, should allow the estimation of the gap between the body parts and thus guarantee a good external appearance of the vehicle. Finally, a tool for generating high-resolution synthetic images is created to experiment and validate the proposed solution on the bus application, both for depth acquisition and for part segmentation.

Key words : segmentation, multi-scale, RGB-D, CNN, active contours, large dimension, hybrid method, structured light, low measurement error, high resolution, bus.

TABLE DES MATIÈRES

TABLE DES FIGURES

LISTE DES TABLEAUX

CHAPITRE 1

INTRODUCTION

1.1 Contexte Général

Depuis la transformation numérique des entreprises et notamment depuis le passage à l'industrie 3.0 et 4.0, le produit fini doit répondre à toujours plus de contraintes, que ce soit au niveau métrologique, sanitaire, sécuritaire ou encore esthétique. Tout cela engendre une augmentation des contrôles sur les caractéristiques du produit. Pour ne pas détériorer l'objet avec ces mesures, des méthodes de contrôle non-destructif sont utilisées, comme les systèmes de vision industrielle par exemple. Un système de vision est une combinaison de matériel et de logiciel. Une telle solution est basée sur l'acquisition de la scène d'une part, et le traitement d'images pour en extraire des informations d'autre part, et ainsi répondre à un service défini. Les systèmes de vision industrielle reposent généralement sur une ou plusieurs caméras industrielles, ainsi que, pour certains systèmes, d'autres émetteurs et récepteurs tel qu'un projecteur ou un Lidar. Ils sont utilisés pour contrôler les caractéristiques métrologiques du produit comme ses dimensions, le placement des pièces constitutives, la déformation et les imperfections du produit, ou encore le rendu visuel. Les caractéristiques à analyser sont donc spécifiques à chaque produit considéré. Le cas d'étude sélectionné dans le cadre de cette thèse est le contrôle du montage des véhicules industriels et plus particulièrement ceux de type autobus et autocar.

Les autocars et autobus sont produits en petites voire très petites séries par rapport à

l'industrie automobile. Pour un constructeur, tel qu'IVECO BUS par exemple, ce sont environ 1600 bus qui sont montés chaque année sur la ligne d'assemblage. L'industrie automobile se doit de produire rapidement et en grandes quantités des véhicules alors que les constructeurs d'autobus et autocars se concentrent sur la personnalisation de ces produits pour répondre aux attentes de leurs clients. Cette personnalisation et la faible production rendent difficile l'automatisation du montage et du contrôle des véhicules sur des chaînes de production situées dans un environnement contrôlé.

Ainsi, contrairement aux voitures, l'assemblage des pièces d'autocars et autobus conserve encore actuellement une forte part manuelle. Cela peut s'avérer délicat pour le montage des pièces de carrosserie extérieures dont certaines peuvent atteindre de grandes dimensions. En particulier, le réglage de position d'une pièce par rapport à une autre pose des difficultés récurrentes, liées aux conditions de montage en usine. La Figure 1.1 montre une zone où il est nécessaire de régler le jeu entre deux pièces de carrosseries.

FIGURE 1.1 – Exemple d'une zone d'un bus où le jeu entre deux pièces (manchette et coins rond droit) doit être réglé

Sur les postes de montage de la face arrière, une quinzaine de pièces environ sont assemblées successivement : la girouette, le bloc arrière, les coins et manchettes, le hayon, etc. (Figure 1.2).

Le temps passé sur un poste, en général entre 30 et 45 minutes, peut varier en fonction des difficultés rencontrées par les opérateurs. Surtout, la méthode d'assemblage actuelle entraîne une reprise quasi-systématique du positionnement des pièces sur le poste de retouche afin que les jeux autour de chaque pièce soient homogènes. En effet, le contrôle qualité de la face arrière d'un bus est réalisé à l'aide d'outils qui ne permettent pas d'estimer assez finement le positionnement des pièces car ces outils ne fournissent pas une vue d'ensemble de la scène. Les contrôles sont ainsi localisés, sans garantie que l'espacement sur un côté de la pièce est bien le même sur le côté opposé

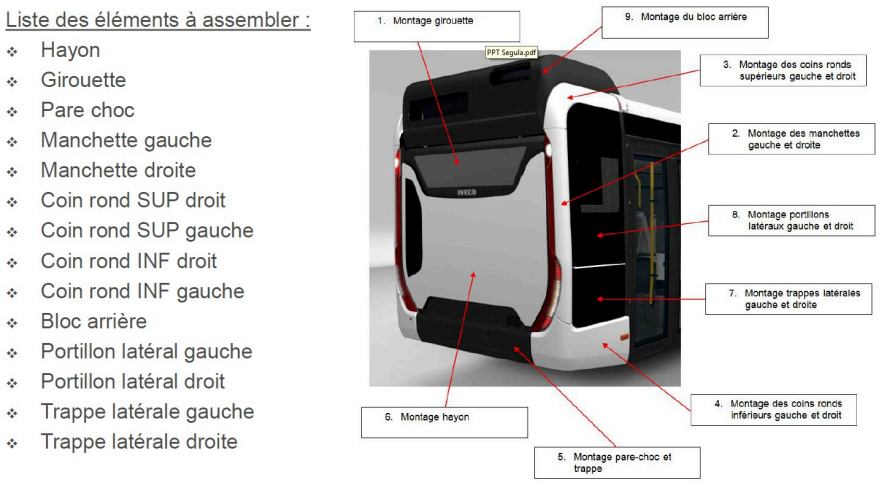


FIGURE 1.2 – Pièces de la face arrière d'un véhicule IVECO Bus

(selon une symétrie axiale de la pièce). De plus, le fait que le véhicule et les pièces soient de grandes dimensions n'aide pas à estimer les erreurs de positionnement.

L'objectif de la thèse est de proposer une solution permettant de contrôler l'assemblage de pièces de la face arrière d'un bus. Pour résoudre cela, le choix est fait d'utiliser un système de vision qui est composé de 3 blocs fonctionnels (Figure 1.3) :

— l'estimation de la profondeur de la scène ;

— la segmentation des objets de la scène de façon fine ;

— la vérification du placement des pièces.

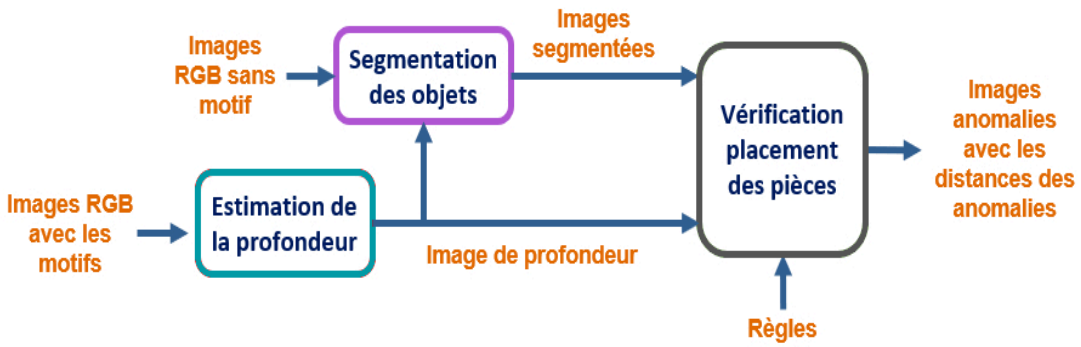


FIGURE 1.3 – Blocs composant le système de vision proposé

Le bloc fonctionnel **"Vérification de placement des pièces"** permet de délivrer les images d'anomalies de la scène, c'est-à-dire toutes les zones où les pièces sont mal placées. Il nécessite en entrée un ensemble de règles (écarts théoriques, règles de symétrie) et les écarts/jeux réels des pièces de la scène. Les écarts réels entre les pièces doivent être mesurés avec une erreur très faible pour respecter les règles fixées par le

constructeur en fonction du modèle et des pièces. Pour cela, il est nécessaire de connaître la mesure en dimensions et en tout point de la scène, avec une erreur très faible. Il faut également connaître l'emplacement de chaque pièce du bus et en particulier, identifier leurs bords.
La mesure selon les trois dimensions est obtenue grâce au bloc **"Estimation de la profondeur"**. Une image de profondeur peut par la suite être obtenue grâce aux mesures en trois dimensions.
L'emplacement et le bord des objets sont obtenus grâce au bloc de **"Segmentation des objets"** qui délivre, à la fin du processus, une image segmentée.

Développer ces blocs permettant de répondre à la problématique du contrôle d'assemblage de la face arrière d'un bus est la motivation pour laquelle la société SEGULA Technologies [1] a engagé cette thèse. Cette solution doit cependant pouvoir s'adapter à d'autres contextes. En effet, SEGULA Technologies en tant que groupe d'ingénierie pourrait être amené à proposer ces solutions pour d'autres applications industrielles et d'autres clients que les constructeurs d'autobus et autocars. La thèse porte sur les deux premiers blocs car le dernier est spécifique à chaque constructeur. Cette thèse CIFRE est encadrée par Mme Morand, Responsable Recherche et Innovation pour la société SEGULA Technologies, et Mme Fresse et M. Konik, Maîtres de conférence au laboratoire Hubert Curien.

1.2 Cahier des charges

Une solution non invasive doit être trouvée pour permettre de réduire l'erreur de positionnement des pièces au montage et améliorer l'ergonomie des travaux, sans pénaliser le temps nécessaire au réglage (temps réel) mais en s'exonérant d'une reprise systématique sur le véhicule. La perspective finale est de pouvoir fournir aux opérateurs un dispositif intelligent qui leur indique les opérations de réglage nécessaires en se basant sur la position des pièces en cours de montage par rapport à leur position cible pour atteindre le niveau de finition requis sur le véhicule. Ce dispositif doit répondre aux critères listés ci-dessous et doit être évolutif pour prendre en compte de nouvelles contraintes ou technologies qui pourraient intervenir. Pour cela la solution doit respecter les points suivants.

1. Groupe d'ingénierie français, SEGULA Technologies intervient dans de nombreux secteurs industriels comme l'automobile, le naval, le ferroviaire, etc. Au titre de ses activités, SEGULA intervient régulièrement comme bureau d'études externalisé pour ses clients, des grands noms de l'industrie. Malgré ses activités diversifiées, SEGULA n'assure pas le montage des autobus et autocars. A ce titre, l'accès à des usines d'assemblage ne dépend pas seulement de l'industriel mais aussi de ses clients, fabricants des véhicules.

Erreur de mesure

Outre sa simplicité de mise en œuvre, la solution proposée doit permettre de mesurer, avec une faible erreur de mesure, la position relative des pièces entre elles. Il s'agira ainsi de s'assurer que l'espacement entre deux éléments reste constant sur toute la longueur de l'arête. A ce titre, la mesure doit avoir une erreur de mesure maximale de 1 millimètre.

L'erreur de mesure reflète le fait qu'il est impossible d'acquérir exactement les valeurs de l'objet de la scène. Pour caractériser l'erreur, la moyenne arithmétique est utilisée pour caractériser l'erreur systématique , et l'écart type pour caractériser l'erreur aléatoire, c'est à dire la dispersion des résultats, de la solution.

Les dimensions hors-tout, i.e. les dimensions maximales, mesurées toutes saillies comprises, hors rétroviseurs, feux de gabarit et indicateur de changement de direction, de ces véhicules sont les suivantes : H : 3,50m, L : 2,55m, P : 12 ou 18m. La longueur P de l'autobus n'intervient pas directement sur la problématique de positionnement des pièces de la face arrière. Le volume à analyser est assimilé à un pavé de 4m x 3m x 1m50 (H x L x P), ce qui constitue la "scène" dans laquelle se déroulent les travaux (Figure 1.4). Ce volume est calculé à partir de la taille du bus. Cela tient aussi compte du fait que la position d'un véhicule dans la zone de montage reste approximative (deux autobus arrivant successivement sur le poste de montage ne seront, a priori, pas positionnés strictement à la même place). Cela a une influence notable sur les opérations de contrôle de positionnement des pièces qui doivent être réalisées.

Par conséquent, il devient nécessaire de prendre en compte dans la solution proposée, la distance du véhicule par rapport au dispositif de contrôle, i.e. la profondeur de la scène, et ce en tout point des pièces concernées par la mesure. En effet, il n'est pas assuré que la face arrière soit dans un plan parallèle à celui du système d'acquisition. Pour garantir une mesure entre les pièces valide au millimètre près, il est décidé de mesurer la profondeur en tout point de la scène (nécessaire au contrôle) en réduisant, là encore, l'erreur de mesure à 1 millimètre au maximum.

Pour cela, le système de vision sera placé à 5m du bus, avec une plage d'acquisition en profondeur allant de 4m25 et 5m75. Ces valeurs sont définies pour prendre en compte deux contraintes contradictoires. Tout d'abord, le dispositif doit être assez loin pour projeter des mires sur l'intégralité de la surface du bus. A l'inverse, il doit être assez proche pour ne pas gêner les opérations en cours et les déplacements dans l'atelier.

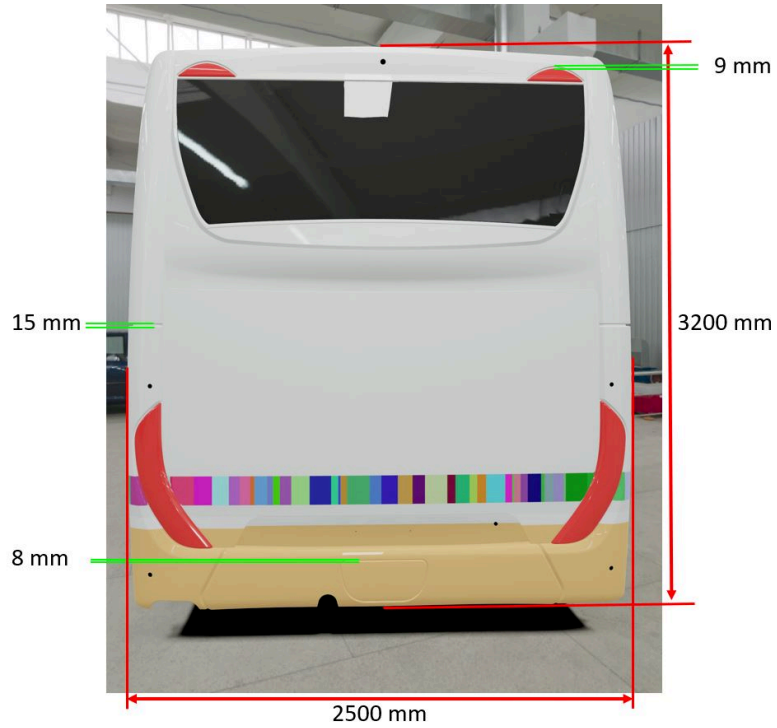


FIGURE 1.4 – Face arrière d'un bus avec des exemples de côtes et de valeurs d'espacement entre les pièces de carrosserie

Adaptabilité aux pièces

Dans un véhicule, une quinzaine de pièces sont à positionner sur la face arrière. Elles sont caractérisées par des dimensions très disparates (quelques centimètres pour les feux supérieurs jusqu'à une hauteur d'environ 1m50 pour le hayon), par des formes et des couleurs différentes. De plus, plusieurs gammes d'autobus passent successivement sur chaque poste de montage. Avec cette grande diversité de véhicules assemblés sur la chaîne, la solution recherchée doit donc être fortement adaptable.

Adaptabilité à l'environnement

Bien que le montage des autobus et autocars soit réalisé à l'intérieur, dans un atelier, l'environnement n'est pas parfaitement maîtrisé et des changements peuvent intervenir. L'environnement fait référence à :

— l'éclairage qui peut varier en fonction de la météo (temps ensoleillé ou couvert) et de l'heure du contrôle (jour ou nuit) ;

— le fond de l'atelier qui est modifié en permanence du fait de l'activité industrielle ;

— le positionnement du véhicule sur la ligne d'assemblage, qui peut varier entre deux véhicules en raison de la mise en place qui est faite "au jugé" par les opérateurs.

Il faut donc que cette solution soit suffisamment robuste pour que ses performances ne soient pas impactées par ces contraintes.

Rapidité

Enfin, la solution doit pouvoir être exécutée dans un temps relativement court, fixé en première approche à 5 minutes pour l'ensemble du processus pour ne pas ralentir les autres opérations sur le poste de montage. Le traitement de l'information nécessite de pouvoir paralléliser des tâches. Au niveau matériel, cela se traduit par l'utilisation des GPU pour pouvoir accélérer les fonctions et algorithmes mis en place. Les méthodes mises en oeuvre seront donc adaptées à ce matériel et à sa structure interne multicoeur pour optimiser la solution proposée et minimiser le temps de traitement.

1.3 Contributions

Les contributions réalisées durant cette thèse sont au nombre de 3 :

— Méthode hybride permettant l'acquisition avec une faible erreur de mesure d'une scène 3D à longue distance ;

— Méthode de segmentation multi-échelle, couplant CNN et CA, d'une scène en haute résolution ;

— Conception d'un outil de génération d'une base de données de synthèse en haute résolution.

1.3.1 Méthodes d'estimation de la profondeur

Plusieurs systèmes de vision permettent d'acquérir la profondeur d'une scène. Ils sont classés en deux catégories (systèmes passifs ou actifs) en fonction de la manière dont l'acquisition est réalisée.

1. Les systèmes passifs utilisent uniquement des capteurs, comme des caméras. La stéréovision [6] et la photogrammétrie [4] sont les deux méthodes les plus utilisées ;
2. Les systèmes actifs projettent, dans la scène à analyser, des informations grâce à un émetteur. Les caméras temps de vol (Time of Flight) [7], le scan à impulsion laser [10], la triangulation laser [5], la lumière structurée [14] et l'interférométrie [3] sont les méthodes les plus utilisées de cette catégorie.

Les caméras utilisées pour les méthodes temps de vol, scan à impulsion laser et interférométrie ne sont pas des systèmes adéquats pour notre problématique car il est nécessaire d'obtenir une image de profondeur associée à l'image couleur. Cela est important pour pouvoir par la suite segmenter finement les pièces de carrosserie du bus.

Les systèmes utilisant la photogrammétrie sont écartés car leur temps de traitement est relativement long. Ils sont, de plus, peu adaptés à une utilisation en atelier. En effet, ils nécessitent d'obtenir des images de différents points de vues avec une erreur de positionnement du système faible lors de l'acquisition.

Les systèmes de visions étudiés par la suite sont donc la triangulation laser, la stéréovision et la lumière structurée.

1.3.1.1 Systèmes de visions

Triangulation laser

Le système utilise un laser et une caméra pour estimer la profondeur. Un point ou un trait est projeté par le laser sur l'objet. Le principe est de faire le lien entre le point/trait perçu par la caméra et la position du laser. La triangulation utilise le lancer de rayon pour connaître la profondeur de la scène. Les caractéristiques de la caméra, focale et centre optique, ainsi que la position et l'orientation du laser par rapport à la caméra doivent donc être connus.

Ce type de système permet de délivrer une profondeur au millimètre, voire au micromètre près. Il permet d'acquérir de multiples points à chaque acquisition (haute densité). Cependant l'acquisition de la profondeur ne se fait qu'à l'emplacement du laser sur l'objet. Il est donc nécessaire de déplacer le laser, et donc le système, sur l'objet pour pouvoir acquérir la profondeur de l'objet en entier, ou alors de déplacer l'objet lui même. Les systèmes industriels utilisant la triangulation laser peuvent être utilisés pour scanner des objets situés entre 15 et 100 cm de ceux-ci.

Stéréovision

La stéréovision utilise plusieurs caméras, généralement deux, pour faire le lien entre plusieurs points caractéristiques (coins, bordures, intersections...) sur des images issues des deux caméras. Ce système, également, utilise le lancer de rayon pour connaître la profondeur de la scène. Pour estimer la profondeur, il est nécessaire de connaître les caractéristiques des caméras, c'est-à-dire leurs focales et leur centres optiques, ainsi que la position et l'orientation de la première caméra par rapport à la deuxième. La qua-

lité d'acquisition de ce type de système est tributaire de la texture de l'objet à analyser. En effet, plus la texture de l'objet contient des formes complexes, plus il y a de points caractéristiques et plus la densité d'acquisition de la scène, c'est-à-dire le nombre de points estimés sur la scène pour lesquels la profondeur est connue, sera élevée. Ce type de système permet de délivrer une profondeur de l'ordre du millimètre, voire moins, a fortiori si la résolution des caméras est importante. Il est utilisé pour des objets positionnés à une distance de 0,2 à 50 m de la caméra.

Lumière Structurée

La lumière structurée, quant à elle, utilise une caméra et un projecteur. Le projecteur affiche des motifs sur l'objet à analyser. En fonction des motifs projetés, l'estimation de la profondeur n'est pas réalisée de la même manière. Il existe cependant deux grands principes pour estimer la profondeur : soit par lancer de rayon, soit par décalage de phase.

Le lancer de rayons permet d'acquérir une grande quantité de points sur l'objet (haute densité) de la dizaine de micromètre au millimètre près. Le décalage de phase utilise le décalage spatial des motifs projetés sur l'objet. Il impose de faire, au préalable, l'acquisition d'une surface de référence. Celle-ci est utilisée pour obtenir la profondeur relative (par rapport à la surface de référence) de l'objet. Ce principe permet d'acquérir une grande quantité de points sur l'objet (haute densité) avec une erreur de mesure allant du micromètre au millimètre. De plus, il est possible d'améliorer les résultats grâce à l'utilisation des images au subpixel, c'est-à-dire entre les pixels.

Les systèmes industriels utilisant la lumière structurée sont calibrés pour être placés de 2 à 50 cm de l'objet à analyser. Ils peuvent acquérir en général, la profondeur d'un volume maximal de 50 x 35 x 30 cm. Cependant ils peuvent être utilisés sur de grandes dimensions.

1.3.1.2 Choix du système

Grâce au Tableau 1.1, il est possible de comparer les différentes méthodes étudiées précédemment.
La carrosserie des autobus et autocars pouvant être de couleur unie, la densité d'acquisition de points ne sera pas homogène et sera faible sur une grande partie de la scène. Dans ces conditions, une solution s'appuyant sur la stéréovision ne semble pas convenir et n'est donc pas sélectionnée.

La triangulation laser est peu sensible à l'éclairage extérieur et très rapide pour délivrer la profondeur de la zone analysée. Toutefois, il n'est pas possible d'acquérir la scène dans son intégralité en une seule prise. Elle n'est donc pas retenue.
La lumière structurée semble ainsi le système le plus approprié en réponse à notre problématique.

TABLE 1.1 – Caractéristiques des 3 systèmes de vision retenus pour l'acquisition de la profondeur de la scène

Système 3D	Rapidité d'exécution	Erreur de mesure	Densité des points	Sensible à l'éclairage extérieur	Acquisition en une seule prise
Stéréovision	Très rapide	Moyenne	Faible	Oui	Oui
Lumière Structurée	Rapide	Moyenne à très faible	Haute	Peu	Oui
Triangulation Laser	Très rapide	faible	Haute	Peu	Non

En effet, le système délivre une profondeur avec une faible erreur de mesure pour de petites pièces (micromètres). Il est peu sensible aux lumières extérieures en fonction de l'intensité de la lumière projetée sur l'objet et en fonction des algorithmes choisis pour analyser l'objet. Enfin, il permet d'acquérir une surface pouvant être conséquente, en une seule fois.

Certains distributeurs tels que Stemmer Imaging, Faro, Isra Vision ou encore Cognex proposent des solutions intégrées. Ces systèmes regroupent les aspects matériels et algorithmiques. Cependant, dans les propositions des systèmes de vision industriels permettant d'acquérir la profondeur de l'objet, aucun dispositif ne correspond tout à fait aux exigences énoncées dans le chapitre 1.2. En effet, plusieurs points contraignants ne permettent pas de mettre en œuvre directement un système industriel complet et de faire les mesures sur un objet plus grand que ce qui est prévu par les spécifications du constructeur. Si les systèmes peuvent fonctionner sur des objets plus grands et plus éloignés, l'information qu'ils fournissent comporte plus d'erreur. Ils ne permettent pas d'obtenir la valeur recherchée. La faible erreur de mesure de l'estimation de la profondeur du système de vision pour des objets lointains constitue donc le verrou à lever dans ce bloc. La contribution proposée [18] combine plusieurs méthodes de lumière structurée pour le résoudre.

1.3.2 Méthode de segmentation multi-échelle d'une scène en haute résolution

La plupart des méthodes de segmentation prennent en entrée l'image complète. En lien avec l'ensemble des objectifs du projet, c'est-à-dire l'adaptabilité du dispositif, sa

fiabilité et sa robustesse, l'algorithme alliant les meilleures performances pour segmenter des objets inconnus a priori est le CNN (Convolutional Neural Network). Les CNN utilisent énormément de ressources, notamment lors de l'entraînement, pour pouvoir se paramétrer ce qui limite la taille des images en entrée. Pour segmenter une image ayant une résolution très importante par rapport aux ressources matérielles disponibles, deux solutions sont envisageables : la réduction des images [8] [2] ou le partitionnement des images en tuiles (i.e. des "fragments" de l'image d'entrée) [15] [1]. Ce partitionnement est réalisé de façon fixe (par exemple, 4 tuiles par image) ou grâce à une fenêtre qui se déplace dans l'image.

La première solution est optimale lorsque :

— la segmentation peut être grossière ;

— l'objet à segmenter n'est pas déterminé par un très faible nombre de pixels ;

— la réduction de l'image originale est faible.

Elle est utilisée dans la majorité des applications car l'erreur pixelique n'est pas requise et les objets dans la scène sont composés d'un nombre important de pixels.

La deuxième solution est utilisée majoritairement dans les applications de type analyse d'image aérienne, médicale ou multispectrale. En effet, elle est optimale lorsque :

— l'image est en haute résolution ;

— les objets à analyser sont petits ;

— la réduction est trop importante par rapport aux objets à analyser ;

— la segmentation ne doit pas comporter d'erreur.

Dans le cas du bus, l'erreur de segmentation voulue doit être pixelique et l'image d'entrée du CNN est en très haute résolution. La deuxième solution semble donc toute désignée. Cependant, des pièces de très grandes dimensions sont également présentes dans l'image et ne peuvent être découpées au risque de perdre les informations nécessaires pour une segmentation en haute résolution sans erreur (par exemple le hayon au centre de la face arrière).

Ces contraintes forment un verrou car il est impossible d'utiliser un CNN avec des images haute résolution. Cependant ces deux contraintes doivent être toutes les deux valides pour pouvoir respecter le cahier des charges.

La solution proposée permet de combiner les deux approches pour segmenter avec une faible erreur tous les objets de la scène (Figure 1.5) [17].

La solution prend en entrée une image RGB-D (couleur + profondeur) en haute résolution et donne en sortie une image segmentée en haute résolution. Elle est découpée

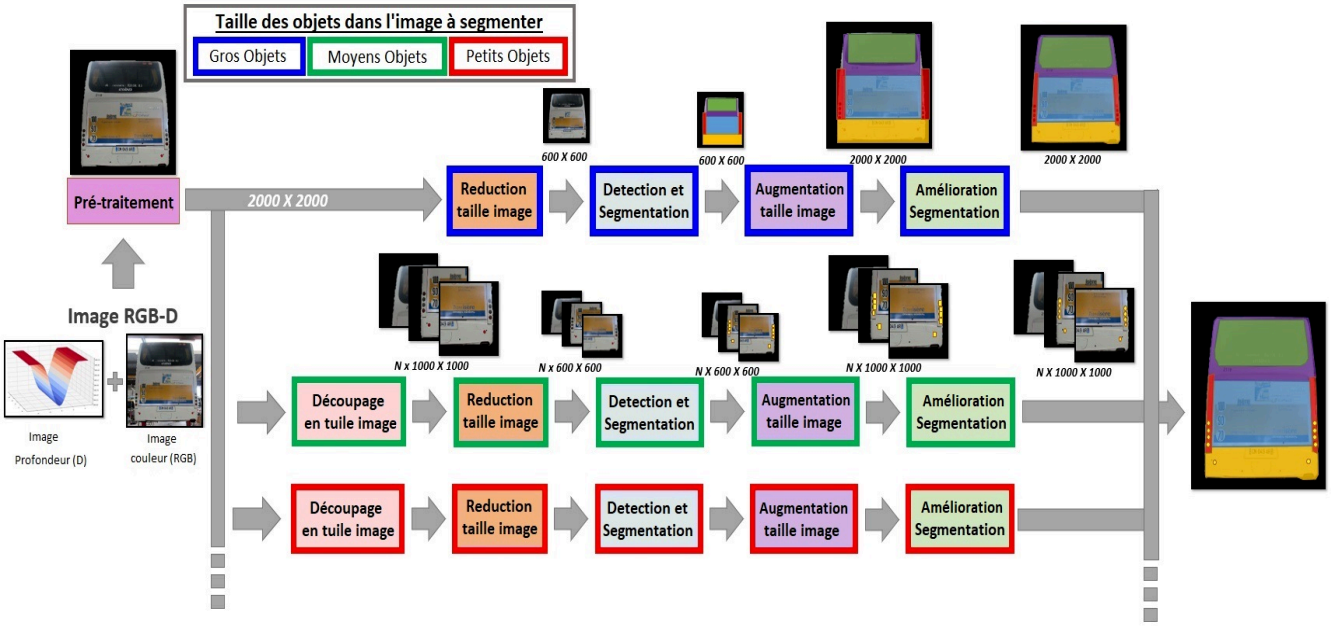


FIGURE 1.5 – Méthode proposée permettant de segmenter avec une erreur très faible toutes les pièces du bus

en branches qui permettent de segmenter les objets en fonction de leur taille. Les plus grands objets de l'image sont segmentés en premier. Viennent ensuite les objets de taille moyenne puis ceux de petite taille.

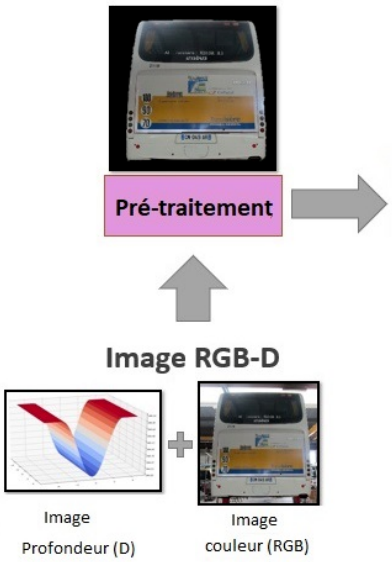


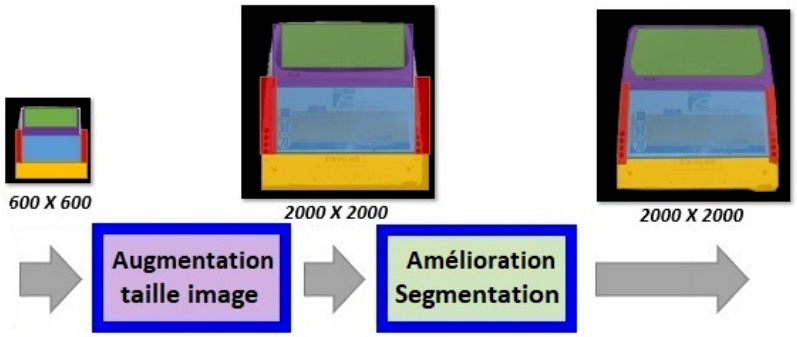
FIGURE 1.6 – Soustraction du fond

La première étape du processus consiste à enlever le fond pour ne pas avoir d'erreurs lors de la segmentation. Cela peut être réalisé notamment avec l'image en profondeur

(Figure 1.6). En effet, les objets sont dans un fond « lointain », ils ont donc une valeur de profondeur élevée.

L'image en haute résolution sans le fond passe ensuite dans la première branche. Cette image est réduite pour passer dans une méthode hybride. Celle-ci combine une méthode utilisant des CNN pour réaliser une première segmentation et un algorithme de traitements d'images pour améliorer cette segmentation.

(a) Blocs permettant de réduire l'image et de la segmenter les objets de taille importante



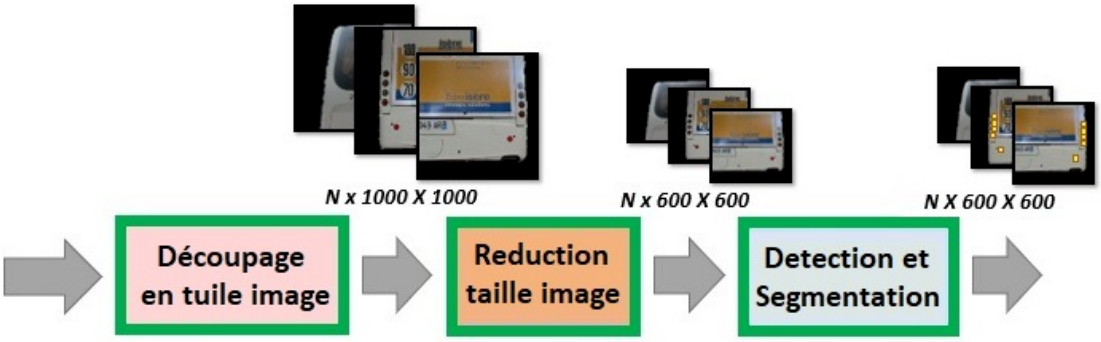
(b) Blocs permettant d'améliorer la segmentation les objets de taille importante

FIGURE 1.7 – Étapes de la méthode permettant d'améliorer la segmentation des objets de taille importante

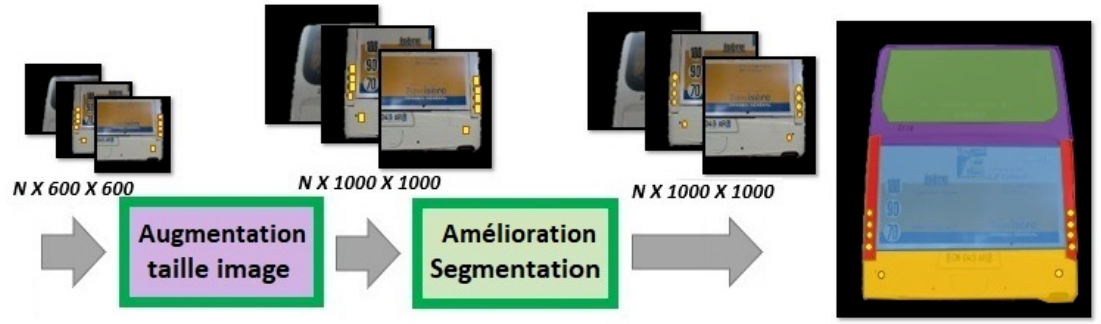
Dans la solution multi-échelle, le CNN segmente les objets les plus imposants dans la première branche (Figure 1.7a). Cette segmentation est mise à l'échelle pour retrouver la pleine résolution de l'image grâce à des interpolations. Cette image segmentée en haute résolution est améliorée avec l'algorithme de traitement d'images (Figure 1.7b).

Lorsque l'objet est segmenté, l'image est passée dans la deuxième branche pour trouver des objets de taille moyenne. L'image est découpée en tuiles qui sont ensuite passées dans le CNN (Figure 1.8a). Cela permet de garder au maximum les détails des objets pour la segmentation. Les tuiles comportent une zone de recouvrement calculée pour que le plus grand des objets de "taille moyenne" à analyser puisse être vu en entier

dans au moins une image. En sortie du CNN, l'image segmentée est mise à sa résolution initiale et comme pour la branche précédente, la segmentation est améliorée grâce aux contours actifs (Figure 1.8b).



(a) Blocs permettant de réduire l'image et de la segmenter de taille moyenne



(b) Blocs permettant d'améliorer la segmentation d'objets de taille moyenne

FIGURE 1.8 – Étapes de la méthode permettant d'améliorer la segmentation des objets de taille moyenne

La méthode hybride proposée pour la segmentation fine d'un objet est composée d'un CNN et d'un contour actif [16]. Le CNN permet la segmentation de l'image et le contour actif permet de déformer un contour initial vers les bords des objets grâce à des forces spécifiques qui seront développées dans le manuscrit. Le contour est initialisé avec l'image de segmentation issue du CNN.

1.3.3 Conception d'un outil de génération d'une base de données de synthèse en haute résolution

Les CNN utilisent des bases conséquentes pour être paramétrés et prédire de façon fiable et avec une faible erreur la segmentation de la scène. La base nécessaire pour le cas du bus doit être composée d'images RGB-D (couleurs + profondeur) et labélisées en haute résolution. La scène considérée doit représenter des bus dotés de textures,

éclairages et positionnements différents. Bien qu'il existe plusieurs bases contenant des images RGB-D et labélisées, comme NYUv2 [9] ou SUNRGBD[13], ou alors en haute résolution [11][12], ces bases ne correspondent pas tout à fait aux exigences énoncées précédemment. Le troisième verrou provient donc du fait qu'il n'existe pas de bases de données sur les bus et qu'il n'est pas possible d'en acquérir une réelle tant que le système de vision complet proposé n'est pas fonctionnel. De plus, les temps d'acquisition et de traitement sont très longs. Enfin, la segmentation finale ne sera pas parfaite car réalisée par un humain. Pour lever ce verrou, une base de données de synthèse, spécifique aux autobus, est créée en tenant compte de l'ensemble des exigences énoncées précédemment.

1.4 Organisation du manuscrit

Le manuscrit de thèse est découpé en 3 parties.

La première partie correspond à l'état de l'art des méthodes d'acquisition utilisant la lumière structurée, ainsi qu'au développement et à l'analyse de la méthode proposée pour répondre à la problématique d'acquisition des objets de grandes dimensions (plusieurs mètres de côté) au millimètre près.

La seconde partie répondra à la problématique d'amélioration de la segmentation grâce aux images RGB-D. Pour cela, un état de l'art des méthodes de segmentation tel que les CNN et contours actifs, ainsi que les méthodes les combinant est réalisé. La méthode proposée est par la suite décrite et analysée.

La dernière partie présentera la solution complète appliquée au cas des autobus, depuis la génération des images de la scène à l'analyse des différents blocs présentés précédemment.

Enfin, une conclusion et des perspectives termineront ce mémoire.

BIBLIOGRAPHIE

[1] Mark AMO-BOATENG et al. "Instance segmentation scheme for roofs in rural areas based on Mask R-CNN". In : *The Egyptian Journal of Remote Sensing and Space Science* (2022). ISSN : 1110-9823. DOI : https://doi.org/10.1016/j.ejrs.2022.03.017. URL : https://www.sciencedirect.com/science/article/pii/S1110982322000461.

[2] Anupam DAS. "Adaptive UNet-based Lung Segmentation and Ensemble Learning with CNN-based Deep Features for Automated COVID-19 Diagnosis". In : *Multimedia Tools and Applications* (avr. 2022).

[3] GEOREVA. *HoIBIS-FM -Interférométrie radar*. 2022. URL : https://georeva.eu/fr/produit/ibis-fm/.

[4] Arun GYAWALI et al. "Comparison of Individual Tree Height Estimated from LiDAR and Digital Aerial Photogrammetry in Young Forests". In : *Sustainability* 14.7 (2022). ISSN : 2071-1050. DOI : 10.3390/su14073720. URL : https://www.mdpi.com/2071-1050/14/7/3720.

[5] STEMMER IMAGING. *3D laser scanning sensors*. 2021. URL : https://www.stemmer-imaging.com/en-gb/products/category/3d-laser-scanning-sensor/.

[6] STEMMER IMAGING. *3D stereo cameras*. 2021. URL : https://www.stemmer-imaging.com/en-gb/products/category/3d-stereo-cameras/.

[7] STEMMER IMAGING. *3D Time of Flight*. 2021. URL : https://www.stemmer-imaging.com/en-gb/products/category/Time-of-flight/.

[8] Seifedine KADRY et al. "Automated segmentation of leukocyte from hematological images-a study using various CNN schemes". In : *The Journal of Supercomputing* 78 (avr. 2022). DOI : 10.1007/s11227-021-04125-4.

[9] Pushmeet Kohli NATHAN SILBERMAN Derek Hoiem et Rob FERGUS. "Indoor Segmentation and Support Inference from RGBD Images". In : (2012).

[10] Qi QIU et al. "An adaptive down-sampling method of laser scan data for scan-to-BIM". In : *Automation in Construction* 135 (2022), p. 104135. ISSN : 0926-5805. DOI : https://doi.org/10.1016/j.autcon.2022.104135. URL : https://www.sciencedirect.com/science/article/pii/S0926580522000085.

[11] Maryam RAHNEMOONFAR et al. "FloodNet: A High Resolution Aerial Imagery Dataset for Post Flood Scene Understanding". In : *IEEE Access* 9 (2021), p. 89644-89654. DOI : 10.1109/ACCESS.2021.3090981.

[12] Amber L. SIMPSON et al. "A large annotated medical image dataset for the development and evaluation of segmentation algorithms". In : *ArXiv* abs/1902.09063 (2019).

[13] Shuran SONG, Samuel P. LICHTENBERG et Jianxiong XIAO. “SUN RGB-D: A RGB-D scene understanding benchmark suite”. In : (2015), p. 567-576. DOI : 10.1109/CVPR.2015.7298655.

[14] *StereoScan neo*. fr-FR. URL : https://www.hexagonmi.com/fr-FR/products/structured-light-scanners/aicon-stereoscan-neo (visité le 30/06/2022).

[15] Shunjun WEI et al. “HRSID: A High-Resolution SAR Images Dataset for Ship Detection and Instance Segmentation”. In : *IEEE Access* 8 (2020), p. 120234-120254. DOI : 10.1109/ACCESS.2020.3005861.

[16] Boussit YOANN et al. “CNN-CA: Convolutional Neural Network Combined With Active Contour for Image RGB-D Segmentation”. In : *Proceedings of Seventh International Congress on Information and Communication Technology: ICICT 2022, London, Volume 4*. 2022.

[17] Boussit YOANN et al. “Quality control based on deep learning methods in RGB-D Images”. In : 4th European Machine Vision Forum, 2019.

[18] Boussit YOANN et al. “Structured light algorithm for low 3D measurement error at long distances”. In : Applied Optics, Submitted.

CHAPITRE 2

ASSOCIATION DE MÉTHODES DE LUMIÈRE STRUCTURÉE ESTIMANT LA PROFONDEUR D'OBJETS DE GRANDES DIMENSIONS AVEC UNE FAIBLE ERREUR

La majorité des caméras actuelles acquièrent seulement la scène en 2 dimensions, c'est-à-dire une projection du monde en 3 dimensions sur le capteur de la caméra.

Des systèmes de visions permettant d'acquérir une scène en couleur et en 3D avec un seul capteur existent et sont au centre de plusieurs sujets de recherches.

Cependant, les systèmes de vision utilisant plusieurs capteurs et emetteurs permettent d'obtenir la profondeur de la scène avec une plus grande résolution et/ou une plus faible erreur de mesure que les systèmes de vision précédents.[30][17][3]

Les systèmes de vision non destructifs, c'est-à-dire n'étant à aucun moment en contact avec l'objet et ne le détériorant pas, sont utilisés en industrie. Ces systèmes non destructifs peuvent être des systèmes actifs, c'est-à-dire qu'ils utilisent une source de lumière extérieure pour extraire les informations 3D de l'objet, ou passifs, i.e. qui n'utilisent pas de source extérieure. Les systèmes passifs sont généralement choisis car ils permettent d'acquérir une scène avec une meilleure densité d'acquisition et une plus faible erreur de mesure. C'est grâce aux informations contenues dans la source lumineuse, comme les formes projetées sur l'objet, la phase de l'onde lumineuse ou le temps que met l'onde pour faire l'aller-retour entre le capteur et l'objet, qu'il est possible d'extraire les informations 3D. Des systèmes tels que la triangulation laser [39], la lumière structurée [15], la déflectométrie [26] ou encore le temps de vol (time of flight) composent cette catégorie.

La finalité de notre système de vision est d'acquérir une image couleur et une image de profondeur associée.

L'image de couleur sera utilisée pour segmenter les objets de la scène. L'image de profondeur doit avoir une densité de mesure importante dans le respect de l'antagonisme entre des pièces de grandes dimensions (+3 mètres de côté) et des espaces entre les pièces de quelques millimètres. L'image de profondeur peut être estimée grâce aux points 3D (coordonnées X, Y et Z) de la scène en calculant leur norme car la caméra est située sur l'origine. L'image de profondeur est utilisée par la suite. Elle vise à améliorer les résultats du bloc de segmentation des objets mais aussi, pour l'application visée dans cette thèse, c'est-à-dire les autobus, à calculer les écartements entre les pièces de carrosserie.

Le système de vision retenu pour répondre à l'ensemble de ces exigences est la lumière structurée. En effet, ce type de système permet d'estimer la profondeur de petites pièces avec une faible erreur de mesure (de l'ordre de la dizaine de micromètres). Il est également peu sensible aux lumières extérieures en fonction de l'intensité de la lumière projetée sur l'objet et en fonction des algorithmes choisis pour l'analyse de l'objet. De plus, il permet d'acquérir en une seule fois une surface de largeur et hauteur supérieures à 2 mètres, et ainsi la scène peut être acquise rapidement et du même endroit.

Un état de l'art sur les méthodes de calibrage et d'estimation de la profondeur est réalisé. Ensuite, une méthode combinant deux algorithmes de lumière structurée est proposée. Elle estime en haute résolution une scène de plusieurs mètres de côté. Une analyse de cette méthode est réalisée par la suite.

2.1 Principe des systèmes de visions utilisant la lumière structurée

Les systèmes de vision utilisant la lumière structurée sont généralement composés au moins d'un projecteur et d'une caméra. Ces systèmes doivent être préalablement **calibrés** pour pouvoir les utiliser et estimer la profondeur de la scène. Le rôle du projecteur est d'afficher des motifs sur l'objet. Cela permet "d'ajouter" de l'information sur l'objet grâce aux motifs projetés (Figure 2.1).

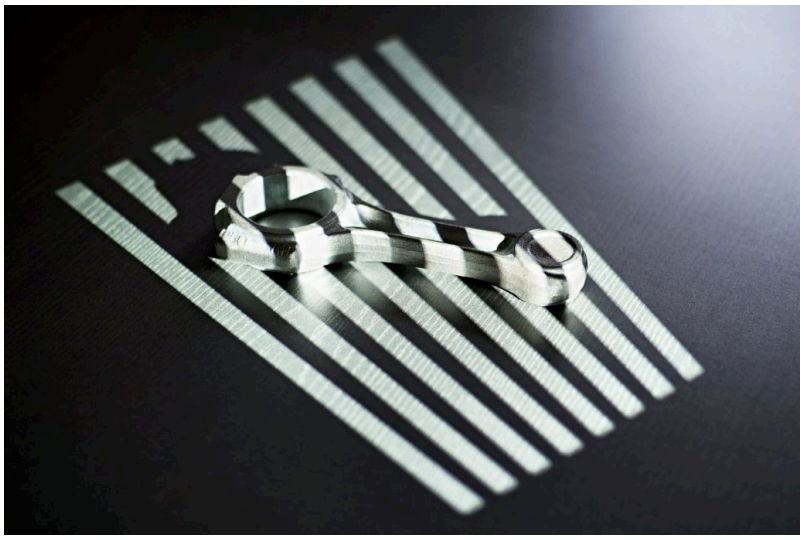


FIGURE 2.1 – Image de projection d'un motif sur un objet par un système de vision utilisant la lumière structurée

La déformation de ces motifs vue par la caméra permet de calculer la profondeur. En fonction des motifs utilisés et de la méthode d'estimation de la profondeur sélectionnée, le résultat final présentera une densité et une erreur de mesure des points 3D plus ou moins importantes.
Les algorithmes d'estimation de la profondeur peuvent être séparés en deux types : les algorithmes utilisant le **déphasage** d'une onde d'une part et ceux utilisant la **triangulation** d'autre part.

2.1.1 Calibrage

Les systèmes de vision permettant d'acquérir la profondeur d'une scène nécessitent un ou plusieurs récepteurs ainsi que des émetteurs pour les systèmes dits actifs. Pour

que le système puisse fonctionner et délivrer des informations, de surcroît avec une erreur de mesure faible, il est nécessaire de calibrer le système. Le calibrage est l'action de déterminer les paramètres d'un appareil afin d'obtenir un comportement fidèle à la réalité.

2.1.1.1 Principe de calibrage d'un système utilisant la lumière structurée

En lumière structurée, le système peut être modélisé par deux modèles sténopés, chacun composé d'une matrice intrinsèque et d'une matrice extrinsèque. Avec un seul modèle sténopé, il n'est pas possible d'estimer les coordonnées 3D. Pour cette raison, deux modèles sténopés au minimum sont nécessaires. Un des modèles sténopés représente la caméra et l'autre, le projecteur.

Un modèle sténopé peut être défini grâce à l'équation suivante :

$$\left\| \begin{matrix} u \\ v \end{matrix} \right\| = s. \left\| Intrinseque \right\| . \left\| Extrinseque \right\| . \left\| \begin{matrix} X \\ Y \\ Z \end{matrix} \right\| \tag{2.1}$$

Avec :

— s un facteur d'échelle ;

— u et v les coordonnées image, en 2D ;

— X,Y et Z les coordonnées 3D ;

Le système est modélisé par le schéma ci-dessous (Figure 2.2).

Matrice Intrinsèque

La matrice intrinsèque permet de définir les caractéristiques internes de l'émetteur ou du récepteur. Les informations importantes et utilisées sont le centre optique (généralement le centre du capteur), la focale de la lentille, ainsi que la taille des pixels. Cette matrice permet de passer d'une information 3D en 2D (pour la projection sur la caméra). La matrice intrinsèque est composée de la manière suivante :

$$Intrinseque = \left\| \begin{matrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{matrix} \right\| \tag{2.2}$$

Avec :

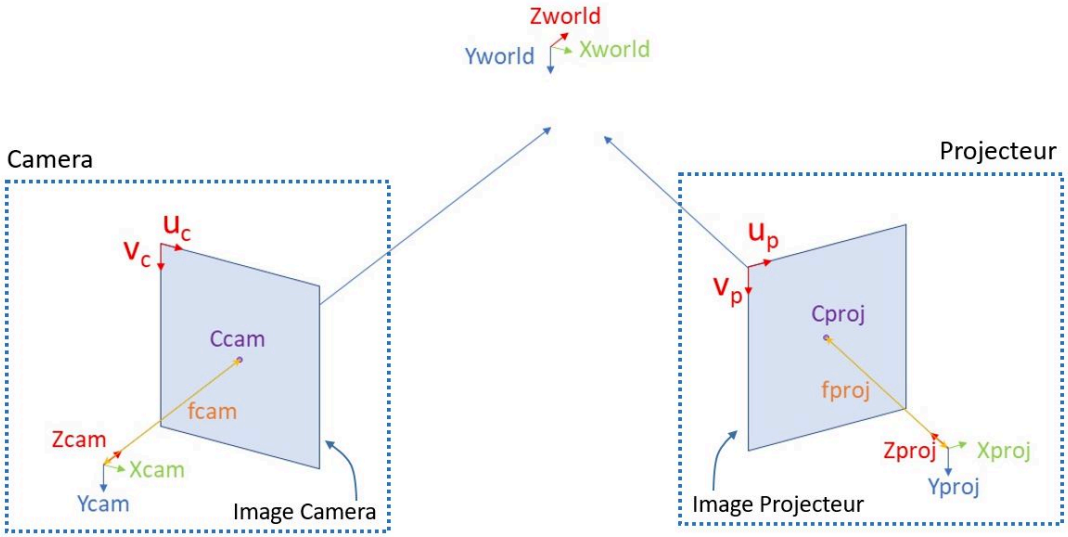


FIGURE 2.2 – Schéma du système de vision avec 2 modèles sténopés, une caméra et un projecteur

— fx et fy : la focale horizontale et verticale de la lentille en pixel ;

— cx et cy le centre optique horizontal et vertical du capteur en pixel.

Un matrice de déformation peut être estimée pour compenser les déformations induites par les lentilles de la caméra et du projecteur. Le but de cette matrice est de modifier les coordonnées images qui sont associées à un point 3D de la caméra. Cette matrice ne sera pas utilisée lors des différents tests pour des raisons de simplification.

Matrice Extrinsèque

La matrice extrinsèque permet de placer les modèles sténopés dans le repère monde défini par l'utilisateur. Elle est en fait constituée de 2 matrices, une correspondant aux mouvements de translation et une pour les mouvements de rotation entre le repère monde et le placement de la caméra ou du projecteur. La matrice de translation comporte ainsi les translations selon les axes (X, Y et Z) alors que la matrice de rotation contient les angles autour de ces 3 axes (Figure 2.3).

La matrice de rotation est une matrice 3x3 alors que celle de translation est une matrice 3x1. La matrice extrinsèque est représentée selon l'équation suivante :

$$Extrinseque = R|T = \begin{Vmatrix} R11 & R12 & R13 \\ R21 & R22 & R23 \\ R31 & R32 & R33 \end{Vmatrix} \Big| \begin{Vmatrix} Tx \\ Ty \\ Tz \end{Vmatrix} = \begin{Vmatrix} R11 & R12 & R13 & Tx \\ R21 & R22 & R23 & Ty \\ R31 & R32 & R33 & Tz \end{Vmatrix} \quad (2.3)$$

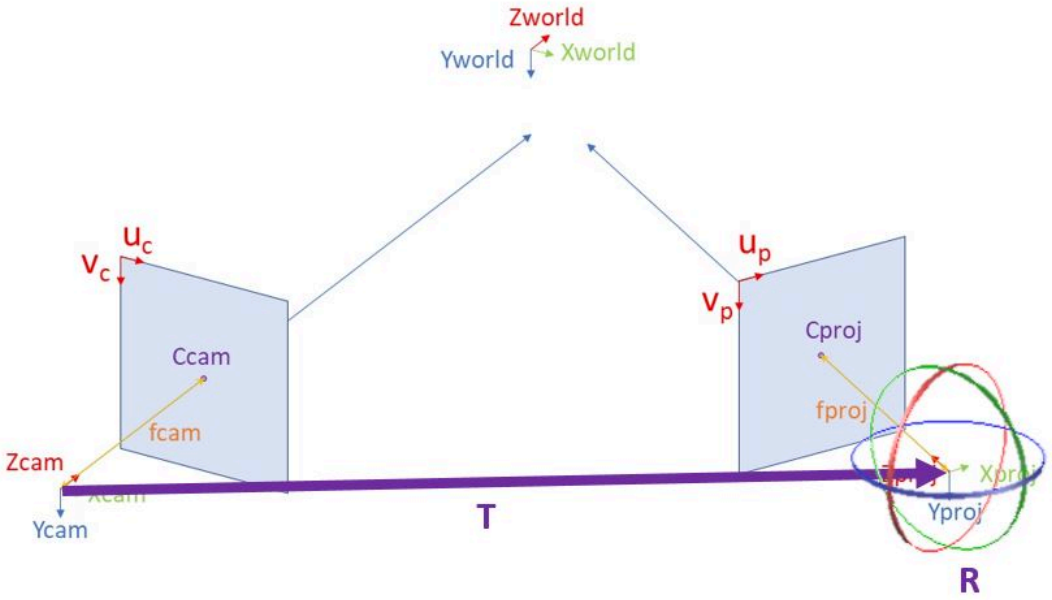


FIGURE 2.3 – Système de vision avec la représentation de la translation T et de la rotation R de la matrice extrinsèque

Avec :

— Tx,Ty et Tz les translations selon les axes X,Y et Z ;

— R11 à R33 les rotations selon les axes X,Y et Z.

Système complet

Le système de vision composé des deux modèles sténopés, de la caméra et du projecteur, peut ainsi être défini comme ci-dessous :

$$\begin{Vmatrix} u \\ v \\ 1 \end{Vmatrix} = s \begin{Vmatrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{Vmatrix} . \begin{Vmatrix} R11 & R12 & R13 & Tx \\ R21 & R22 & R23 & Ty \\ R31 & R32 & R33 & Tz \end{Vmatrix} . \begin{Vmatrix} X \\ Y \\ Z \\ 1 \end{Vmatrix} \tag{2.4}$$

— s un facteur d'échelle ;

— u et v les coordonnées image, en 2D ;

— X,Y et Z les coordonnées 3D ;

— fx et fy : la focale horizontale et verticale de la lentille en pixel ;

— cx et cy le centre optique horizontal et vertical du capteur en pixel.

— Tx,Ty et Tz les translations selon les axes X,Y et Z ;

— Tx,Ty et Tz les translations selon les axes X,Y et Z ;

— R11 à R33 les rotations selon les axes X,Y et Z .

Suite au calibrage, les matrices intrinsèques et extrinsèques sont connues. Les variables inconnues sont les facteurs s_{cam} et s_{proj}, ainsi que les coordonnées 3D du point réel considéré. Grâce aux deux modèles sténopés, 5 variables (i.e. les coordonnées et les deux facteurs) sont à déterminer avec 6 équations.

2.1.1.2 Étapes calibrage

Pour pouvoir estimer ces paramètres, c'est-à-dire les matrices intrinsèques et extrinsèques, plusieurs étapes de calibrage sont nécessaires. Tout d'abord, une mire est choisie. Plusieurs motifs peuvent être utilisés comme la mire Charucco, Arucco ou le cercle asymétrique. Le damier est toutefois la forme la plus usitée et sera utilisée pour ces travaux. La Figure 2.4 présente différentes mires utilisées dans la littérature[14] [22][4][43].

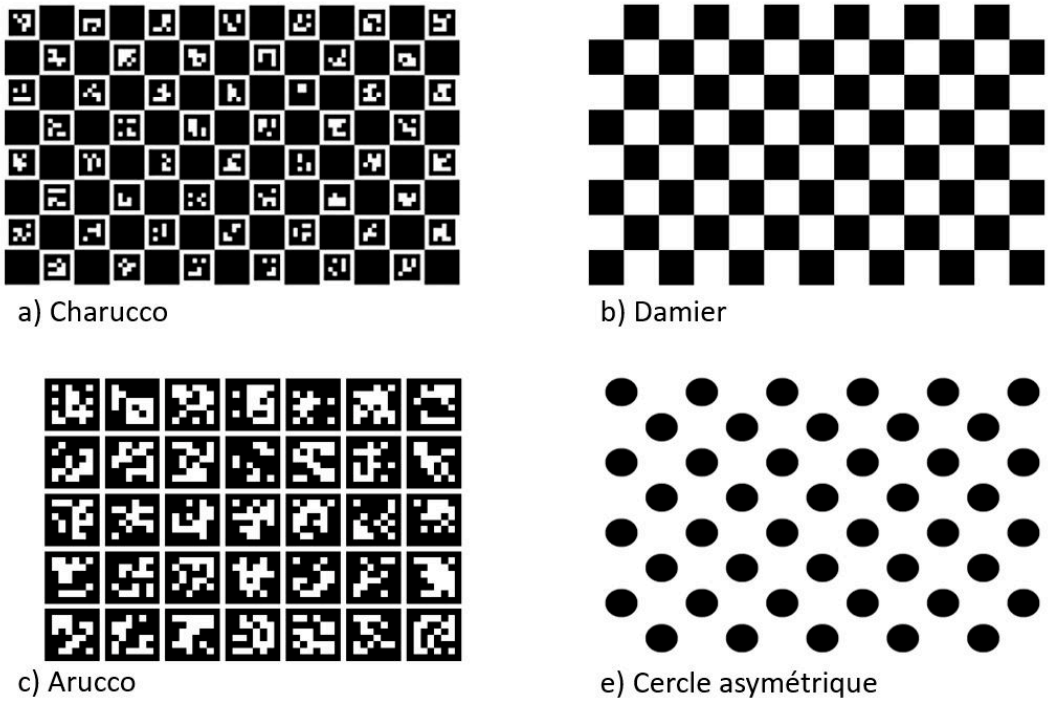


FIGURE 2.4 – Différentes mires de calibrage

2.1.1.3 Calibrage faible distance

Le calibrage pour un système comprenant une caméra et un projecteur se déroule selon le procédé suivant :

— la caméra est calibrée et permet de trouver la matrice intrinsèque de celle-ci ;

— le projecteur est calibré et permet de trouver la matrice intrinsèque de celui-ci ;

— le système complet est calibré et permet de trouver la matrice extrinsèque.

Calibrage caméra

Pour calibrer la caméra, les dimensions de la mire, c'est-à-dire la taille des carrés, ou le placement des coins ou des centres des cercles de la mire doivent être connus. Ces dimensions sont relevées en millimètres. Des images de la mire sont ensuite acquises avec la caméra. La première étape consiste à identifier, dans les images ainsi obtenues, la mire de calibrage. Cette étape nous permet de connaître, dans chaque photo, les points 2D de la mire (Figure 2.6).

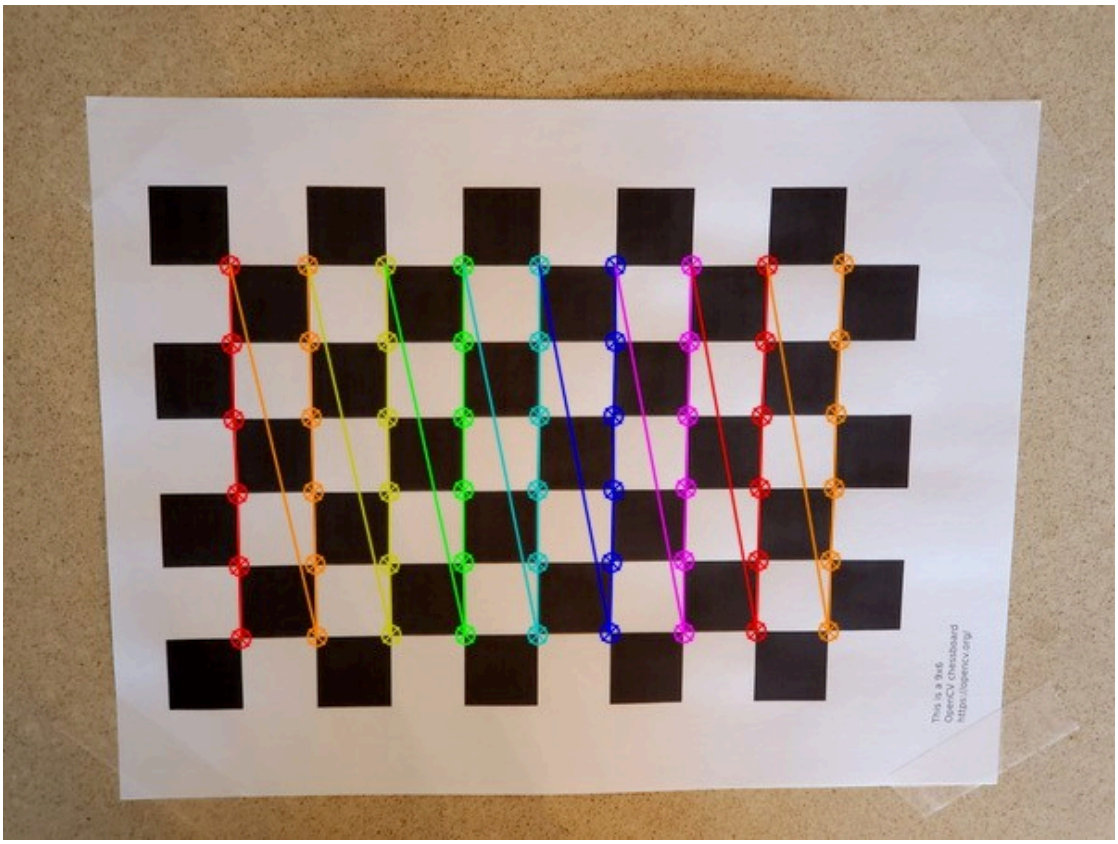


FIGURE 2.5 – Reconnaissance de la mire de calibrage

La deuxième étape consiste à positionner les points 3D de la grille dans le repère monde. Pour ce faire, le repère monde est défini sur celle-ci, à l'un des angles de la

mire avec l'axe X et Y allant dans la même direction que l'axe X et Y de l'image de la caméra, c'est à dire l'axe X allant vers la droite et Y vers le bas lorsque l'on se place du point de vue de la caméra. La position des autres angles de la mire est donc connue : ils sont décalés du centre du repère monde de la taille et du nombre des carrés de la mire qui sont connus. La mire étant plane, la profondeur est initialisée à 0.

La dernière étape permet d'estimer les paramètres intrinsèques de la caméra ainsi que ses paramètres extrinsèques par rapport au repère monde, notamment grâce à des fonctions de la librairie OpenCV. Ces fonctions utilisent notamment l'algorithme Levenberg-Marquardt[23] qui permet de trouver une solution numérique minimale à un problème.

Calibrage projecteur

Le calibrage du projecteur repose sur le même principe que celui de la caméra.

La principale modification repose sur le fait que les points 3D de la matrice de calibrage réelle sont remplacés par les points 3D de la matrice de calibrage projetée par le projecteur sur le même plan que la mire réelle. Pour remonter aux points 3D de la mire projetée, les matrices extrinsèques de la caméra sont utilisées.

Le plan où la mire est projetée bouge alors que le projecteur est immobile. Les points 3D changent donc constamment. Ce n'est toutefois pas le cas des points 2D de la mire car c'est l'image de la mire projetée qui est prise comme référence. L'estimation des paramètres intrinsèques du projecteur est réalisée grâce aux points 3D estimés et aux points 2D de la mire projetée.

Calibrage du système

Le calibrage du système reprend les matrices intrinsèques de la caméra et du projecteur ainsi que les images de la mire projetée pour estimer la matrice extrinsèque du système. Celle-ci n'est pas basée sur le repère monde mais sur un repère sélectionné, généralement celui de la caméra. La matrice extrinsèque du système, c'est-à-dire les translations (T) et rotations (R) entre le projecteur et la caméra, est donc unique pour le système considéré dès lors que celui-ci constitue un ensemble fixe. Elle ne change pas lorsque le système en entier bouge ou qu'il est mis en place sur la zone à acquérir.

2.1.1.4 Calibrage longue distance

Dans notre problématique d'acquisition d'objets lointains, la méthode de calibrage doit être adaptée. En effet, la méthodologie présentée ci-dessus comporte plusieurs difficultés. La principale difficulté est celle du paramétrage de la caméra. En effet, pour le réaliser, il faut disposer d'une mire de grandes dimensions et positionnée à une distance appropriée pour le système. Si la mire est de petite taille, la caméra, du fait de la distance, verra la mire avec peu de pixel et ne pourra pas avoir un calibrage satisfaisant. De plus, si la planéité de la mire n'est pas garantie en raison d'une déformation, même légère, le calibrage n'est pas fiable.
Un calibrage à longue distance en trois étapes est proposé [1] : 2 pour configurer les matrices intrinsèques de la caméra et du projecteur et 1 étape pour la matrice extrinsèque. Toutefois, la première étape porte cette fois sur le calibrage du projecteur.
Le projecteur est focalisé sur l'objet au loin, généralement un mur, alors que la caméra est focalisée sur la mire qui est placée à faible distance. Le fait que le projecteur ne soit pas focalisé sur la mire rend les motifs flous. Des franges binaires sont projetées qui sont perçues par la caméra comme des sinusoïdes. Celles-ci permettent d'extraire une phase qui pourra ensuite être exploitée.
Cette phase est unique et permettra ainsi de relier les points caméra et projecteur. Cependant, cela n'est pas suffisant pour obtenir les coordonnées des points 3D. Les mires en franges binaires sont ensuite projetées sur une mire réelle composée de cercles. L'espacement entre les différents cercles étant connu, les points 3D de la mire sont donc également connus. Il est alors possible de faire le lien entre la position 3D des centres des cercles de la mire et la position 2D des points de la mire projetée (Figure 2.6).

La deuxième étape calibre la caméra avec la matrice réelle. La caméra est focalisée sur le mur au loin. Étant donné qu'il est possible d'estimer le centre d'un cercle même si ces bords sont flous, il est alors possible de faire le lien entre les points 2D, les centres des cercles dans les images de la mire prise par la caméra, et les points 3D des centres de la mire réelle. La matrice intrinsèque de la caméra est ainsi estimée.

La dernière étape utilise les matrices intrinsèques de la caméra et du projecteur, ainsi que des coordonnées données par une caméra de type Kinect pour estimer la matrice extrinsèque du système. Une nouvelle salve d'acquisitions est réalisée pour établir le lien entre les points 2D de l'image de la mire projetée, les points 2D de la mire projetée vue par la caméra et les points 3D estimés grâce à la caméra Kinect.

La méthode la plus récente [24] utilise la méthode précédente en proposant d'ajouter un algorithme permettant de compenser les erreurs résiduelles issues des lentilles

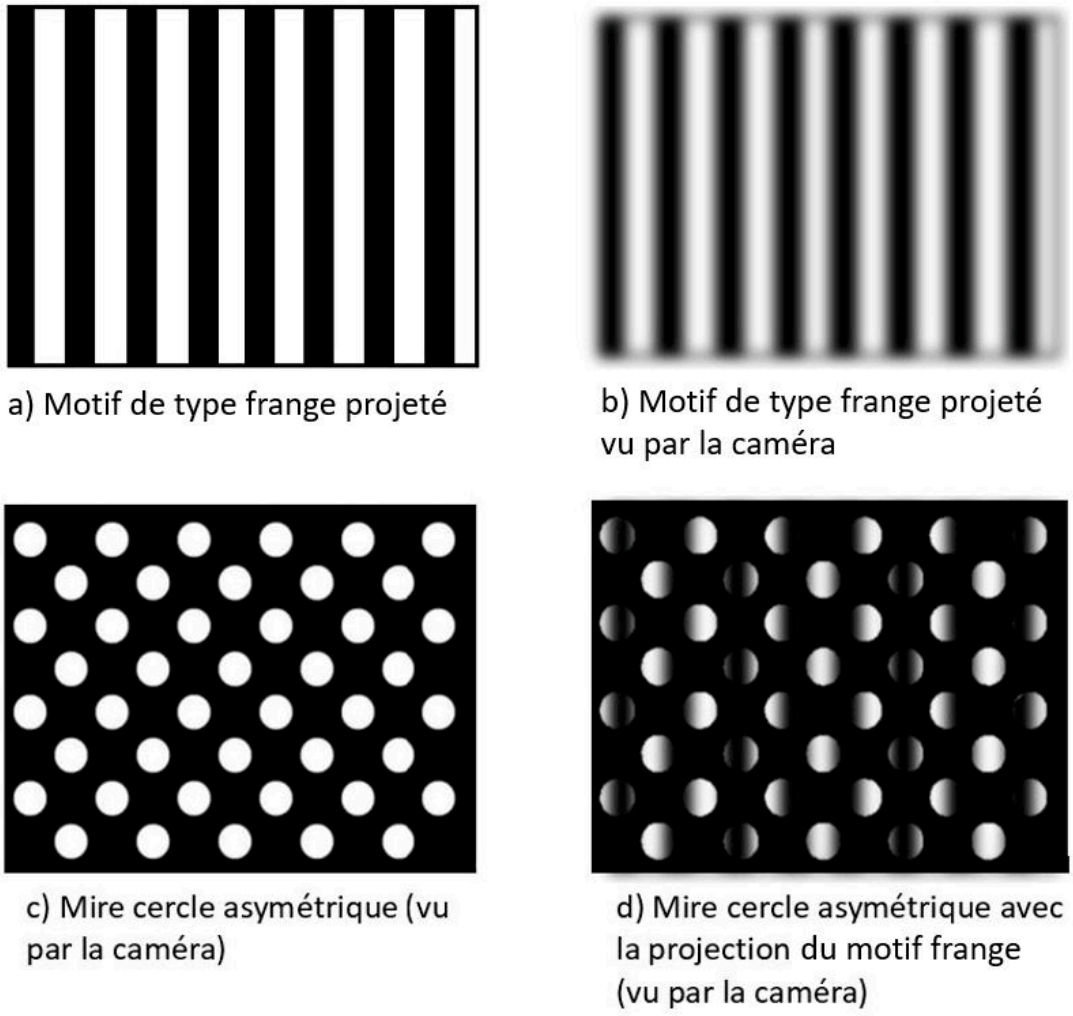


FIGURE 2.6 – Mélange du motif type frange (binaire) projeté et de la mire réelle, vu par la caméra

commerciales. En effet, ces lentilles ne sont pas parfaites et ne suivent donc pas exactement le modèle sténopé et déforment les images perçues par la caméra [27]. Pour cela, il prend en entrée les matrices intrinsèques et la matrice extrinsèque du système. Pour corriger les erreurs, il utilise tous les points 2D dans les images de la caméra montrant la mire projetée mais aussi le fait que la mire est plane. Cette méthode permet d'obtenir un résultat dont l'erreur est en-dessous du millimètre pour une mesure contenue dans un volume de 1200 (Hauteur) × 800 (Largeur) × 1000 (Profondeur) mm^3) [24].

Le calibrage permet donc d'avoir les informations du système qui seront utilisés pour estimer la profondeur des objets de la scène. L'estimation de la profondeur peut être réalisée grâce à deux méthodes différentes : le décalage de phase et la triangulation.

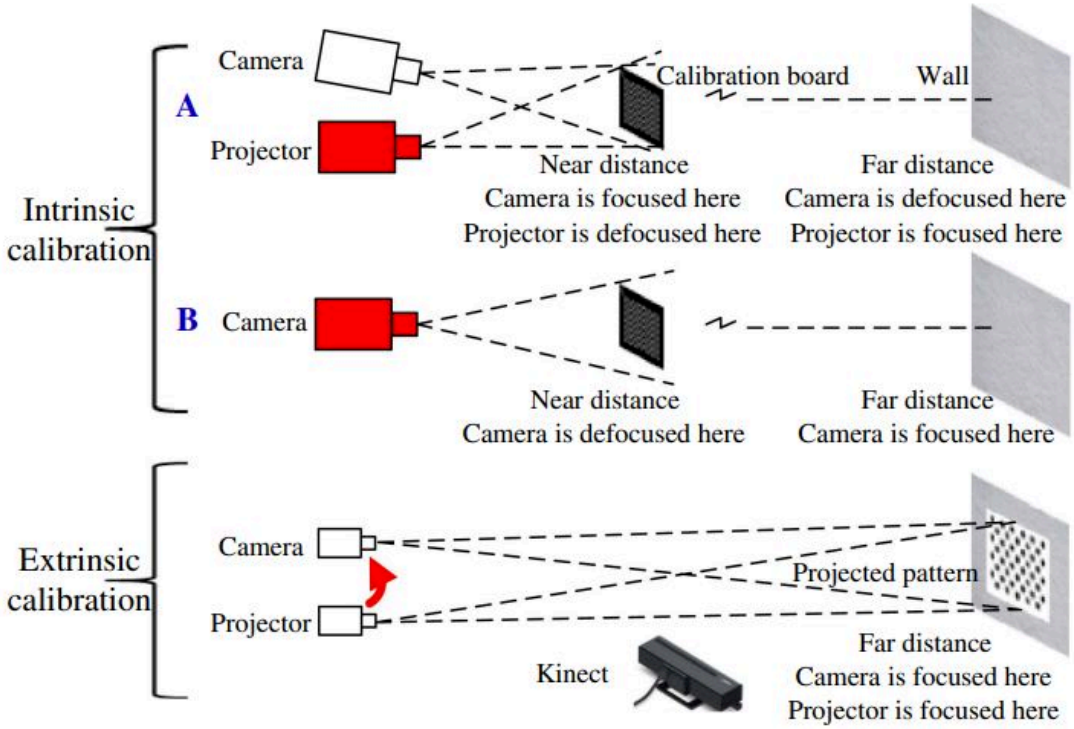


FIGURE 2.7 – Étapes du calibrage à longue distance [1]

2.1.2 Décalage de phase

Les méthodes utilisant le décalage de phase sont utilisées dans de nombreux contextes comme l'analyse de produits industriels [37] en mécanique, en médecine [2] et en aérospatiale [29], ou encore en agriculture [31]. Ces méthodes utilisent des motifs contenant des franges, pour la grande majorité, continues. Le principe de ces méthodes est que les motifs projetés sont codés comme des signaux (sinusoïdes pour les plus connus). Il est possible d'extraire des informations de ces signaux tels que leurs phases. Les deux méthodes les plus courantes sont la profilométrie par transformée de Fourier (FTP) [38] et la profilométrie par déphasage (PSP) [45].

La profilométrie par transformée de Fourier (FTP) utilise un seul motif contenant des franges à hautes fréquences. La phase est extraite en utilisant un filtre passe-bande dans le domaine fréquentiel. Cette méthode est très utilisée et efficace pour l'analyse de formes 3D en mouvement. Il convient également de noter que la méthode utilisant la transformée d'ondelettes est la méthode permettant d'obtenir la profondeur de la scène avec une erreur de mesure la plus faible, par rapport à celles développées avec ce principe [42].

Contrairement à la méthode précédente, la profilométrie par déphasage (PSP) utilise

généralement plusieurs motifs, au moins 3, pour extraire la phase. Elle est utilisée pour analyser principalement des objets statiques. En comparaison avec la profilométrie par transformée de Fourier, la profilométrie par déphasage est généralement plus robuste à l'éclairage ambiant et aux réflexions des objets. Elle permet d'acquérir avec une haute densité les points de mesure, c'est à dire en chaque pixel de l'image analysé, et a une erreur de mesure inférieure aux méthodes de profilométrie par transformée de Fourier. Son temps de traitement est cependant plus long. Pour l'ensemble de ces raisons, elle est étudiée dans les paragraphes suivants.

2.1.2.1 Principe

La profilométrie par déphasage utilise des informations temporelles pour mesurer la hauteur relative d'un objet, c'est-à-dire la différence de hauteur entre un objet et une référence. Elle comporte plusieurs étapes :

1. la construction des motifs ;
2. l'acquisition des motifs projetés ;
3. la transformation des images en phase enroulée ;
4. la transformation de la phase enroulée en phase déroulée ;
5. la différence de phase déroulée ;
6. l'estimation de la profondeur.

Construction des motifs

Pour pouvoir créer les motifs qui seront projetés sur l'objet, il est nécessaire de connaître la fréquence et le déphasage les plus appropriés pour une acquisition avec une faible erreur de mesure des objets de la scène. La fréquence correspond au nombre de périodes visibles dans l'image. Le déphasage est le nombre de motifs projetés nécessaires pour calculer la phase de l'image à analyser. Dans chaque motif, le signal est décalé de la valeur du déphasage multiplié par π et par le numéro du motif moins un. Par exemple, pour un déphasage de 3, 3 motifs seront projetés. Le premier motif aura un déphasage de 0, le deuxième de $\pi/3$ et le troisième de $2\ \pi/3$.

Ces paramètres sont à régler en fonction des résolutions de la caméra et du projecteur, du placement du projecteur par rapport à la caméra ainsi que du placement du système par rapport aux objets à analyser. Plus les résolutions de la caméra et du projecteur sont importantes, plus la fréquence peut être élevée. Si l'une des deux résolutions

est faible alors la fréquence maximale baisse (Figure 2.8). Le déphasage n'est pas impacté de la même manière. Plus le déphasage est important, moins la variance de la mesure sera grande. Une fois ces deux paramètres choisis, les motifs peuvent être créés en appliquant les équations suivantes :

$$I^n_{Vert}(x,y) = A + B\cos(2\pi fx - 2\pi n/N) \tag{2.5}$$

$$I^n_{Hor}(x,y) = A + B\cos(2\pi fy - 2\pi n/N) \tag{2.6}$$

Avec :

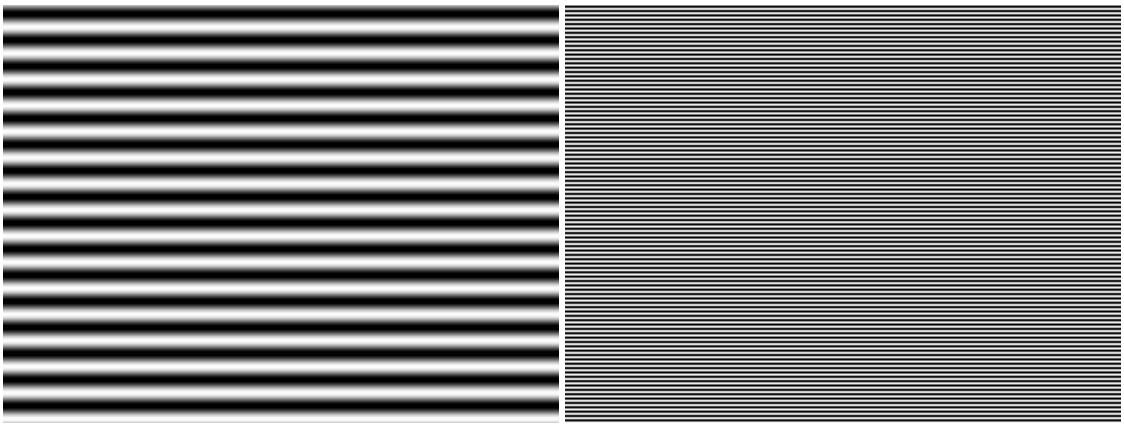
— A : constante permettant d'avoir des images comprises entre 0 et 1 (généralement mis à 0.5) ;

— B : constante représentant l'amplitude des sinusoïdes (généralement mis à 0.5 pour une amplitude entre min et max de 1) ;

— f : la fréquence ;

— n : l'index du déphasage (0, 1, 2, ..., N-1) ;

— N : nombre de déphasages et nombre total d'images à acquérir.

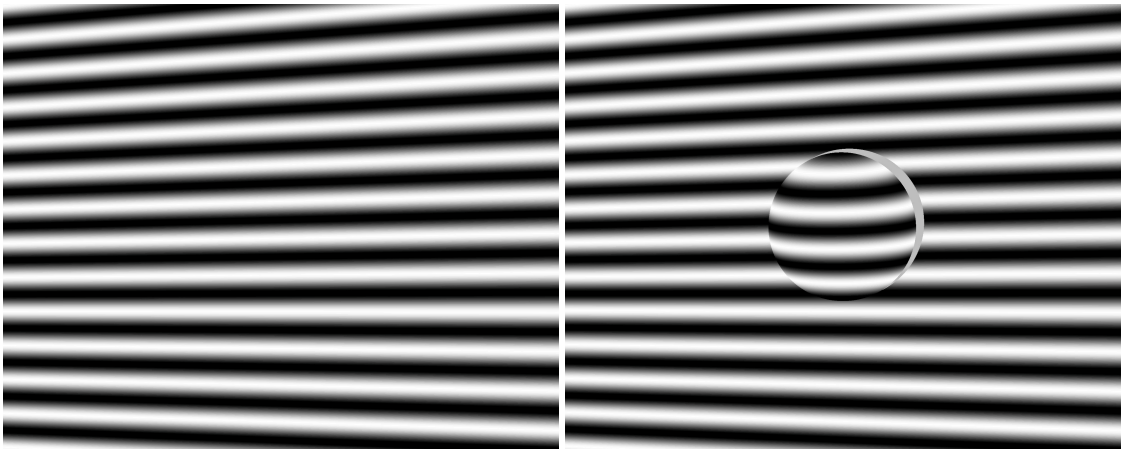


(a) Motif de fréquence 16 (b) Motif de fréquence 96

FIGURE 2.8 – Motifs projetés

Acquisition des motifs projetés

La projection des motifs est réalisée sur un plan de référence puis sur ce même plan devant lequel l'objet a été intercalé. Pour toutes les étapes suivantes, les transformations et les calculs sont réalisés séparément sur les images du plan de référence et sur les images du plan de référence avec l'objet intercalé (Figure 2.9). Par la suite, le terme "scène" désigne le plan de référence avec l'objet intercalé devant lui.



(a) Image de la projection de la mire de fréquence 16 sur le plan de référence

(b) Image de la projection de la mire de fréquence 16 sur la scène

FIGURE 2.9 – Images des motifs projetés sur le plan de référence et la scène

Phase enroulée

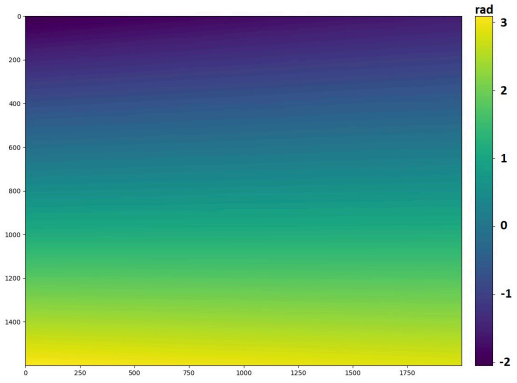
La troisième étape consiste à transformer les données d'intensité des pixels des images en information de phase du signal projeté. La méthode d'acquisition de la phase est spécifique au type de frange du motif projeté ainsi qu'au nombre de motifs utilisés. La méthode la plus simple utilise 3 motifs constitués de sinusoïdes : la première est déphasée de $-\pi/3$, la deuxième est non déphasée et la troisième est déphasée de $\pi/3$. En utilisant ces motifs, l'équation permettant la conversion des informations des images en phases est la suivante :

$$\phi(x,y) = \arctan\left(\frac{\sum_{n=0}^{N-1} I_n(x,y)\sin(\frac{2\pi n}{N})}{\sum_{n=0}^{N-1} I_n(x,y)\cos(\frac{2\pi n}{N})}\right) \tag{2.7}$$

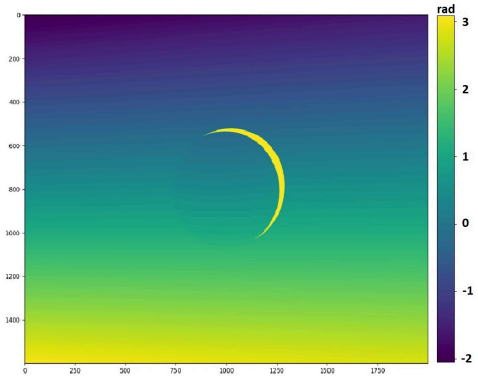
Cette phase est comprise entre $-\pi$ et π. Elle est discontinue. La Figure 2.10 montre la phase de la fréquence 1 et 96 sur un plan de référence et une scène contenant une sphère et un plan incliné.

Phase déroulée

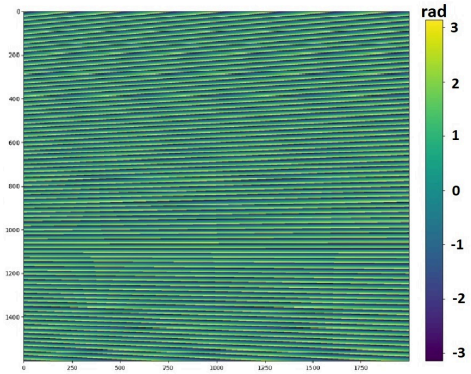
La quatrième étape, dite du déroulement de phase, permet de passer d'un signal discontinu à un signal continu (Figure 2.11). À chaque changement de période, la valeur 2π est ajoutée par rapport aux phases des périodes précédents. Les solutions proposées pour dérouler la phase peuvent être classées en 2 groupes : traitement dans le domaine temporel ou dans le domaine spatial.



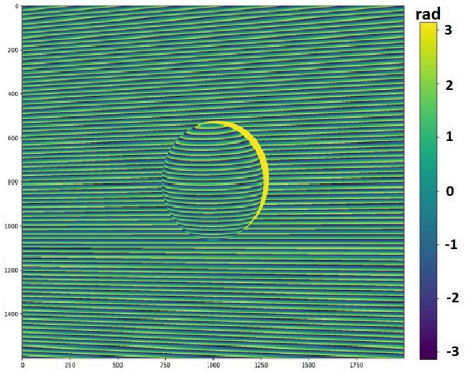
(a) Image du plan de référence avec la phase de fréquence 1



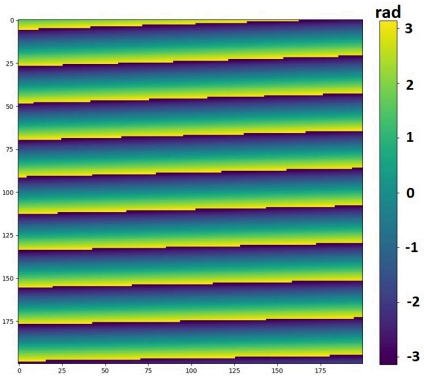
(b) Image de la scène avec la phase de fréquence 1



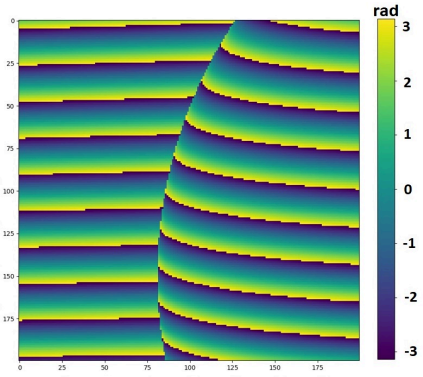
(c) Image du plan de référence avec la phase de fréquence 96



(d) Image de la scène avec la phase de fréquence 96



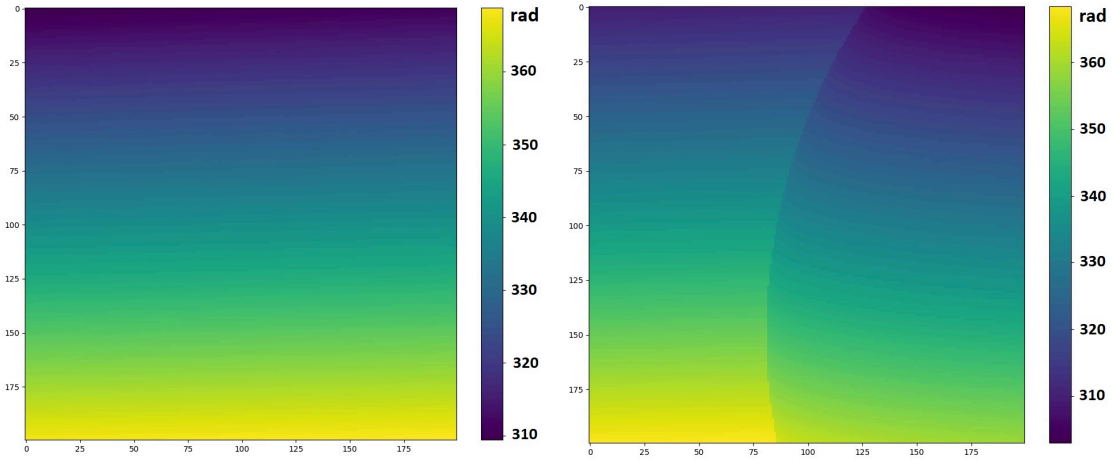
(e) Agrandissement du plan de référence avec la phase de fréquence 96



(f) Agrandissement de la scène avec la phase de fréquence 96

FIGURE 2.10 – Images du plan de référence et de la scène à différents instants de la méthode d'estimation de la profondeur ;
(a) et (b) pour la phase issue des images de fréquence 1, (c) et (d) la phase issue des images de fréquence 96 et (e) et (f) agrandissements pour la phase de fréquence 96

Les traitements dans le domaine spatial prennent en considération le fait que la phase est continue. La phase déroulée est donc calculée, par itérations, en analysant le voisinage des pixels de la phase enroulée. Un grand nombre de méthodes sont proposées



(a) Agrandissement de l'image de la phase déroulée du plan de référence

(b) Agrandissement de l'image de la phase déroulée de la scène

FIGURE 2.11 – Images des phases déroulées du plan de référence et de la scène

comme la méthode de Goldstein[10], de Flynn [40] ou encore minimum Lp-norm method [9]. Cependant elles sont peu robustes aux forts changements de contraste et notamment aux discontinuités présentes dans l'image (bord du plan, objet pointu...).

Les traitements dans le domaine temporel n'utilisent pas seulement l'image de phase enroulée (signal compris entre $-\pi$ et π) mais aussi d'autres informations pour connaître l'ordre des franges. Cela se traduit par l'ajout d'une autre image de phase enroulée, d'un motif particulier dans les motifs projetés ou d'une image encodée grâce à des motifs binaires. Le fait de pouvoir traiter chaque pixel de façon indépendante par rapport à son voisin spatial permet de pouvoir analyser des objets ayant des discontinuités importantes et de ne pas répercuter l'erreur due à un pixel erroné sur ces voisins, ce qui pourrait impacter fortement le déroulement de la carte de phase. La problématique rencontrée précédemment est donc levée et il est possible de calculer la phase absolue du signal en chaque pixel. Ces méthodes seront étudiées dans la partie suivante.

Différence de phase

La phase déroulée du plan de référence avec l'objet est soustraite à la phase déroulée du plan de référence seul. Les pixels de la caméra qui aperçoivent le plan de référence dans les deux images ont la même phase, la différence en ces points est donc égale à 0. Par contre, dans la zone où est placé l'objet, la différence est proportionnelle à la hauteur qui doit être calculée (i.e. celle de l'objet).

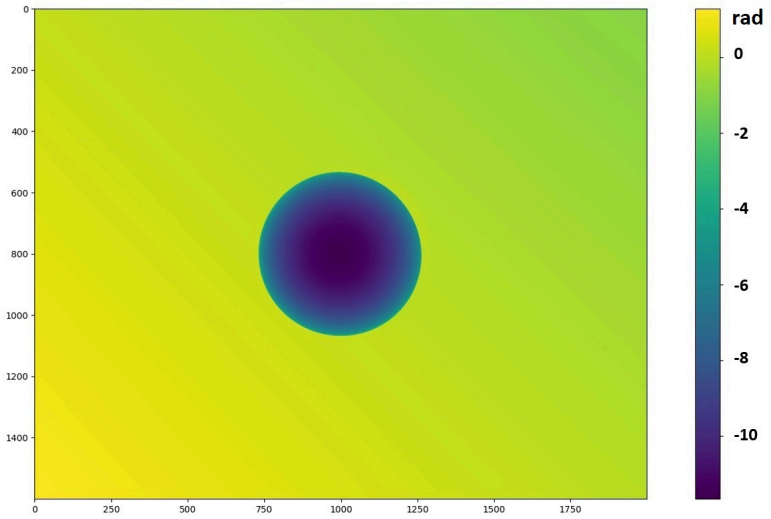


FIGURE 2.12 – Image de la différence de phase entre le plan de référence et la scène

Estimation de la profondeur

La dernière étape est donc le calcul de la hauteur de l'objet en fonction de la différence de phase calculée précédemment (Figure 2.12).

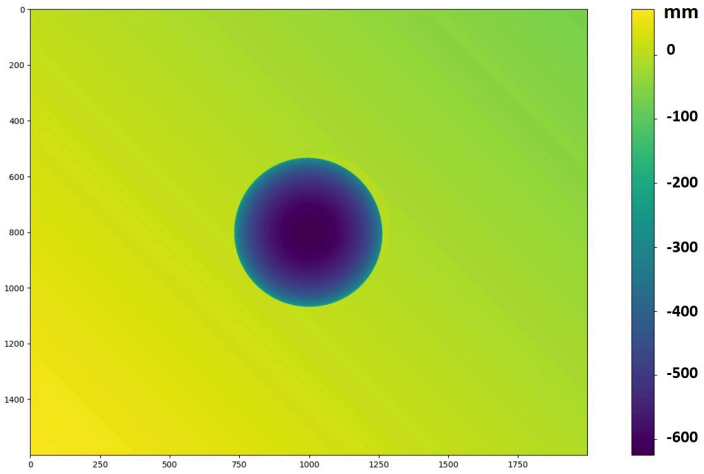


FIGURE 2.13 – Image de la profondeur de la scène (vérité terrain)

Pour pouvoir calculer la profondeur (Figure 2.13), des approximations sont faites. La première est que la surface de l'objet est constamment considérée comme perpendiculaire à l'axe caméra-projecteur. Grâce à cela, il est possible d'utiliser le théorème de Thalès (Figure 2.14).

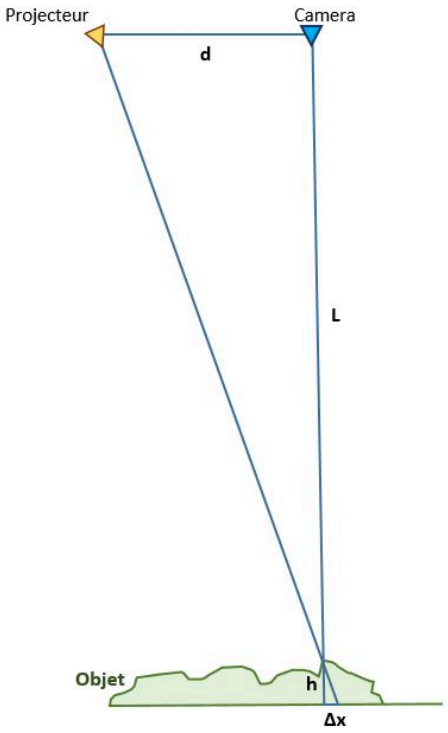


FIGURE 2.14 – Schéma du décalage de phase

Avec :

— d : la distance entre le projecteur et la caméra ;

— L : la distance entre la caméra et le plan de Référence ;

— h : la hauteur de l'objet ;

— Δx : le décalage entre un point de l'objet et un point sur le plan de référence qui sont les projections d'un même pixel du projecteur.

Le théorème de Thalès permet d'écrire la relation suivante :

$$h = \frac{\Delta x \times L}{d} \tag{2.8}$$

La deuxième approximation est que la longueur d'onde prise sur le plan de référence est la même que celle sur l'objet, quelle que soit sa hauteur.

Étant donné que la longueur d'onde (λ) est la distance d'une période (soit 2 π), il est alors possible de faire le lien entre Δx et $\Delta\phi$, la différence de phase.

$$\Delta x = \frac{\Delta\phi \times \lambda}{2\pi} \tag{2.9}$$

donc :

$$h = \frac{\Delta\phi \times \lambda \times L}{d \times 2\pi} \tag{2.10}$$

2.1.2.2 Algorithmes d'estimation de la phase absolue

La phase absolue est la phase déroulée et unique du signal. En chaque pixel de coordonnée (x,y), elle peut être modélisée par l'équation suivante :

$$\Phi(x,y) = \phi(x,y) + 2\pi k(x,y) \tag{2.11}$$

Avec :

— Φ : la phase déroulée absolue ;

— ϕ : la phase enroulée ;

— k : le numéro de frange.

Les méthodes d'estimation de la phase absolue sont divisées en 3 grands groupes. Le premier estime le numéro des franges grâce à des motifs supplémentaires projetés sur le plan de référence et l'objet, le deuxième grâce aux motifs initiaux modifiés et codés avec plusieurs fréquences et le dernier grâce aux motifs initiaux modifiés et codés avec un motif de Speckle.

D'après la littérature [45], deux méthodes se démarquent et sont particulièrement bien adaptées à la problématique du bus : l'algorithmes Bi-frequency (3+2)et Dual-frequency .

Algorithme Bi-frequency (3+2)

La méthode Bi-frequency (3+2) [44] utilise des motifs sup-plémentaires pour estimer la phase absolue. Ainsi, deux types de motifs sont mis en oeuvre : d'une part les motifs originaux de décalage de phase avec une fréquence importante (au moins 16 périodes sont visibles dans les motifs projetés), et d'autre part les motifs permettant l'estimation des numéros des franges. Ceux-ci sont composés d'au moins 3 motifs (Figure 2.15) :

— 2 motifs de fréquence égale à 1 et donc de déphasage $\pi/2$;

— 1 motif (nommé A) obtenu en moyennant les motifs de hautes fréquences .

L'équation permettant d'estimer la phase basse fréquence est la suivante :

$$\phi_b = \arctan \frac{I_b^0 - A}{I_b^1 - A} \tag{2.12}$$

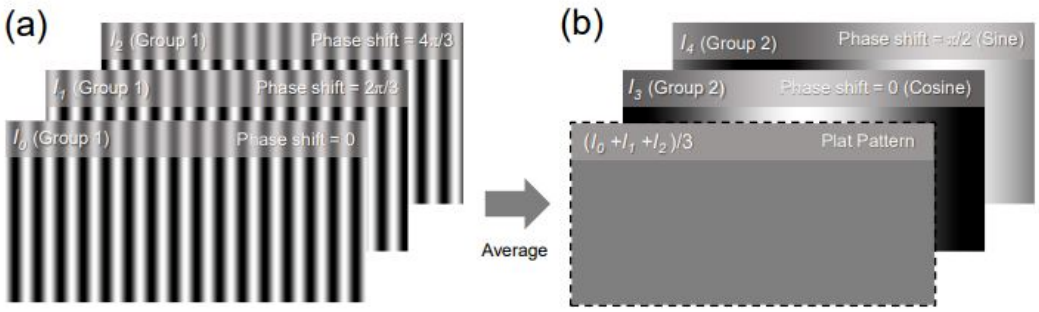


FIGURE 2.15 – Motifs nécessaires pour le fonctionnement de la méthode Bi-frequency. (a) les motifs haute fréquence et (b) les motifs avec des fréquences basses.

Avec :

— ϕ_b : phase estimée basse fréquence ;

— I_b^0 : la première image basse fréquence ;

— I_b^1 : la deuxième image basse fréquence décalée de $\pi/2$;

— A : la moyenne des motifs haute fréquence.

Cette phase, ϕ_b, est utilisée pour calculer en chaque pixel le numéro de la frange à laquelle il appartient. La Figure 2.16 montre les images des franges pour le plan de référence et la scène. L'équation permettant de passer des phases au numéro de la frange en chaque pixel est la suivante :

$$\kappa(x,y) = PartieEntiere(\left[\frac{\lambda_b/\lambda_h\phi_b(x,y) - \phi_h(x,y)}{2\pi}\right]) \tag{2.13}$$

Avec (en chaque pixel) :

— k : numéro de frange ;

— λ_b : valeur de la basse fréquence ;

— λ_h : valeur de la haute fréquence ;

— ϕ_b : phase estimée basse fréquence ;

— ϕ_h : phase estimée haute fréquence.

Cette méthode présente les avantages :

— d'être très robuste aux problèmes de déroulement de phase ;

— d'avoir une faible erreur de mesures ;

— d'être peu coûteuse en termes de calcul car elle nécessite 6 images au minimum ;

— d'être efficace contre les problèmes de focalisation de la lentille qui peuvent intervenir lorsque l'objet sort de la zone où l'image est nette.

Algorithme Dual-frequency

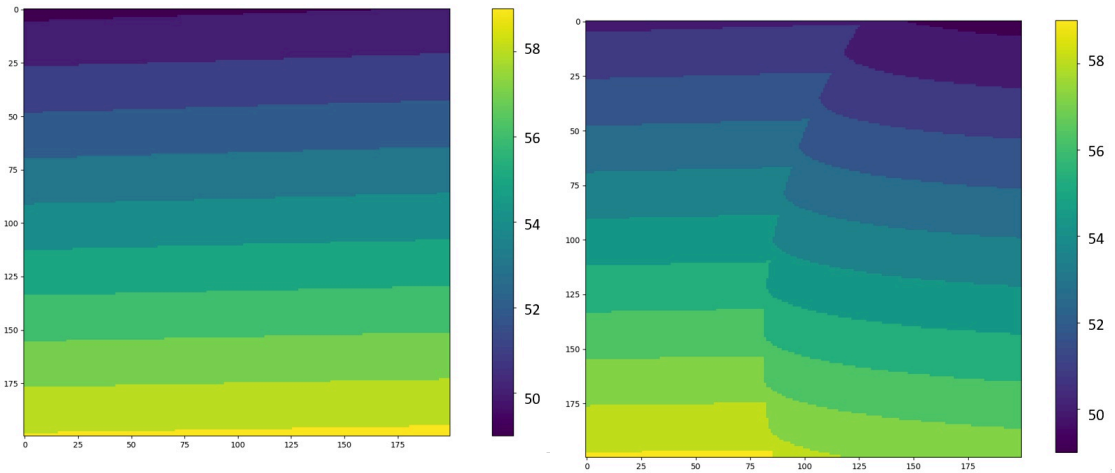
Contrairement à la méthode précédente, la méthode Dual-frequency n'utilise pas de motifs supplémentaires [21]. Elle utilise par contre deux types de fréquences pour estimer les numéros des franges : une fréquence importante (au moins 16 périodes visibles dans les motifs projetés) et une fréquence faible. Pour cela, les motifs ne sont pas codés avec une seule fréquence mais avec les deux types de fréquences en même temps. Comme pour les télécommunications, la fréquence basse est utilisée comme onde porteuse pour la haute fréquence. Pour estimer la phase de la scène, il est nécessaire d'avoir au minimum 5 motifs (Figure 2.17).

$$I^n_{Vert}(x,y) = A + B_h \cos(2\pi f_h x - 2\pi n/N) + B_b \cos(2\pi f_b x - 4\pi n/N) \tag{2.14}$$

$$I^n_{Hor}(x,y) = A + B_h \cos(2\pi f_h y - 2\pi n/N) + B_b \cos(2\pi f_b y - 4\pi n/N) \tag{2.15}$$

Avec :

— B_h : constante représentant l'amplitude des sinusoïdes haute fréquence (généralement



(a) Image des franges du plan de référence zoomée

(b) Image des franges de la scène zoomée

FIGURE 2.16 – Images des franges issues de l'image de phase de fréquence 1 et 96

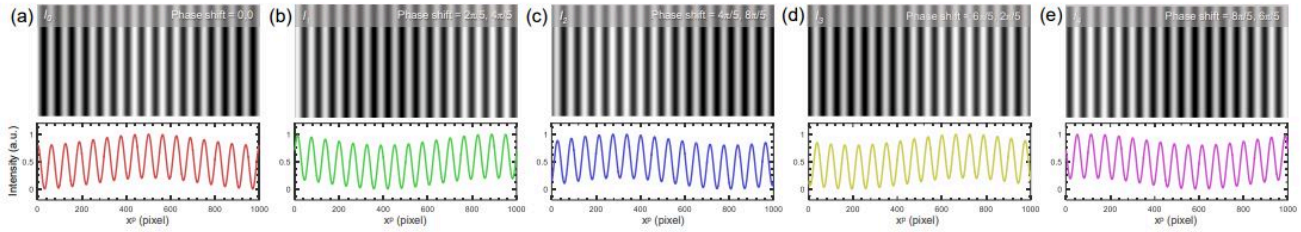


FIGURE 2.17 – Motifs nécessaires pour le fonctionnement de la méthode Dual-frequency. Les motifs sont composés d'une fréquence haute et d'une fréquence porteuse de basse fréquence.

mis à 0,4 pour une amplitude de min à max de 1) ;

— B_b : constante représentant l'amplitude des sinusoïdes basse fréquence (généralement mis à 0,1 pour une amplitude de min à max de 1) ;

— A : constante permettant d'avoir des images comprises entre 0 et 1 (généralement mis à 0,5) ;

— f_h : la fréquence haute ;

— f_b : la fréquence basse ;

— n : l'index du déphasage (0, 1, 2, ..., N-1) ;

— N : nombre de déphasages et nombre total d'images à acquérir.

Les amplitudes B_h et B_b doivent être réglées pour pouvoir à la fois avoir une faible erreur de mesure et être robustes au déroulement de phase. La somme des amplitudes B_h et B_b doit être aussi égale à 0,5 pour que l'amplitude finale des motifs soit comprise entre 0 et 1. Plus l'amplitude de B_b est importante, moins les erreurs de déroulement de phase sont importantes. Plus l'amplitude de B_h est importante, moins les erreurs de mesure des phases de l'image sont importantes. Un compromis est donc à trouver pour pouvoir allier une faible erreur de déroulement de phase et d'estimation des phases. L'amplitude B_b est généralement égale à 0,1 alors que l'amplitude B_h est égale à 0,4 [45].

L'extraction de la phase issue des motifs à haute fréquence est la même que dans le principe général. L'équation permettant d'estimer la phase des motifs à basse fréquence est la suivante :

$$\phi_b(x,y) = \arctan\left(\frac{\sum_{n=0}^{N-1} I_n(x,y)\sin\left(\frac{4\pi n}{N}\right)}{\sum_{n=0}^{N-1} I_n(x,y)\cos\left(\frac{4\pi n}{N}\right)}\right) \tag{2.16}$$

Cette phase est comprise entre $-\pi$ et π.

Comme pour la méthode précédente, l'équation 2.13 permet de trouver en chaque pixel le numéro de frange associé grâce aux phases basses et hautes fréquences. Le numéro de frange peut ensuite permettre de calculer la phase absolue.

Cette méthode a l'avantage :

— d'être robuste aux problèmes de déroulement de phase ;

— d'avoir une faible erreur de mesure ;

— d'être peu coûteuse en termes de calcul car elle ne comporte que 5 images au minimum ;

— et d'être efficace contre les problèmes de focalisation de la lentille qui peuvent arriver lorsque l'objet sort de la zone où l'image est nette.

2.1.3 Lancer de rayons

2.1.3.1 Principe

Le lancer de rayons est utilisé dans de nombreuses applications comme l'aéronautique[7], les jeux vidéos ou encore le rendu graphique [20].

Pour le lancer de rayons, deux rayons issus de deux sources connues sont lancés. Leur intersection permet d'obtenir la distance entre cette intersection et le référentiel défini auparavant.

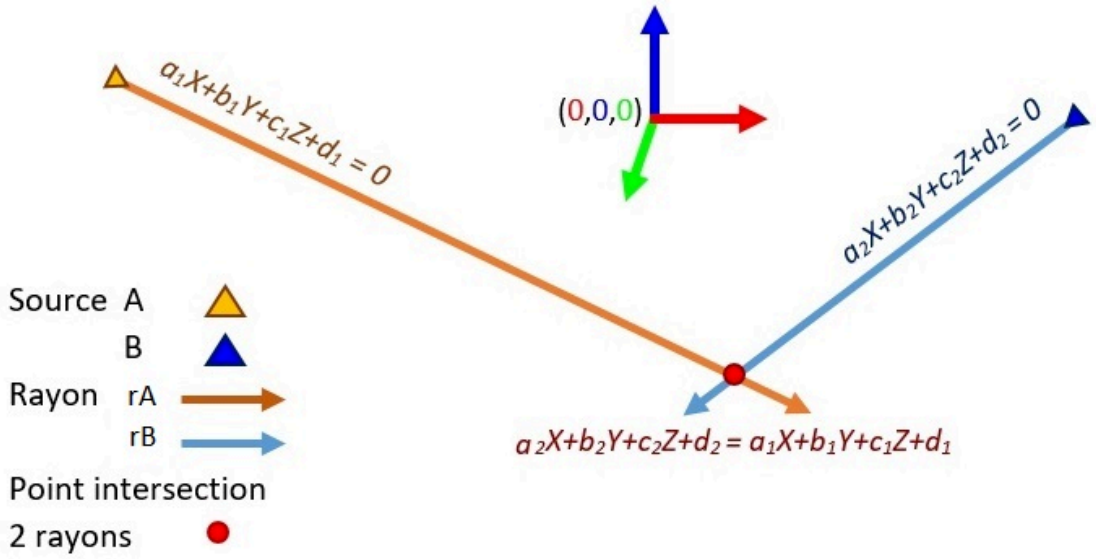


FIGURE 2.18 – Schéma du lancer de rayons

Dans le cas de la lumière structurée, ce n'est pas un mais une multitude de rayons qui sont lancés via le projecteur et la caméra. Les intersections, c'est-à-dire les coordonnées X, Y et Z des zones de l'objet, sont les informations recherchées.

Pour pouvoir estimer les intersections d'une multitude de rayons, plusieurs conditions sont requises.

La première condition est le calibrage du système constitué par la caméra et le projecteur. En effet, la matrice de translation (T) de la matrice extrinsèque modifie la source des rayons. La matrice de rotation (R) de la matrice extrinsèque ainsi que la matrice intrinsèque avec le centre optique et la focale de la caméra et du projecteur modifient la direction du rayon. Une faible modification du calibrage peut ainsi avoir des répercussions conséquentes, a fortiori si l'intersection des deux rayons (objet) se trouve loin des deux sources, ici le projecteur (émetteur) et la caméra (récepteur).

La deuxième condition est de pouvoir appairer les rayons de la caméra et du projecteur. Si les rayons appairés forment la bonne paire de rayons, c'est à dire que ce sont bien les deux rayons qui devaient être appairés, alors l'estimation de la profondeur sera correcte avec une faible erreur de mesure. Par contre, si les rayons appairés ne font pas partie de la même de paire, l'erreur de l'estimation de la profondeur sera importante. En s'appuyant sur l'équation détaillée dans le chapitre 2.1.1 du calibrage, cela revient à appairer les pixels de l'image du projecteur à ceux de l'image du capteur.

Cet appariement est divisé en 2 étapes :

1. la projection des mires encodant une information utile pour l'appariement ;
2. puis le décodage des motifs affichés par le projecteur sur l'objet et vus par la caméra, et donc l'appariement.

Le but du décodage est de pouvoir donner un "identifiant" unique à un pixel ou groupe de pixels. Le décodage est réalisé pour le projecteur et pour la caméra. Pour le projecteur, ce sont les motifs projetés par celui-ci qui permettent d'obtenir l'image avec des identifiants uniques. Pour la caméra, ce sont les motifs projetés sur l'objet vu par celle-ci qui permettent d'obtenir l'image avec des identifiants uniques.

Le type de mire projetée est associé à une méthode de codage et donc de décodage de l'information d'appariement.

Le décodage des mires permet de découper l'image en plusieurs zones. Plus les zones sont petites, plus l'appairage des rayons sera efficace et appaira les bons rayons.

Tout l'enjeu des différents types d'algorithmes dit d'appariement est de proposer

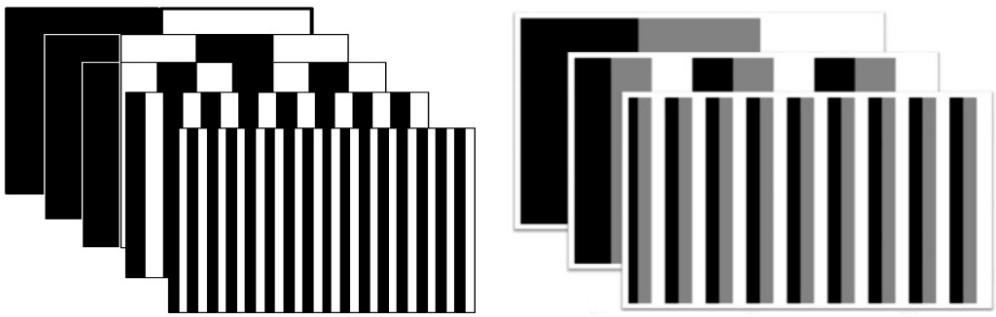
des méthodes permettant de coder le maximum de zones pour obtenir une haute densité d'acquisition avec une faible erreur d'appairage, tout en prenant en compte des contraintes comme la rapidité du traitement ou encore le fait que l'objet peut être en mouvement.

Les algorithmes d'appariement présentés dans la littérature [8] [34] [33] sont divisés en 3 grands groupes : les algorithmes utilisant des informations temporelles, spatiales ou hybrides.

2.1.3.2 Algorithmes d'appariement

Algorithmes utilisant des informations temporelles

Le premier groupe représente toutes les méthodes utilisant plusieurs motifs qui sont projetés sur l'objet. Ils utilisent principalement des motifs discrets, c'est-à-dire composés de bandes. Ils peuvent avoir des motifs séquentiels binaires[28][12] ou en niveaux de gris[13]. Ces méthodes sont pour la grande majorité utilisées pour des objets statiques.



(a) Motifs séquentiels binaires (b) Motifs séquentiels niveaux de gris

FIGURE 2.19 – Exemple de motifs séquentiels binaires ou en niveaux de gris

Le principe de ces algorithmes est de découper la zone de projection en plusieurs sous-zones. Elles sont codées grâce aux motifs qui sont projetés une à une dans le temps. En fonction du motif projeté, les zones ne vont pas recevoir la même valeur. Par exemple, pour l'algorithme Gray Code qui est représenté dans la Figure 2.20, les pixels visualisant une partie de la scène ayant été éclairés par le motif ont une valeur égale à 1, et les autres à 0. Pour avoir un code unique, chaque motif correspond à un bit du code. Donc pour deux motifs, le code de l'image pourra avoir 4 valeurs, allant de 00 à 11. Plus la zone de projection reçoit de motifs distincts, et donc d'informations, plus les zones sont petites.

La Figure 2.20 montre le découpage de ces zones dans le cas de l'algorithme Gray Code [12].

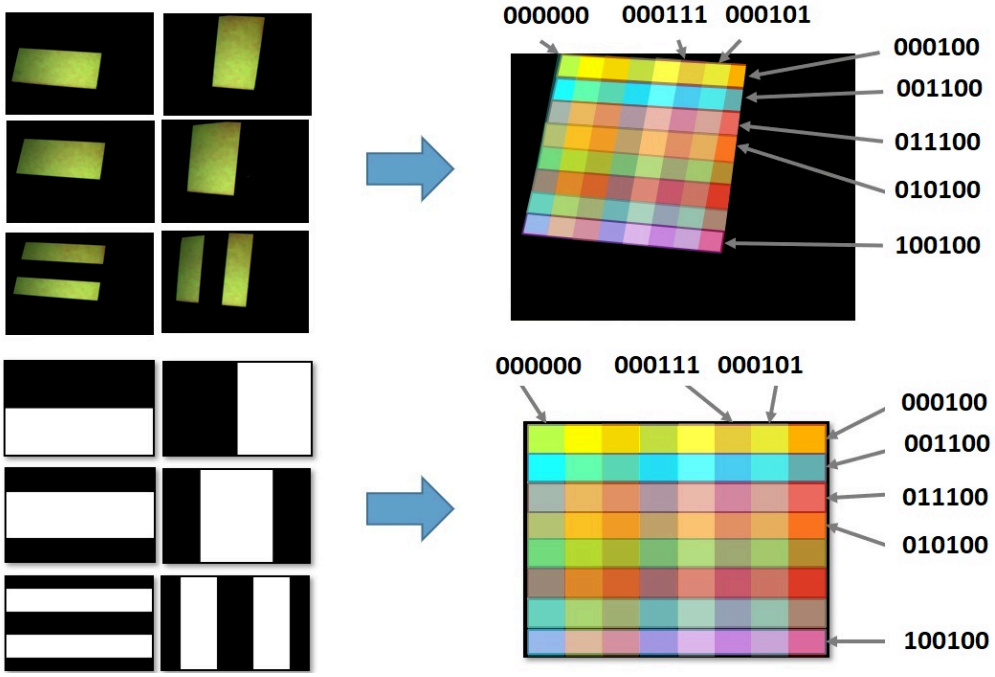


FIGURE 2.20 – Schéma de visualisation des mires et images successives vues par la caméra et découpage de ces images en zones recevant chacune un code unique. Dans cet exemple, les images de droite qui sont les motifs ou les images de la caméra voyant les motifs projetés permettent de retrouver le code de l'image. Si un code a comme valeur 000100, alors cette zone n'a été éclairé que par le quatrième motif projeté, sur 6 motifs projetés au total.

Le nombre de motifs utilisés définit donc l'erreur de mesure des zones estimées. En effet, plus la zone de projection est découpée en de multiples sous-zones, plus l'approximation d'appariement sera faible. Pour améliorer le découpage de la zone de projection, des bandes verticales et horizontales sont utilisées.

Dans le cas de la méthode Binary Code ou Gray Code, le nombre de zones codées est égal à $2^{nombreMotif}$. Donc pour 20 motifs (10 horizontaux et 10 verticaux), il est possible de coder finement 1 048 576 zones. Étant donné que la densité des zones codées est importante, l'approximation de l'appairage des rayons, et ainsi, l'erreur de mesure sont plus faibles.

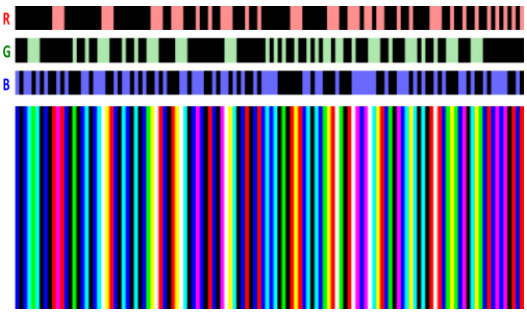
Les algorithmes utilisant des bandes discrètes sont sensibles à la défocalisation de la lentille ou au fait que l'objet sorte de la portée requise pour que celui-ci soit net.

Étant donné qu'il est nécessaire d'acquérir et de traiter plusieurs motifs pour calculer la profondeur de la scène, le temps de calcul est relativement long par rapport aux

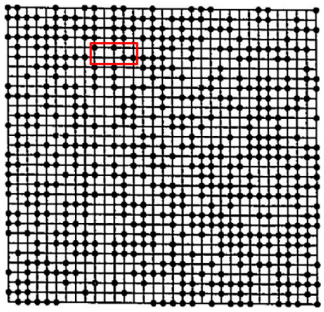
autres types d'algorithmes. Cependant il est à noter que la densité des zones acquises est importante par rapport à ces mêmes types d'algorithmes.

Algorithmes utilisant des informations spatiales

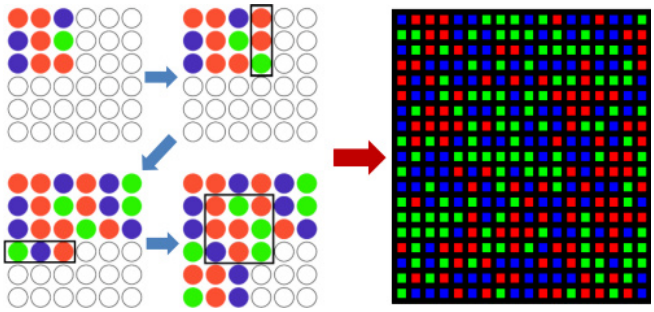
Le deuxième groupe représente toutes les méthodes qui utilisent un seul motif projeté sur l'objet pour extraire l'information de profondeur. Ils utilisent l'information spatiale, c'est-à-dire les données autour du pixel à analyser pour prédire l'identifiant de celui-ci. De ce fait, les objets en mouvement peuvent être acquis. Ces méthodes peuvent utiliser des motifs contenant des grilles [32][18], des raies [41] ou des matrices de points [6] grâce à des motifs en noir et blanc (binaire), en niveaux de gris ou en couleur.



(a) Exemples de motifs basés sur la séquence de Bruijn [41]



(b) Exemples de motifs grille [18]



(c) Exemples de motifs de matrices de points [6]

FIGURE 2.21 – Exemple de motifs spatiaux

Étant donné que ces algorithmes obtiennent l'identifiant unique du pixel à analyser grâce aux informations spatialement proches de ce pixel, les motifs sont codés de façon à ce que toutes les zones définissant un pixel soient uniques. Pour cela, ils parcourent l'image grâce une fenêtre glissante définie par avance par l'algorithme et déterminent l'identifiant des pixels grâce à la combinaison unique des informations présentes dans cette fenêtre.

Dans le cas de l'algorithme M-Array [6], la Figure 2.20 montre deux fenêtres permettant d'identifier un pixel spécifique dans le motif projeté et dans l'image, vue par la

caméra, du motif projeté sur l'objet.

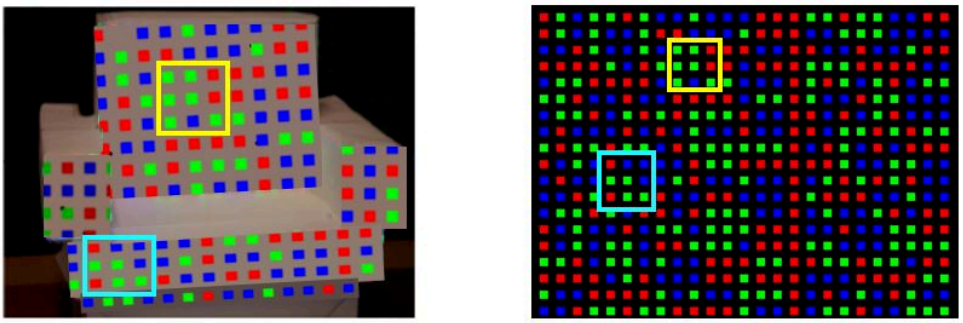


FIGURE 2.22 – Exemple de détection code point spécifique

Ces méthodes sont plus rapides à calculer mais la densité des points est plus faible et l'erreur de mesure plus importante que les méthodes utilisant plusieurs motifs.

Algorithmes hybrides

Le dernier groupe utilise plusieurs méthodes à la suite dont celles présentées précédemment ainsi que d'autres utilisant des motifs continus présentés dans le chapitre 2.1.2 pour estimer la profondeur. Combiner plusieurs méthodes vise à améliorer la densité, rapidité d'acquisition et/ou réduire l'erreur de mesure.

C'est notamment le cas de la méthode qui associe Gray Code et Phase Shifting [35]. La méthode Gray Code utilise des motifs binaires alors que la méthode Phase Shifting utilise des motifs continus. L'algortihme Gray Code utilise 6 motifs pour pré-coder l'image en grandes zones alors que le Phase Shifting permet d'avoir une phase unique en chaque pixel de chaque zone et ainsi un code unique. Cela permet d'estimer la profondeur grâce au lancer de rayons avec les pixels appairés de la caméra et du projecteur. Les motifs continus ont l'avantage de pouvoir coder une information continue et qui peut donc être estimée en subpixel, c'est-à-dire entre les pixels, ce qui n'est pas le cas en utilisant des motifs binaires. Cela permet ainsi de réduire encore plus l'erreur de mesure de cette méthode.

Cette méthode permet aussi de fortement diminuer le nombre de motifs utilisés et donc le temps de calcul. En effet, la méthode usuelle (Gray Code) nécessite au moins 20 motifs pour pouvoir coder une image 1024x1024 pixels alors que seulement 12 motifs, 6 pour le GrayCode et 6 pour le Phase shifiting, sont nécessaires pour une erreur de mesure identique, voire plus faible, avec la méthode combinée. Celle-ci permet d'avoir une erreur de mesure plus faible pour un temps de calcul plus restreint par rapport au Gray Code utilisé seul.

TABLE 2.1 – Comparaison de plusieurs méthodes, d'après la review [33]

Algorithme	**Méthode d'acquisition de profondeur**	**Densité**	**Erreur**	**Temps**
Posdamer[28]	Lancer de rayons Informations temporelles	+ +	-	-
Horn[13]	Lancer de rayons Informations temporelles	+ -	+	+ -
Gühring[11]	Décalage de phase Informations temporelles	+ +	+ +	- -
De Bruijn [16]	Lancer de rayons Informations spatiales	+-	+-	++
Salvi[32]	Lancer de rayons Informations spatiales	- -	- -	+ +
Morano[25]	Lancer de rayons Informations spatiales	-	-	+ +
Sato[36]	Lancer de rayons Informations temporelles	+ -	+ -	+

2.1.4 Bilan

Les systèmes de vision utilisant la lumière structurée nécessitent d'être calibrés avant de pouvoir estimer la profondeur. Pour cela, plusieurs méthodes existent dont certaines sont spécifiques pour des objets situés à plus de 3 mètres du système. Les algorithmes utilisés pour estimer la profondeur sont séparés en deux groupes. Le premier groupe utilise le décalage de phase. Deux méthodes semblent prometteuses et permettent d'obtenir assez rapidement une phase absolue et donc une distance relative avec une faible erreur de mesure et avec une robustesse aux erreurs dues aux forts décalages de profondeur et aux discontinuités liées aux objets. Ces algorithmes peuvent aussi être améliorés en augmentant la fréquence, c'est à dire le nombre de période visible sur les motifs, et le déphasage, c'est à dire le nombre de motifs projetés nécessaire pour calculer la phase.

Ce type d'algorithme a cependant un inconvénient : il ne donne pas de distances absolues par rapport à la caméra. La distance obtenue reste dépendante du fond de référence qui est obligatoire pour calculer la profondeur en différentiel.

Le deuxième groupe utilise la méthode du lancer de rayons. Il est constitué de plusieurs types d'algorithmes : les algorithmes utilisant des informations temporelles ou spatiales.

Le Tableau 2.1 comparant les différentes méthodes montre que les algorithmes utilisant des informations temporelles permettent d'obtenir la profondeur avec une densité des points d'acquisition supérieure et une erreur de mesure inférieure aux algorithmes utilisant des informations spatiales. Il est nécessaire de disposer de nombreux motifs pour obtenir une densité importante avec les algorithmes utilisant des informations temporelles. Cela implique un temps de calcul long.

Les méthodes hybrides associent plusieurs méthodes pour compenser les défauts des unes, i.e. en réduisant le temps de calcul, l'erreur de mesure et/ou en augmentant la densité des points de mesure. Certains de ces algorithmes utilisent des informations temporelles. C'est le cas du Gray Code + Phase shifting qui permet de réduire le temps de calcul et l'erreur de mesure.

Cette méthode, ainsi que les autres algorithmes utilisant le lancer de rayons, reste fortement tributaire de la qualité du calibrage. En effet, les paramètres estimés ne sont pas calculés pour l'appariement des points caméra-projecteur mais sont utilisés pour calculer les coordonnées 3D de l'objet. Enfin, que ce soit dans le cas des algorithmes de décalage de phase ou de ceux de triangulation, les tests réalisés dans la littérature sont faits sur des objets de petite taille (30 cm de hauteur) à courte distance (1 à 2 mètres). Dans le cas de notre application, par contre, l'acquisition réalisée sur des bus porte sur des pièces de plusieurs mètres de côté (de l'ordre de 3 mètres) mais l'erreur de mesure recherchée est 1000 fois inférieure. Les pièces sont de plus à une grande distance du système (au moins 5 mètres).

Étant donné qu'aucune étude ne compare les méthodes actuelles avec toutes les caractéristiques précédentes, l'étude de l'erreur de mesure sera réalisée pour justifier de la pertinence des algorithmes choisis dans la méthode proposée.

2.2 Choix des méthodes d'acquisition de lumière structurée

2.2.1 Comparaison des méthodes de décalage de phase

Les méthodes de décalage de phase permettent d'acquérir la profondeur des objets avec l'erreur de mesure la plus faible. Cependant, aucune métrique n'est donnée pour les comparer. Il est donc nécessaire de procéder à des tests pour comparer les approches. Pour acquérir la scène en 3D composée d'objets de grandes dimensions, plusieurs pa-

ramètres sont fixés.

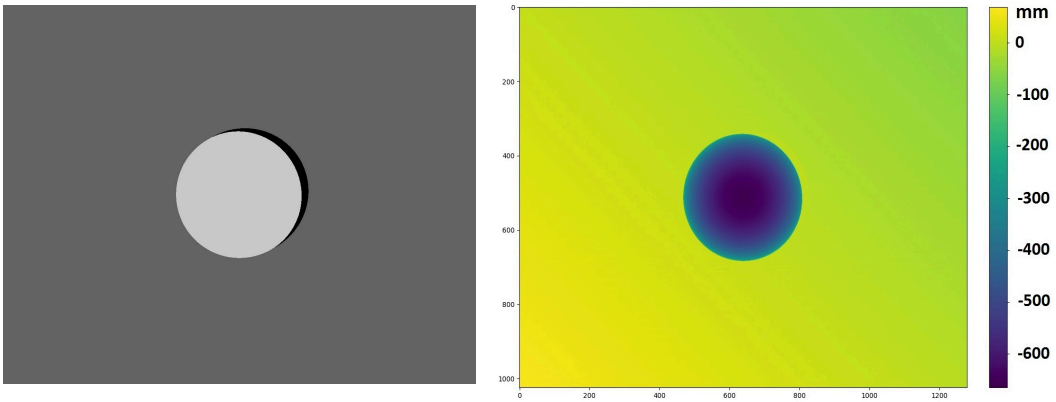
— le projecteur est situé approximativement à 1 m de la caméra, sur le même plan horizontal. Le projecteur est décalé sur le côté ce qui permet de visualiser le décalage des franges et donc d'estimer la profondeur ;

— la scène est fixée comme mesurant 3 mètres de hauteur et 4 mètres de largeur ;

— le système est situé à au moins 4,5 mètres de l'objet le plus proche ;

— l'erreur de mesure maximale acceptée est de plus ou moins 1,5 mm.

Pour avoir des résultats fiables et s'exonérer d'un contrôle par un autre moyen, les tests et analyses sont réalisés grâce à des scènes virtuelles, générées par simulation. Cela permet de fixer, et donc de connaître avec exactitude, le calibrage du système ainsi que les positions des objets pour ensuite qualifier l'erreur de mesure des algorithmes. Pour avoir une grande diversité de cas, c'est-à-dire des surfaces plus ou moins continues, les objets de référence sélectionnés pour estimer l'erreur de mesure des algorithmes sont une sphère positionnée devant un plan incliné. De tels objets sont souvent utilisés dans la littérature pour comparer deux algorithmes [19] [5]. La sphère décalée permet d'avoir des zones continues avec une faible variation de profondeur, des zones avec de fortes variations de profondeur et des zones avec des discontinuités. Le plan incliné permet de couvrir l'intégralité de l'image avec des distances différentes, de créer les zones de discontinuités avec la sphère et de prendre en compte la présence de l'ombre issue de la sphère.



(a) Image de la scène avec en gris foncé le plan incliné, en gris clair la sphère et en noir son ombre

(b) Image de la profondeur de la scène composée d'un plan incliné et d'une sphère

FIGURE 2.23 – Scène utilisée pour mesurer l'erreur de mesure des algorithmes

Les deux méthodes de décalage de phase prennent en paramètres d'entrée la fréquence des motifs, leur déphasage ainsi que la résolution de la caméra et du projecteur.

Étant donné que l'erreur de mesure attendue et la zone ont été fixées précédemment, la résolution de la caméra peut être estimée. Elle est fixée à 2000 x 1600 pixels. Cela permet d'avoir, en largeur, une définition de 2 mm par pixel (4000mm/2000pix = 2 $mm.pix^{-1}$) et, en longueur, de 1,8 mm par pixel (3000 mm / 1600 pix = 1,8 $mm.pix^{-1}$).

La résolution du projecteur peut, intuitivement, être initialisée à la même valeur que celle de la caméra. Cependant plusieurs caractéristiques étudiées dans les paragraphes suivants peuvent modifier ce choix.

Le but des tests est donc de déterminer :

— l'erreur de mesure obtenue avec les deux méthodes d'acquisition de décalage de phase ;

— la valeur de déphasage permettant d'obtenir une erreur de mesure minimale ;

— la fréquence la plus adéquate pour une résolution du projecteur faible.

— l'impact de la résolution du projecteur sur l'erreur de la mesure et sur la fréquence minimisant l'erreur de mesure ;

Les trois premiers points sont évalués dans le Chapitre 2.2.1.1 et la dernière caractéristique dans le Chapitre 2.2.1.2

2.2.1.1 Tests et choix des méthodes

Les tests suivants permettent de déterminer la méthode permettant de minimiser l'erreur de mesure et de déterminer les paramètres associés, comme le déphasage et la fréquence optimale. Pour cela, les tests sont réalisés sur un projecteur de résolution égale à 768 x 1024.

Les déphasages testés sont compris entre 3 et 9 avec un pas de 3, sauf pour la méthode Dual-Frequency qui commence au déphasage 5 comme décrit dans l'état de l'art (chapitre 2.1.2.2). Les fréquences testées sont comprises entre 16 et 384.

Pour pouvoir comparer les deux méthodes, la moyenne des erreurs absolues est utilisée. Les erreurs absolues permettent de prendre en compte toutes les erreurs sans qu'elles s'annulent entre elles.

On peut remarquer tout d'abord dans la Figure 2.24 que les deux méthodes étudiées ont leur moyenne des erreurs absolues qui augmentent fortement lorsque la fréquence est au-dessus de 160, pour atteindre un maximum à la fréquence 384. De plus, la méthode Bi-Frequency est moins impactée, c'est à dire que la moyenne des erreurs

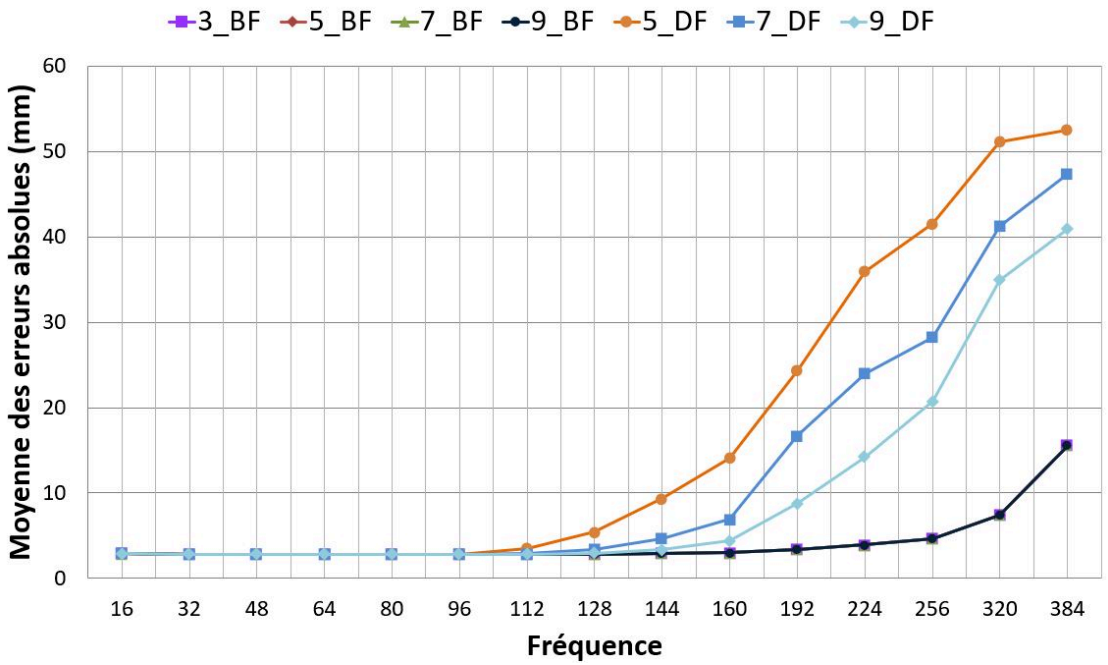


FIGURE 2.24 – Moyenne des erreurs absolues, en mm, sur l'axe Z de la scène comprenant une sphère et un plan incliné. Les marques (points, carrés, ...) sont les tests réalisés et les courbes sont les tendances d'évolution des moyennes

absolues augmente moins pour les fréquences au-dessus de 160, que la méthode Dual-Frequency. Enfin, plus le déphasage est important, moins l'erreur est importante.

En considérant en particulier les fréquences en dessous de 160 (Figure 2.25), les moyennes des erreurs des méthodes sont plus faibles entre la fréquence 64 et 112. Cette intervalle de fréquence est donc conservé pour la suite.

La Figure 2.26 permet de visualiser les deux méthodes d'estimation de la profondeur dans la plage minimisant l'erreur. La méthode Bi-Frequency permet d'obtenir les meilleurs résultats grâce à un déphasage égal à 9 et une fréquence comprise entre 80 et 96. La moyenne des erreurs absolues de la profondeur est alors de 2,819 mm.

Pour valider cette méthode ainsi que la fréquence et le déphasage, la même analyse est réalisée pour les axes X et Y.

La Figure 2.27 montre que les méthodes réagissent de la même manière pour les axes X et Y que pour l'axe Z. La méthode Bi-frequency est donc validée pour le reste du processus avec un déphasage égal à 9. La moyenne des erreurs absolues est de 0,328 mm en Y et de 0,428 mm en X.

Une analyse plus détaillée consiste à séparer les résultats issus de la sphère de ceux issus du plan incliné. En effet, le plan incliné a une profondeur relative par rapport au

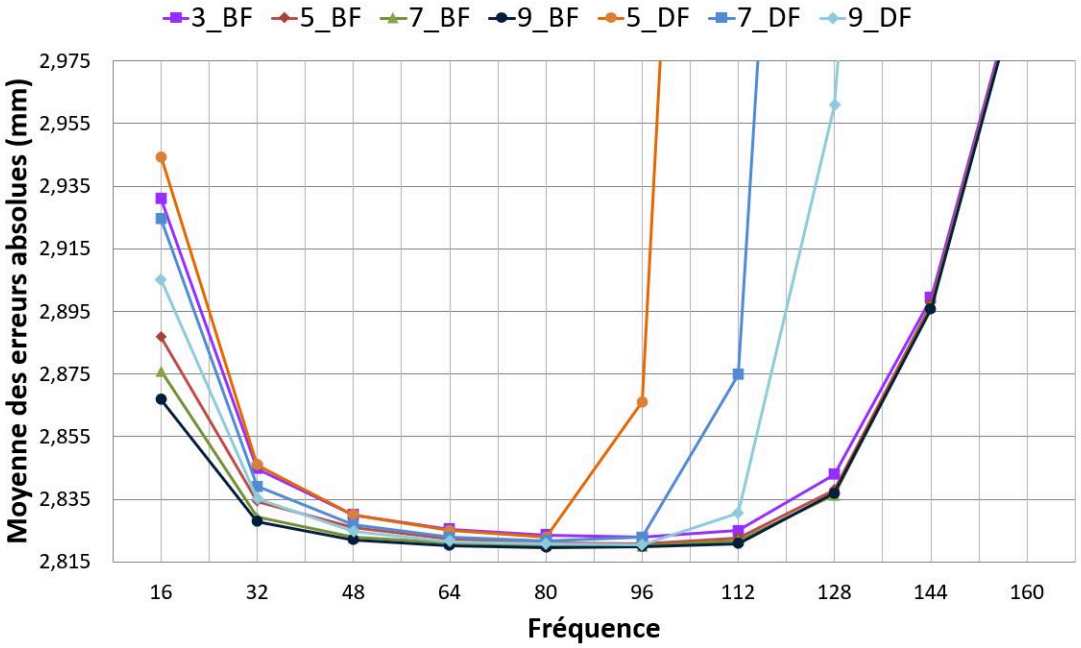


FIGURE 2.25 – Moyenne des erreurs absolues sur l'axe Z de la scène, en mm, comprenant une sphère et un plan incliné centré sur la plage de fréquences allant de 16 à 160. Les marques (points, carrés, ...) sont les tests réalisés et les courbes sont les tendances d'évolution des moyennes.

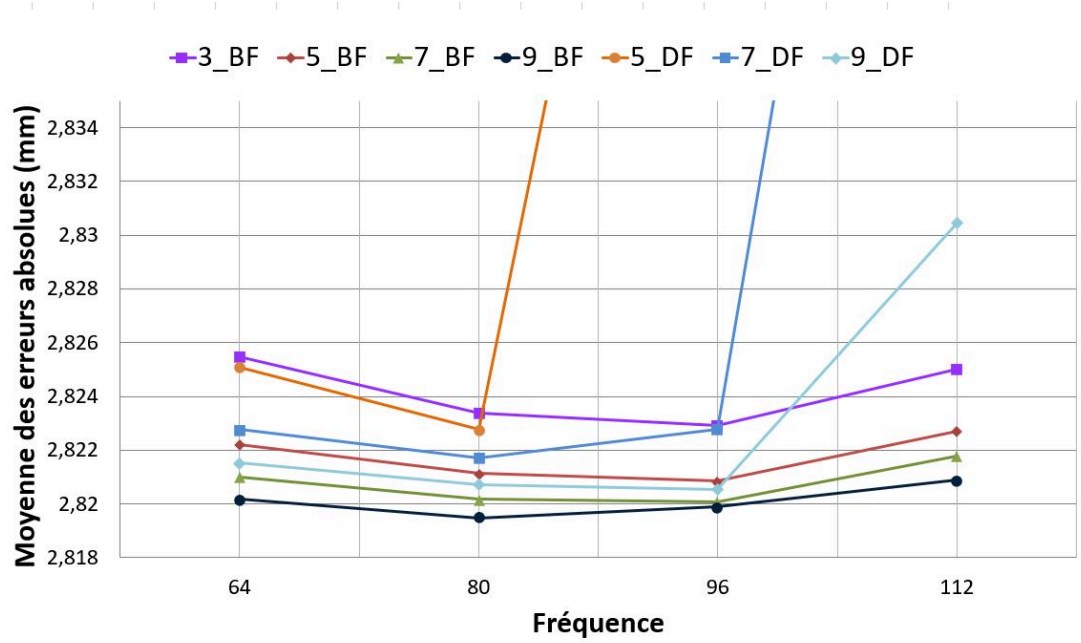
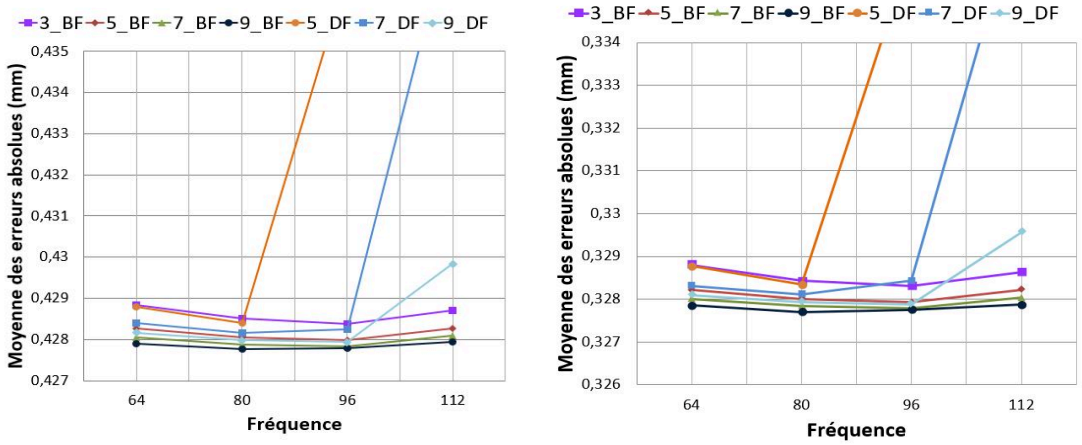


FIGURE 2.26 – Moyenne des erreurs absolues sur l'axe Z de la scène, en mm, comprenant une sphère et un plan incliné centré sur la plage de fréquences allant de 64 à 112. Les marques (points, carrés, ...) sont les tests réalisés et les courbes sont les tendances d'évolution des moyennes.



(a) Résultat des deux méthodes sur l'axe X (b) Résultat des deux méthodes sur l'axe Y

FIGURE 2.27 – Moyenne des erreurs absolues sur les axes X et Y de la scène, en mm, comprenant une sphère et un plan incliné centré sur la plage de fréquences allant de 64 à 112

TABLE 2.2 – Résultat de la méthode Bi-Frequency (BF) sur la scène générale, sur la sphère et le plan incliné pour une fréquence de 96 et un déphasage de 9

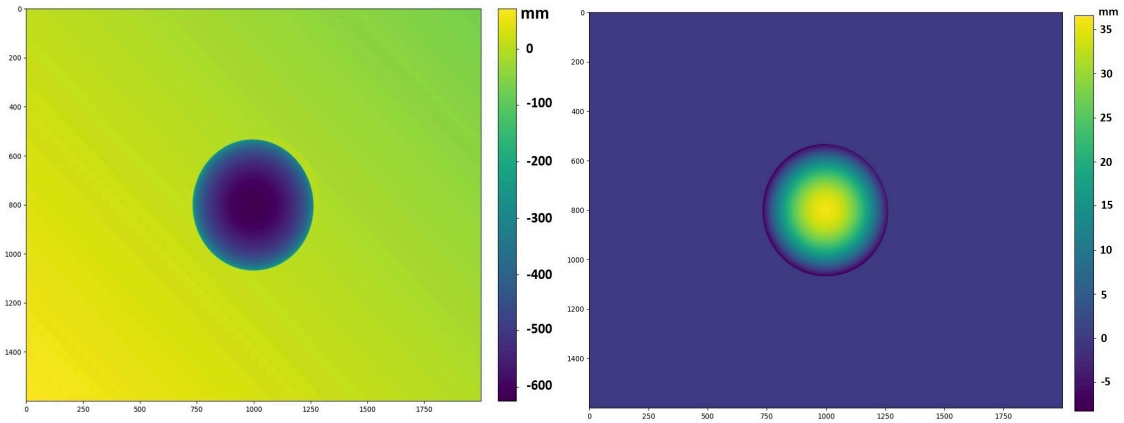
	Axe	Moyenne (mm)	Écart-type (mm)
Plan	X	-0,395	0,463
	Y	0,253	0,382
	Z	0,142	2,256
Sphère	X	-0,002	0,648
	Y	-0,004	0,647
	Z	14,613	12,429

plan de référence comprise entre -100 mm et 100 mm, alors que la sphère, qui a un rayon égal à 500 mm et qui est décalée de 100 mm par rapport au plan de référence, dépasse de 600 mm par rapport au plan de référence. Enfin, pour mieux comprendre les performances de l'algorithme, la moyenne et la variance des erreurs sont utilisées.

Le Tableau 2.2 montre que la moyenne des erreurs sur les 3 axes (X,Y,Z) est proche de 0 (inférieure à 0,4 mm) avec une variance très faible pour les axes X,Y (inférieure à 0,5 mm) et une variance faible pour l'axe Z (2,3 mm), pour le plan. La variance de l'axe Z dépasse cependant l'erreur de mesure maximale voulue.

Pour la sphère, l'erreur moyenne en X et Y est très proche de 0 avec une variance faible (inférieur à 0,7 mm). Cependant, la moyenne (14,6 mm) et la variance (12,4 mm) sur l'axe Z sont fortes. Cela peut s'expliquer par le fait que la sphère est plus éloignée du plan de référence que le plan incliné. La distance par rapport au plan de référence est

donc essentielle. Il s'avère que plus cette distance est faible, plus l'erreur est faible. La Figure 2.28 permet de visualiser la sortie de la méthode Bi-Frequency ainsi que l'erreur d'estimation de la méthode sur la scène.



(a) Image de sortie de la méthode Bi-Frequency

(b) Erreur de la méthode Bi-Frequency

FIGURE 2.28 – Image de sortie de la méthode Bi-frequency (a) et l'image de l'erreur de cette méthode sur la scène (b), sur l'axe Z

2.2.1.2 Impact de la résolution du projecteur

Les résultats présentés précédemment ne sont valables que pour une résolution de projecteur donnée. Pour choisir cette valeur, il est plus simple de sélectionner la résolution du projecteur permettant d'obtenir la largeur de la zone acquise par un pixel, exprimée en millimètre. La résolution est donc la taille de la surface à analyser(mm) divisée par la zone acquise par un pixel (mm).
Cependant, étant donné que le projecteur est décalé sur le côté par rapport à la caméra, les mires ne sont pas projetées exactement sur la même surface que celle perçue par la caméra. Selon les réglages appliqués, la surface de projection peut être plus petite ou, à l'inverse, plus grande que celle perçue et donc, la surface couverte par un pixel (du projecteur vs la caméra). Par ailleurs, plus le projecteur est proche des surfaces à analyser, moins la surface à analyser est grande. Enfin, plus les surfaces sont perpendiculaires au projecteur, moins la projection est déformée et de taille réduite. La résolution millimétrique en chaque pixel est donc variable en fonction de ces critères et peut même être différente selon la portion de l'image considérée.

Avec un projecteur de 768 pixels de hauteur qui affiche les mires sur une surface de hauteur 3000 mm, la résolution est approximativement égale à 3,9 mm par pixel. Or

c'est bien supérieur à l'erreur voulue (2 mm) sur l'axe X et Y. De plus, la variance des erreurs sur l'axe Z (> 2mm) indique que la méthode Bi-Frequency avec la résolution du projecteur actuel ne permet pas d'obtenir un résultat avec une erreur de mesure plus faible que celle souhaitée.

Pour vérifier que la résolution du projecteur n'a pas d'impact direct sur l'estimation finale de la profondeur, un nouveau test est mené en prenant cette fois en entrée un projecteur de résolution 6000 x 8000 pixels avec la méthode Bi-Frequency.

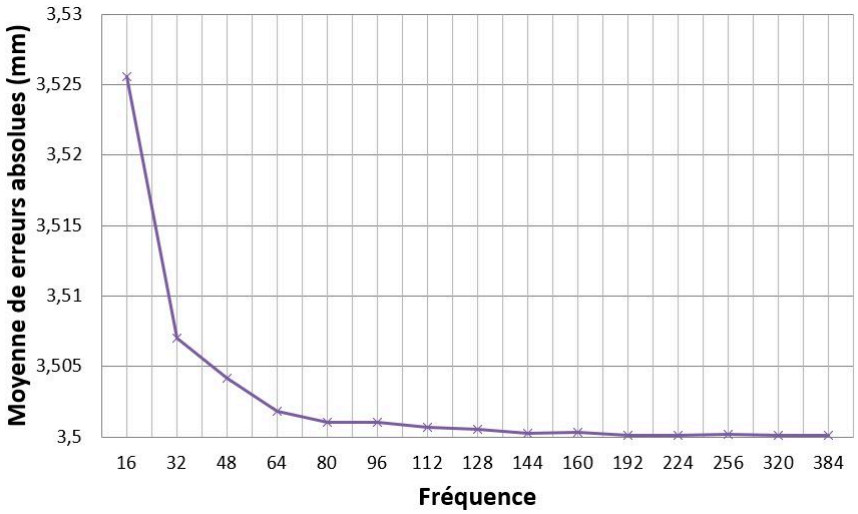


FIGURE 2.29 – Moyenne des erreurs absolues sur l'axe Z de la scène comprenant une sphère et un plan incliné pour un projecteur haute résolution avec la méthode Bi-frequency et un déphasage égale à 9

La Figure 2.29 montre que les résultats des fréquences basses répondent de la même manière qu'avec un projecteur basse résolution, c'est à dire que la moyenne des erreurs absolues des fréquences faibles (entre 16 et 48) est plus importante que la moyenne des erreurs absolues des fréquences supérieurs (entre 80 et 112) . Cependant, les hautes fréquences (>160) ne répondent pas de la même manière. En effet, le minimum n'est pas en la fréquence 96 mais au-delà de la fréquence 384, fréquence limite de cette étude.

Les résultats de la fréquence 96 pour un projecteur basse résolution et les résultats de la fréquence 384 pour un projecteur haute résolution (Tableau 2.3) semblent très proches dans la plupart des cas, avec une erreur légèrement plus élevée dans le cas du projecteur haute résolution. La résolution du projecteur a donc peu d'impact sur l'erreur de mesure de la méthode. Le gain apporté par un projecteur haute résolution étant très faible, il n'est donc pas nécessaire de mettre en place un tel projecteur.

D'après les Figures 2.24, 2.25 et 2.29, il est possible de dire qu'en augmentant la fréquence des motifs projetés par le projecteur haute résolution, l'algorithme Bi-

TABLE 2.3 – Résultat de la méthode Bi-Frequency (BF) sur la scène générale avec des projecteurs basse et haute résolution

		Basse Fréquence		**Haute Fréquence**	
	Axe	**Moyenne (mm)**	**Écart-type (mm)**	**Moyenne (mm)**	**Écart-type (mm)**
Plan	X	-0,395	0,464	-0,395	0,421
	Y	0,253	0,382	0,252	0,338
	Z	0,142	2,256	0,141	2,237
Sphère	X	-0,002	0,648	-0,002	0,628
	Y	-0,004	0,647	-0,003	0,629
	Z	14,613	12,429	14,591	11,833

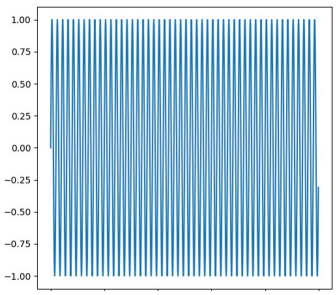
frequency se comporte de la même manière avec un projecteur haute résolution et basse résolution. En effet, l'erreur est plus importante pour les faibles fréquences que pour les hautes. De plus, les résultats décroissent régulièrement jusqu'à former un plateau. La partie des résultats est coupée lorsque la fréquence est très haute. Cela montre qu'utiliser des motifs en haute fréquence est essentiel pour obtenir une erreur de mesure faible sur la profondeur. Cependant, la résolution du projecteur et de la caméra limitent cette fréquence et peuvent dégrader le résultat si la fréquence est trop importante.

Les motifs projetés ainsi ceux perçus par la caméra ne doivent pas être en dessous de la fréquence limite définie par le théorème de Nyquist-Shannon. Ce théorème précise que la fréquence d'échantillonnage d'un signal doit être au minimum égale à deux fois celle du signal à échantillonner. Le motif avec la fréquence la plus haute qui respecte ce théorème est codée par 2 pixels en chaque période. Dans le cas du projecteur basse résolution, la fréquence limite est 384. C'est notamment pour cela que l'erreur augmente fortement jusqu'à cette fréquence. Cependant, même lorsque ce théorème est respecté, les résultats ne sont pas satisfaisants. Ils s'améliorent au fur et à mesure que la fréquence diminue jusqu'à 96. Cette dégradation provient du repliement de phase constaté lorsque la fréquence est supérieure au nombre total de pixels divisé par 8.

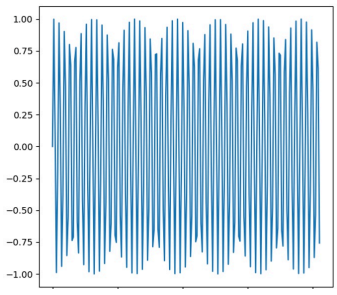
Le repliement de phase permet de reconstruire le signal initial avec cependant l'ajout de basses fréquences non désirées dans les motifs.

Dans la Figure 2.30, 8 pixels pour échantillonner la période permettent de reconstruire assez fidèlement le signal original par rapport aux fréquences plus basses.

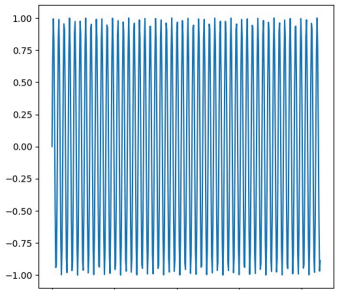
Dans le cas du projecteur basse résolution, la fréquence 96 correspond à une période codée par 8 pixels (fréquence d'échantillonnage). Pour le projecteur haute résolution, la fréquence correspondant à la fréquence d'échantillonnage 8 est 6000/8 = 750.
Si les tests étaient poursuivis avec des fréquences plus importantes pour le projecteur



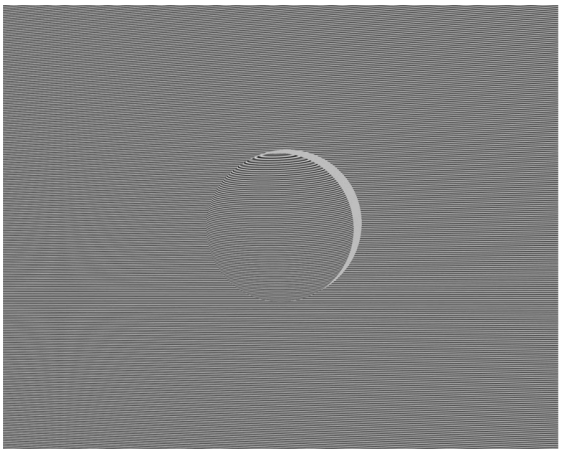
(a) Signal échantillonné par 100 pixels (pas de repliement de phase)



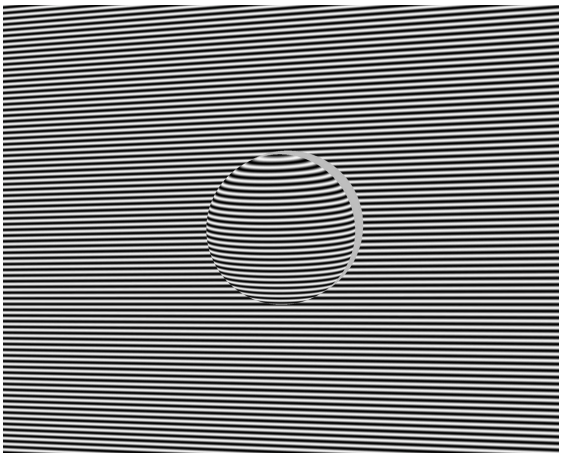
(b) Signal échantillonné par 4 pixels (repliement de phase)



(c) Signal échantillonné par 8 pixels (très peu de repliement de phase)



(d) Scène échantillonnée par 4 pixels par période et donc contenant du repliement de phase



(e) Scène échantillonnée par 8 pixels par période et ne contenant pas de repliement de phase

FIGURE 2.30 – Illustration du repliement de phase sur une sinusoïde parfaite (Figures a, b et c) et sur la scène (Figures d et e)

haute résolution, les meilleurs résultats seraient situés aux alentours de la fréquence 750. Cependant, la réduction de l'erreur de mesure ne serait pas flagrante par rapport à celle obtenue grâce à un projecteur basse résolution.

2.2.1.3 Conclusion

En conclusion, la méthode Bi-Frequency est retenue par rapport à la méthode Dual-frequency car c'est la méthode qui permet de minimiser l'erreur de mesure des objets selon les axes en X, Y et Z, contrairement à se qui est indiqué dans la review [45]. De plus cette méthode est plus robuste pour le déroulement de phase du fait de la séparation de la projection des images de basse fréquence et de haute fréquence se qui permet de bien distinguer les différentes des deux fréquences là où la méthode Dual-frequency combine les deux fréquences sur les motifs projetés.

Le fait que la moyenne et la variance soient proches de 0 mm permet de dire que cette méthode ne déforme pas la scène lors de l'acquisition des coordonnées 3D de celle-ci. Cependant, plus la profondeur de l'objet s'écarte du plan de référence, plus l'erreur est importante. Il est donc nécessaire d'utiliser plusieurs plans de référence pour une bonne acquisition de la scène, bien que cela complexifie et ralentit le processus. Étant donné que l'erreur de mesure obtenue change peu en fonction de la définition du projecteur, un projecteur doté d'une résolution de 768 x 1024 pixels (basse résolution) est sélectionné. Les expériences menées permettent de choisir une fréquence égale à 96 et un déphasage de 9.

Bien que cette méthode soit la méthode utilisant le décalage de phase qui permet d'obtenir une erreur de mesure la plus faible, elle ne permet pas d'obtenir une distance absolue, c'est à dire la distance entre l'objet et la caméra. Les méthodes utilisant le lancer de rayons permettent de délivrer une image de profondeur absolue. Le chapitre suivant test la méthode Gray Code + Phase shifitng et la compare à la méthode Bi-frequency.

2.2.2 Comparaison méthodes Bi-frequency et Gray Code + Phase Shifting

La méthode Bi-Frequency permet d'acquérir la scène avec une déformation très faible selon les axes X et Y, et ce avec une erreur satisfaisante au niveau de la profondeur (Z), surtout si l'objet est proche du plan de référence. Cependant, la méthode hybride associant Gray Code et Phase Shifting semble très robuste et permet d'appairer les pixels de la caméra avec ceux du projecteur au niveau subpixelique .

Des tests sont donc réalisés pour comparer les deux approches et déterminer, dans un premier temps, si l'une d'elle permet une meilleure estimation des coordonnées 3D de la scène. Dans un second temps, il s'agit de qualifier l'impact de l'erreur de calibrage sur chacune des deux méthodes.

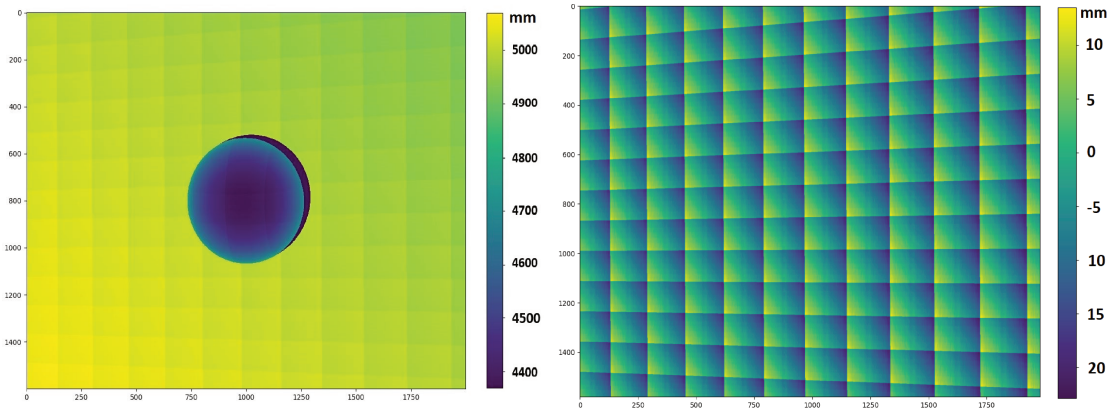
2.2.2.1 Test Gray Code associé à Phase Shifting

Contrairement à la méthode Bi-Frequency, l'association de Gray Code + Phase shifting (GC+PS) permet d'estimer une profondeur absolue et non relative. La distance est ainsi mesurée par rapport à la caméra positionnée dans le référentiel monde, et non par rapport à un plan donné.

Scène complexe

Les tests sont réalisés sur la même scène que précédemment : une sphère située devant un plan incliné. De plus, les résolutions et emplacements de la caméra et du projecteur sont inchangés.

L'image obtenue en sortie ainsi que l'erreur de la méthode est visible sur la Figure 2.31. Cette sortie correspond bien à l'ordre de grandeur des profondeurs attendues, mais délivre énormément d'erreur. Les artefacts, i.e. des carrés, sont visibles dans la Figure 2.31. Ces carrés ne sont toutefois pas réels, c'est à dire qu'ils ne sont pas présent sur l'objet à analyser, et proviennent directement de la méthode Gray Code + Phase Shifting.



(a) Image de sortie de la méthode GC+PS (b) Erreur de la méthode GC+PS

FIGURE 2.31 – Image de sortie de la méthode Gray Code + Phase Shifting (a) et l'image de l'erreur de cette méthode sur la scène (b), sur l'axe Z

Comme pour les méthodes de décalage de phase, la méthode Gray Code + Phase Shifting peut être modulée par la fréquence des franges des motifs binaires et sinusoïdaux. Plusieurs valeurs de fréquences de franges et de déphasage sont donc testées pour identifier celles permettant de minimiser l'erreur sur la distance calculée.

Comme précédemment, les méthodes Gray Code + Phase Shifting et Bi-frequency sont comparées en utilisant la moyenne absolue des erreurs.

D'après la Figure 2.32, les fréquences 16 et 32 sont celles qui permettent de minimiser l'erreur absolue de l'estimation de la profondeur. Le minimum de la moyenne absolue est de 6,59 mm. Il est atteint pour une fréquence de 16 et un déphasage de 7, ce qui constitue la meilleure combinaison. Cependant, un déphasage de 3 permet de travailler avec 6 motifs pour la partie Phase Shifting contre 14 avec un déphasage de 7. Cela réduit significativement le temps de traitement et l'erreur, égale à 6,66 mm, est reste proche de la valeur minimale.

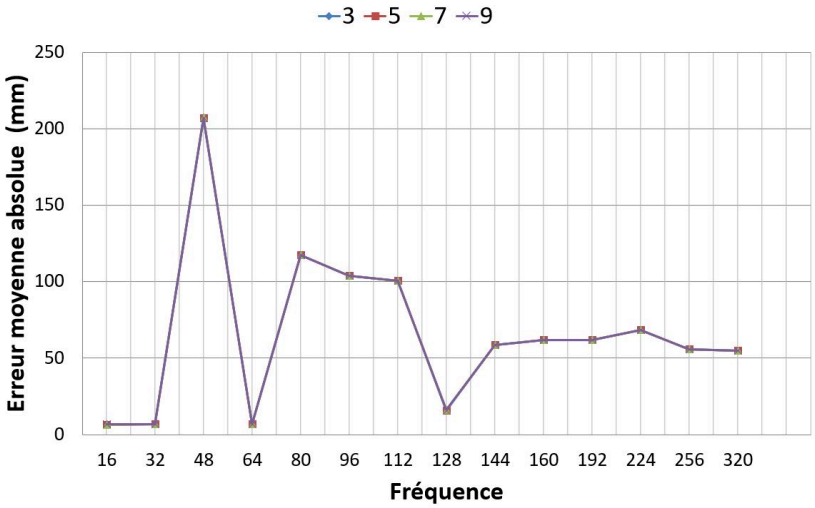


FIGURE 2.32 – Graphique de la moyenne des erreurs absolues sur l'axe Z de la scène comprenant une sphère et un plan incliné pour la méthode GC+PS

TABLE 2.4 – Erreur de la méthode Gray Code + Phase Shifting sur la scène générale, sur la sphère et le plan incliné pour une fréquence de 16 et un déphasage de 3

	Axe	Moyenne (mm)	Écart-type (mm)
Plan	X	-0,153	1,992
	Y	-0,337	1,667
	Z	4,777	7,245
Sphère	X	-0,183	0,342
	Y	-0,309	0,382
	Z	3,704	5,390

Le Tableau 2.4 montre la moyenne et la variance des erreurs. Pour le plan de référence, la moyenne est faible (inférieure à 0,40 mm) mais la variance est au dessus de 1,50 mm pour les axes X et Y. Les valeurs sur selon l'axe Z sont moins bien estimés : la moyenne est de 4,78 mm et surtout la variance atteint 7,25 mm.

Concernant la sphère, les moyennes et variances sont faibles (en dessous de 0,40 mm) pour les axes X et Y. Comme pour le plan, les coordonnées selon l'axe Z sont moins bien estimées, avec cependant une variance plus faible que pour le plan (5,39 mm).

La comparaison des deux approches à l'aide des Tableaux 2.2 et 2.4 montre que la méthode Bi-Frequency donne de meilleurs résultats que Gray Code + Phase Shifting (moyenne des erreurs et variance faibles). Cependant, plus la distance avec le plan de référence est grande, plus l'estimation des coordonnées est impactée.

La méthode Gray Code + Phase Shifting est donc une méthode qui ne peut pas être

utilisée pour estimer avec très peu d'erreur les coordonnées 3D de l'objet. Cependant, la différence d'erreur entre deux objets plus ou moins proche de la caméra, est faible par rapport à la méthode Bi-Frequecy.La méthode Gray Code + Phase Shifting peut donc être utilisée dans le cas où l'on veut connaître un scène dans son ensemble avec une erreur la plus faible possible, quelque soit la position et la distance des objets. Cela est dû notamment au fait que la méthode permet d'acquérir une distance absolue.

2.2.2.2 Test des différentes méthodes avec une erreur de calibrage

Les deux algorithmes utilisent le calibrage pour calculer la profondeur. Toutefois, le calibrage est estimé par des algorithmes et n'est pas parfait. Les deux méthodes sont donc testées pour mettre en exergue la méthode la plus robuste au erreurs de calibrage du système et identifier le gain d'erreur de mesure induite par ces erreurs de calibrage. Pour cela, deux tests sont menés. Le premier introduit une erreur de +0,1 puis de -0,1 sur la matrice de rotation et de translation de la matrice extrinsèque. Cela veut dire, par exemple pour l'erreur +0,1, que pour la matrice extrinsèque de rotation, cette matrice aura ces angles augmentés de 0,1°. Pour la matrice de translation, les valeurs des translations pour chaque axe seront augmentées de 0,1 mm.

Le deuxième introduit une erreur de +0,1 puis de -0,1 sur les matrices intrinsèques de la caméra et du projecteur. Cela se traduit par le fait que pour une erreur de +0,1, la focale sera augmentée de 0,1 mm et le centre optique de 0,1 pixel.

Ces valeurs (0,1 et -0,1) sont choisies car elles sont représentatives de l'erreur qu'il est possible de rencontrer en fin de calibrage. Les erreurs sont appliquées une par une, pour étudier l'influence de ces erreurs sur le résultat final.

D'après les résultats du tableau 2.5, la méthode Gray Code + Phase Shifting est très sensible aux erreurs dues au calibrage.

En effet, lorsque l'erreur est située dans la matrice extrinsèque, la moyenne et la variance sont multipliées par des coefficients respectivement de 7 à 10 et de 2 à 5.
Lorsque l'erreur est introduite dans la matrice intrinsèque, une erreur de $\pm 0,1$ multiplie la moyenne de l'erreur par 19 au minimum et la variance par 11.

L'erreur de calibrage a par contre très peu d'impact sur l'erreur de mesure de la méthode Bi-Frequency, que ce soit sur la moyenne (moins de 0,01 mm de différence) ou sur la variance des résultats (moins de 0,04mm).

TABLE 2.5 – Erreurs absolues de la méthode Gray Code + Phase shifting et Bi-Frequency sur la scène générale, sur la sphère et le plan incliné avec une erreur de plus ou moins 0,1

				Erreur Extrinsèque		**Erreur Intrinsèque**	
	Métrique (mm)	**Axe**	**Pas d'erreur**	**- 0,1**	**+ 0,1**	**- 0,1**	**+ 0.1**
GC+PS	**Moyenne**	**X**	1,3	9,6	11	219	198
	Écart-Type		1,38	6,31	7,28	140,54	119,72
	Moyenne	**Y**	1,1	7,7	9	181	160
	Écart-Type		1,22	4,88	5,83	115,83	97,09
	Moyenne	**Z**	6,6	48,9	59	129	123
	Écart-Type		5,35	10,53	10,86	69,99	58,20
BF	**Moyenne**	**X**	0,42	0,42	0,42	0,42	0,42
	Écart-Type		0,44	0,44	0,44	0,44	0,44
	Moyenne	**Y**	0,32	0,32	0,32	0,32	0,32
	Variance		0,33	0,33	0,33	0,33	0,33
	Moyenne	**Z**	2,81	2,81	2,81	2,81	2,81
	Écart-Type		4,76	4,78	4,73	4,74	4,74

2.2.3 Bilan

En conclusion, les tests réalisés permettent de déterminer qu'aucune des trois méthodes étudiées ne respecte les conditions initiales, c'est à dire une erreur de mesure inférieure à 2mm et une image de profondeur absolue.

La méthode Dual-Frequency, bien que mise en avant dans la review comme étant la méthode ayant l'erreur de mesure la plus faible, n'a pas été sélectionné car les tests montrent le contraire (avec cependant un écart très faible entre cette méthode et la méthode Bi-Frequency). De plus, comme indiqué dans la review, cette méthode est moins robuste au déroulement de phase.

La méthode Bi-Frequency s'avère être la méthode ayant l'erreur de mesure la plus faible. Cependant cette erreur dépasse l'erreur maximale souhaitée (2mm) et délivre une profondeur relative. Elle est établie à partir d'un plan de référence. Cette méthode augmente les erreurs d'estimation de la profondeur lorsque l'objet considéré s'éloigne du plan de référence. Cela est donc problématique pour pouvoir estimer avec très peu d'erreur une scène complète avec des objets situés à des profondeurs différentes.

La méthode Gray Code + Phase Shifting est, au contraire de la méthode Bi-frequency, très sensible au calibrage. Elle ne permet pas d'obtenir la valeur de la profondeur avec

une faible erreur. Par contre, elle permet de s'affranchir du plan de référence qui peut être difficile à mettre en oeuvre selon la scène considérée. La valeur de profondeur est ainsi une valeur "absolue", par rapport à la caméra.

En conséquence, il est proposé d'associer ces deux méthodes (Bi-frequency et Gray Code + Phase Shifting) pour combiner leurs avantages et ainsi acquérir avec une très faible erreur de mesure la profondeur en tout point d'une scène de grande envergure et à longue portée.

2.3 Méthode proposée : combinaisons des méthodes d'acquisition de lumière structurée

La méthode proposée combine les méthodes Gray-Code + Phase Shifting et Bi-Frequency pour estimer la profondeur d'une scène. Elle comporte ainsi quatre étapes :

1. le calibrage du système ;
2. l'estimation de l'emplacement des plans de référence ;
3. l'acquisition des plans de référence ;
4. l'estimation de la profondeur dans une scène.

2.3.1 Calibrage du système et des algorithmes

Le système, composé d'un projecteur et d'une caméra, est fixé de façon à ce que le projecteur et la caméra puissent respectivement projeter et capturer l'intégralité de la zone à analyser. Cette zone est définie par une distance minimale et maximale, ainsi que des coordonnées en X et Y définies. Chaque pixel de la caméra voit donc un pixel projeté par le projecteur sur l'objet, comme représenté dans la Figure 2.33.

Une fois les éléments du système positionnés, ils ne doivent plus être modifiés ou déplacés. Le cas échéant, l'ensemble des informations issues des étapes suivantes ne seront plus valables et ces opérations devront être reconduites.

Le calibrage du système est effectué. Comme vu dans le chapitre 2.1.1, plusieurs méthodes sont possibles pour paramétrer un système éloigné de l'objet à analyser.

Cette étape est primordiale et va conditionner les résultats des résultats des méthodes Bi-Frequency et, en particulier, Gray-Code + Phase Shifting.

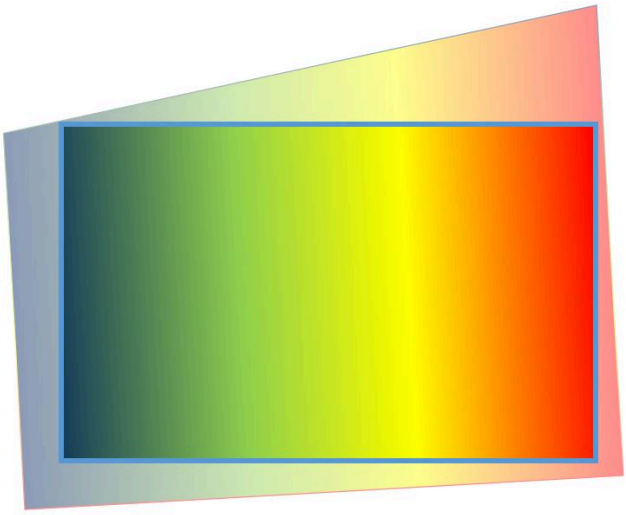


FIGURE 2.33 – Représentation de la partie, et donc des pixels (zone multicolore) , que le projecteur affiche dans la scène et que voit la caméra (carré bleu). La partie multicolore atténuée correspond à la zone éclairée par le projecteur mais non perçue par la caméra.

2.3.2 Estimation de l'emplacement des plans de référence

Comme vu précédemment, la méthode Bi-frequency semble augmenter l'erreur d'estimation en fonction de la distance plan - objet. Deux inconnues sont donc présentes et empêchent l'estimation de la profondeur des plans de référence. Ces deux inconnues sont les erreurs de la méthode : la première selon la distance entre l'objet et le plan de référence, la seconde selon la distance entre le plan de référence et la caméra.

Pour limiter le nombre de tests, des bornes de profondeur sont établies. Dans notre cas, elles sont fixées à 4250 et 5750 mm, comme définis dans le cahier des charges (Chapitre 1.2). Le système est ainsi paramétré pour réaliser une mesure entre ces deux bornes.

La Figure 2.34 représente l'évolution de l'erreur d'estimation en fonction de la distance plan de référence - objet. Les données évoluent proportionnellement à cette distance. Une fonction linéaire et une fonction cubique sont également tracées, respectivement en bleu et en rouge. La courbe rouge semble mieux correspondre à l'évolution de l'erreur et plus centrée sur ces données que la courbe bleue. Et bien que les données suivent une tendance, elles ne sont pas groupées autour de la fonction cubique mais sont disséminées autour de celle-ci avec une variance forte et qui change en fonction de la distance. Pour minimiser l'erreur et atteindre au maximum l'erreur de mesure souhaitée, cette variance doit être prise en compte.

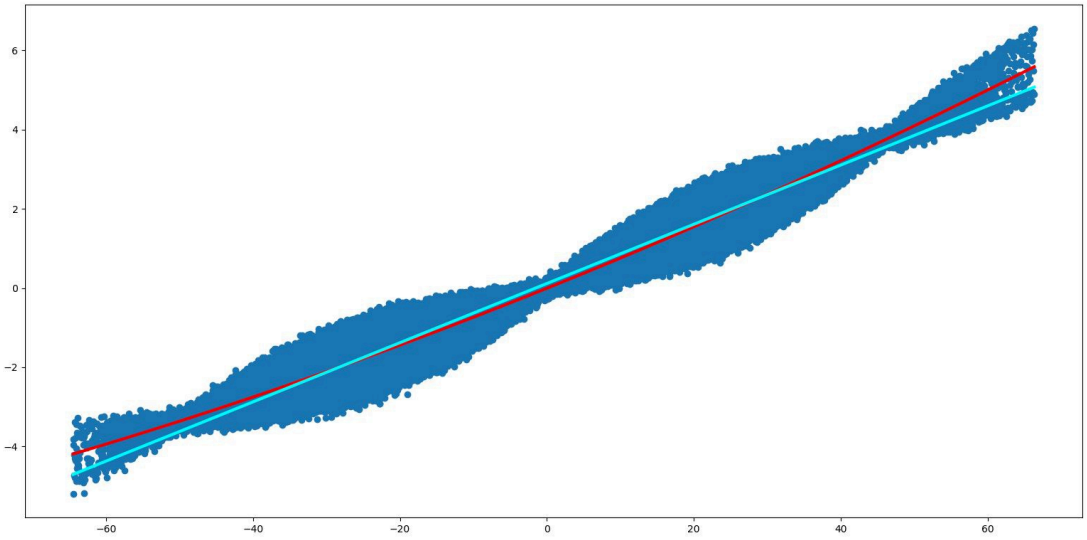
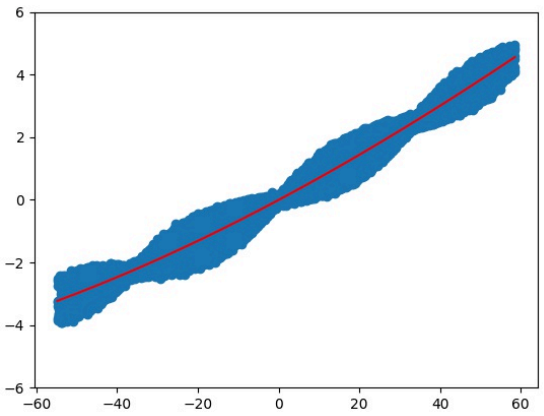
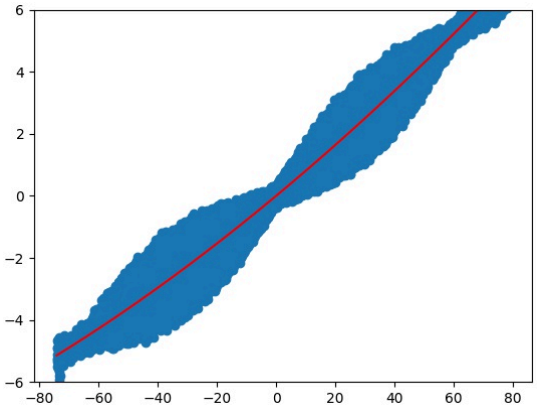


FIGURE 2.34 – Graphique représentant l'erreur sur l'axe Z en fonction de la profondeur entre l'objet et le plan de référence pour des distances de plan égales à 5000 mm. L'évolution des données est modélisée par la courbe rouge, une fonction du troisième degré, et la courbe cyan, une fonction linéaire.



(a) Distance du plan de référence égale à 4250 mm



(b) Distance du plan de référence égale à 5750 mm

FIGURE 2.35 – Graphique représentant l'erreur sur l'axe Z en fonction de la profondeur entre l'objet et le plan de référence pour les distances aux bornes, soit 4250 et 5750 mm.

La Figure 2.35 représente l'évolution de l'erreur d'estimation en fonction de la distance plan de référence - objet lorsque le plan de référence est positionné aux bornes. L'évolution des données pour les deux cas est différente. En effet, pour une distance de 4250 mm, la fonction croît plus lentement que celle associée à une distance de plan égale à 5750 mm. Cela montre que la fonction cubique n'est valable que pour une distance définie. La pente de cette fonction augmente si la distance augmente et diminue lorsque la distance diminue. Il est donc nécessaire de connaître en chaque plan l'évolution de

l'erreur pour pouvoir calculer la profondeur avec une erreur ne dépassant pas l'erreur maximale souhaitée.

L'évolution de l'erreur est cruciale car c'est grâce à elle qu'il sera possible de déterminer l'emplacement des plans de référence.

Pour répondre à cette problématique, une approche en 3 étapes est élaborée. Elle consiste à :

1. Estimer la variation de l'erreur en fonction de la distance plan de référence - objet ;
2. Calculer la distance maximale d'analyse ;
3. Estimer l'emplacement des plans de références.

2.3.2.1 Estimation de la variation de l'erreur par rapport à la distance plan de référence - objet

La première étape est l'estimation de la variation de l'erreur par rapport à la distance plan de référence - objet. Cinq plans positionnés entre les bornes fixées sont choisis pour estimer la variance de l'erreur. Les distances choisies sont les deux extrémités, soit 4250 et 5750 mm, puis les valeurs 4575, 5000 et 5325 mm. Ces plans doivent être répartis le plus uniformément possible sur l'intervalle d'analyse, pour pouvoir prédire avec le moins d'erreur possible la variation de l'erreur. Pour réduire encore plus cette erreur de variation, il possible d'estimer celle-ci grâce à un nombre plus important. Cependant, cela engendre du temps d'acquisition et d'analyse plus important.

2.3.2.2 Calcul de la distance maximale d'analyse

La deuxième étape est l'estimation de l'intervalle pour que l'erreur de toutes les données soit inférieur à l'erreur maximale voulue. Pour cela, en chaque plan, un solveur de l'équation cubique est créé. Il prend en entrée une erreur et fournit en sortie la distance liée à cette valeur sur la fonction cubique. Le solveur utilise la méthode de Cardan pour résoudre l'équation. Les données présentent une variance forte. Pour obtenir un intervalle où aucune donnée ne soit supérieur à l'erreur voulue, les données ayant leur erreur supérieure ou égale à cette valeur sont sélectionnées. L'intervalle la plus petite de ces données est l'intervalle final utilisé pour trouver l'emplacement du nouveau plan.

2.3.2.3 Estimation des emplacements des plans de références

La troisième étape permet de généraliser cette approche et donc de calculer l'emplacement des plans de références nécessaires pour la méthode Bi-frequency. Pour cela, la distance d'analyse maximale est calculée pour les 5 plans définis précédemment. Ces 5 plans vont servir à définir une courbe cubique permettant de représenter l'évolution de l'intervalle de distance maximale en fonction de la distance caméra - plan de référence. Les plans de référence sont ensuite définis en partant d'une des bornes et en estimant par itération la prochaine position du plan de référence grâce à l'intervalle du plan précédent. Cet intervalle est celui où aucune donnée n'a son erreur correspondante en dessus de l'erreur de mesure maximale.

Un nombre important de plans est défini et ils seront acquis lors du calibrage pour que la méthode fonctionne et délivre une erreur de mesure faible, c'est à dire ne dépassant pas l'erreur maximale souhaitée, garantie en chaque pixel.

La Figure 2.36 montre les intervalles des 5 plans (bleu) utilisés pour définir les coordonnées en Z des plans de référence pour la méthode Bi-Frequency (rouge).

2.3.3 Acquisition des plans de référence

La phase d'acquisition des plans de références est une étape importante et complexe à mettre en place au niveau pratique.

La phase d'acquisition consiste à acquérir des plans de référence réels, i.e. acquises par le système de vision. Il est nécessaire d'acquérir des plans de référence réels avec le système car c'est grâce à cela que la méthode de décalage de phase permet de s'affranchir des paramètres intrinsèques et extrinsèques et de la matrice de déformation du système ainsi que de ces potentielles erreurs d'estimation lors du calibrage.

Pour garantir l'erreur de mesure voulue (ici 2 mm), il faut que :

— le plan soit perpendiculaire au système ;

— le système et le plan doivent être espacés d'une distance définie avec une erreur de placement faible (définie à 1 mm au maximum dans notre exemple).

Étant donné que l'emplacement des plans de références doit avoir une erreur de placement faible (inférieure à 1 mm) , il est nécessaire d'avoir une méthode pour estimer la distance du système au plan réel et de corriger le système si celui-ci n'est pas perpendiculaire au plan et n'est pas à la bonne distance.

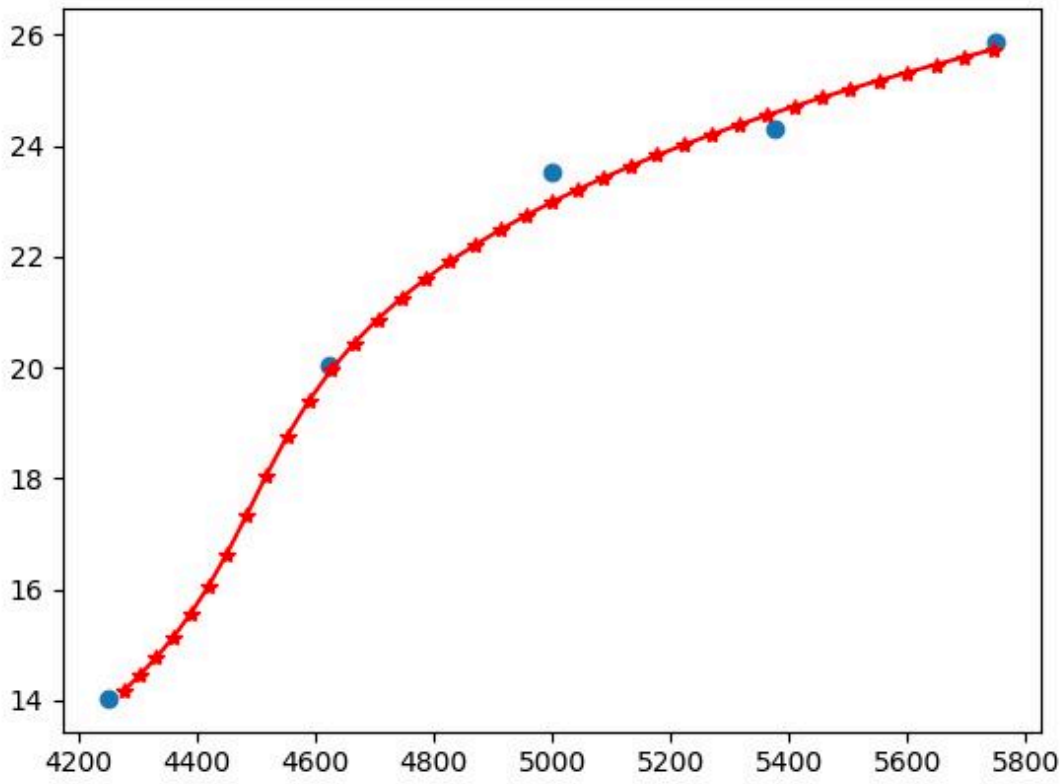


FIGURE 2.36 – Graphique représentant en bleu les distances maximales pour les 5 plans calculés précédemment et en rouge tous les plans nécessaires pour que la méthode puisse estimer la profondeur avec une erreur inférieure à l'erreur maximale (ici 2 mm). L'axe verticale représente le décalage maximale calculée pour les points bleu pour obtenir au maximum l'erreur de mesure souhaitée

Pour estimer si le système est bien placé, des bancs d'essai et du matériel de métrologie peuvent être utilisés. Cela permet d'avoir des résultats avec une erreur très faible, c'est à dire de l'ordre du millimètre ou du micromètre, mais ces équipements ne sont pas facilement disponibles et accessibles. Une méthode utilisant l'algorithme Gray Code + Phase Shifting est proposée pour s'affranchir de ces problématiques.

L'algorithme Gray Code + Phase Shifting est peu sensible au changement de profondeur (Tableau 2.4). Cependant, la variance est importante mais elle semble la même dans chaque carré de la grille perçue dans l'erreur d'estimation de l'algorithme Gray Code + Phase Shifting (Figure). L'idée consiste à :

1. estimer l'angle du plan ;
2. estimer la variation de la moyenne des erreurs en fonction de la distance.

2.3.3.1 Estimation de l'angle du plan

Le plan à acquérir n'est pas seulement codé sur un carré de la grille mais sur un ensemble de carrés. Il est donc possible de trouver un plan permettant de représenter le plan original sans la variance issue de la méthode (Figure 2.37).

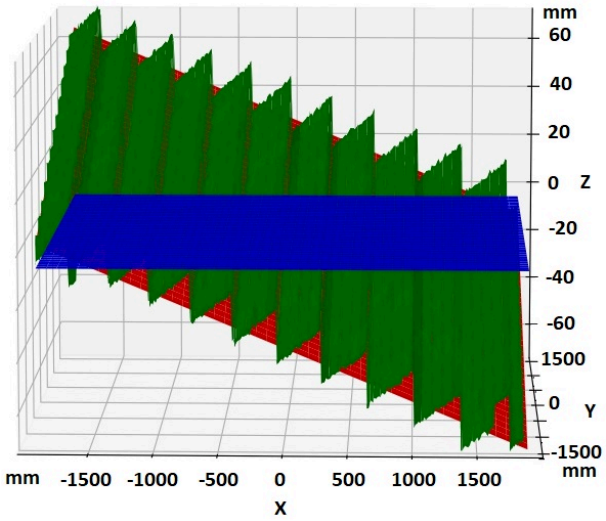


FIGURE 2.37 – Image représentant les différentes données utilisées dans la régression. Le plan est le plan initial, les données représentent l'estimation de la profondeur avec la méthode Gray Code + Phase Shifting. Le plan en rouge représente le plan en sortie de la régression.

Pour cela, une régression est utilisée pour estimer la matrice de rotation permettant de passer du plan que l'on souhaite obtenir à celui qui représente au mieux la sortie de Gray Code + Phase Shifting (Figure 2.38). La régression utilise le GPU ainsi que des librairies spécialisées dans les méthodes utilisant le propagation inverse pour estimer cette matrice.

2.3.3.2 Estimation de la variation de la moyenne des erreurs de profondeur

Avec ce calcul de la matrice de rotation du plan, le problème posé par la variance est fortement atténué. De plus, grâce à cela, il est possible de calculer facilement la différence entre le plan de sortie de la méthode de régression et le plan originel.

Plusieurs acquisitions fictives, par exemple 3, sont réalisées entre les deux bornes limitant la zone d'analyse. Comme dans le chapitre précédent, ces mesures serviront à prédire l'erreur de distance moyenne et ainsi compenser ce décalage.

Au final, la méthode Gray Code + Phase Shifting utilise le calibrage pour estimer

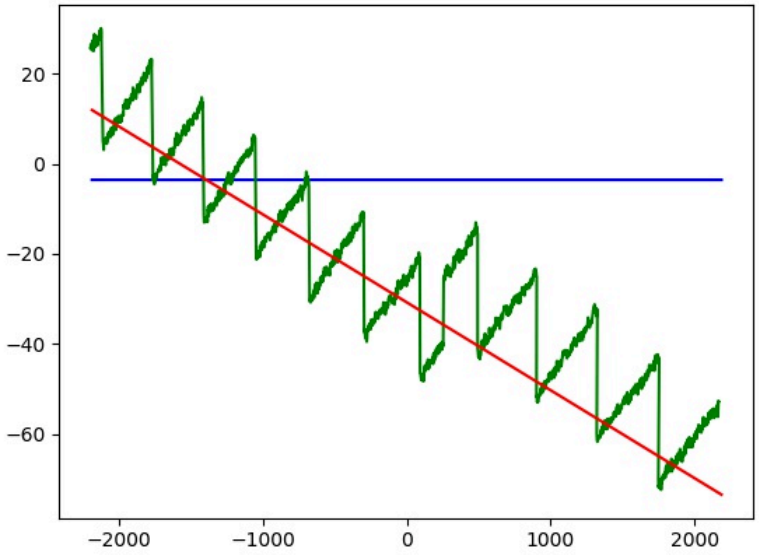


FIGURE 2.38 – Résultat de la régression pour une ligne de l'image initiale. En bleu le plan notre vérité terrain et qui sert de référence pour calculer la rotation, en vert la sortie de la méthode GC+PS et en rouge le plan issu de la régression qui représente les données de la méthode précédente mais sur un plan

la rotation du système par rapport au plan permettant d'acquérir les plans de référence. De plus, elle permet d'estimer la distance du système au plan de référence pour pouvoir corriger celui-ci. Une étude menée grâce à de la simulation permet de corriger le biais de moyenne issue de la méthode Gray Code + Phase Shifting.

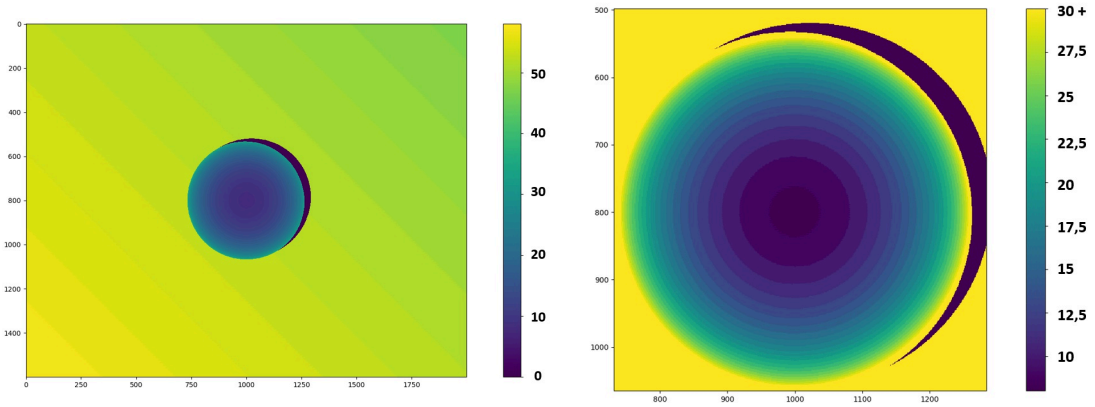
2.3.4 Estimation de la profondeur dans une scène

L'algorithme Bi-frequency est utilisé pour estimer la profondeur dans une scène. Plusieurs plans de références sont ainsi nécessaires.

Pour obtenir la différence de phase finale, i.e. grâce aux images des plans de références et de la scène, deux étapes doivent être réalisées : la première étape est le calcul de la phase enroulée de l'image grâce aux images, acquises par la caméra, des motifs projetés sur les plans de référence et les objets. La seconde étape est le déroulement de la phase enroulée calculée précédemment.

Pour accélérer le processus, les phases déroulées des plans de références peuvent être calculées et sauvegardées après les avoir acquis pendant la phase de calibrage. Ainsi, il est seulement nécessaire de charger ces matrices pour conserver l'ensemble des plans prêts pour l'estimation des coordonnées 3D de l'objet.

La différence de phase entre la phase déroulée du plan de référence et celle de la scène est calculée. Cependant, comme plusieurs plans de référence sont utilisés, il est nécessaire de réaliser cette différence de phase (plan et objet) pour chaque plan. L'index du plan ayant la différence de phase minimale est retenu (Figure 2.39) et permet, par la suite, de connaître en chaque pixel la différence de phase, ainsi que les coordonnées en 3 dimensions des plans de références.



(a) Index des plans

(b) Index des plans centré sur la sphère

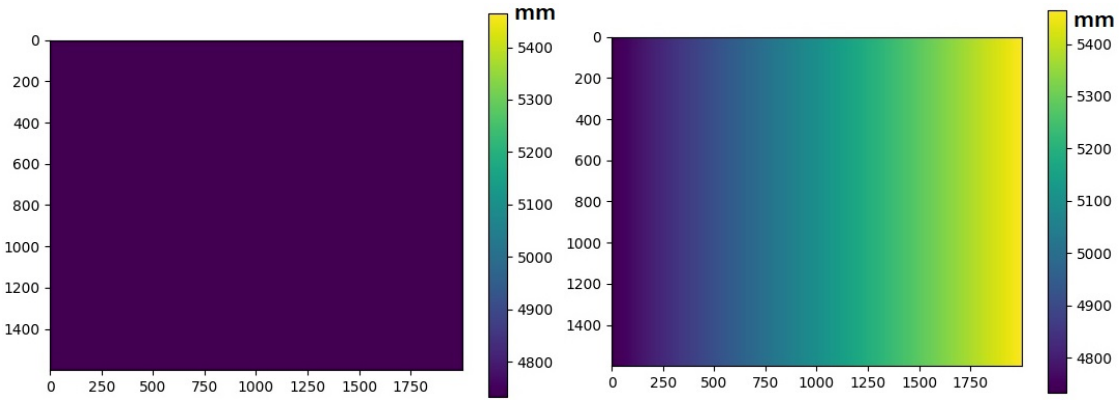
FIGURE 2.39 – Index des plans permettant de calculer la scène

Pour accélérer le processus, il est possible de réaliser une première estimation de la profondeur de la scène grâce à la méthode Gray Code+ Phase Shifting pour pouvoir limiter le nombre de plans de références qui vont être analysés dans la scène. Si les plans de références sont situés avant ou après les objets les plus avancés ou reculés de la scène, ils ne seront pas analysés.

Pour obtenir la profondeur relative de la scène, grâce notamment à la différence de phase, la longueur d'onde du signal en chaque pixel est nécessaire. Elle est constante dans l'image projetée par le projecteur. Cependant, le projecteur est décalé de la caméra. Cela implique que la longueur d'onde sur l'objet n'est pas constante. Pour pouvoir estimer la longueur d'onde réelle, c'est-à-dire sur l'objet, 3 étapes sont nécessaires.

Tout d'abord, un changement de référentiel est opéré pour passer les points 3D de la référence caméra à une référence projecteur (Figure 2.40), par des rotations et translations comme décrit dans l'équation 2.17.

$$coord3Dprojecteur = Rotation_{XYZ}(Translation_{XYZ}(coord3D_{Camera})) \quad (2.17)$$



(a) Profondeur de la scène perçue par la caméra

(b) Profondeur de la scène perçue par le projecteur

FIGURE 2.40 – Profondeur de la scène perçue par la caméra et le projecteur

La deuxième étape est l'estimation de l'angle du champ de vision (Equation 2.18 , également appelé FOV (Field Of View). Cet angle représente l'ouverture de la lentille. Il est donc directement lié à la focale ainsi qu'à la taille du capteur de la caméra.

$$FOV = 2 * \arctan(\frac{resolutionX_{Projecteur}}{2 * focale_{Projecteur}}) \tag{2.18}$$

Avec :

— FOV : l'angle du champ de vision ;

— $resolutionX_{Projecteur}$: le nombre de pixels que peut projeter le projecteur ;

— $focale_{Projecteur}$: la focale de la lentille du projecteur.

Enfin la longueur d'onde, représentée dans la Figure 2.41 est calculée grâce à l'angle du champ de vision, aux coordonnées 3D vues par le projecteur et à la fréquence utilisée pour les motifs (Équation 2.19)

$$\lambda = \frac{coord3D_{Projecteur} * 2 * \sin(FOV/2)}{frquence} \tag{2.19}$$

Avec :

— FOV : l'angle du champ de vision ;

— λ : la longueur d'onde ;

— $coord3D_{Projecteur}$: les coordonnées 3D vues par le projecteur ;

— fréquence : le nombre de période contenues dans le motif projeté.

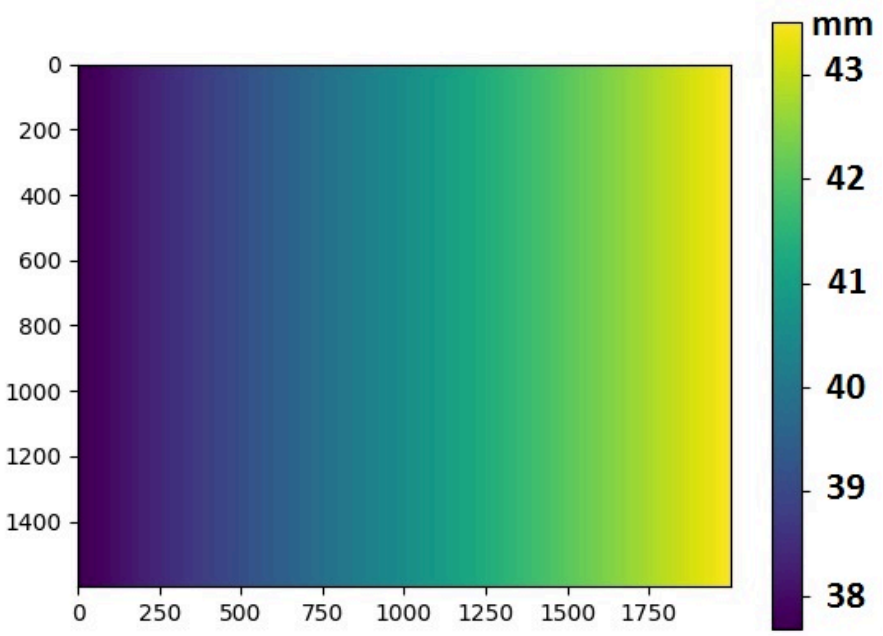


FIGURE 2.41 – Longueur d'onde estimée

Enfin, pour obtenir les coordonnées 3D de la scène, il est nécessaire d'ajouter les coordonnées 3D des plans de référence à la profondeur calculée par la méthode Bi-frequency.

En sortie, les coordonnées ne sont plus par rapport au plan de référence mais par rapport au repère monde situé sur la caméra (Figure 2.42).

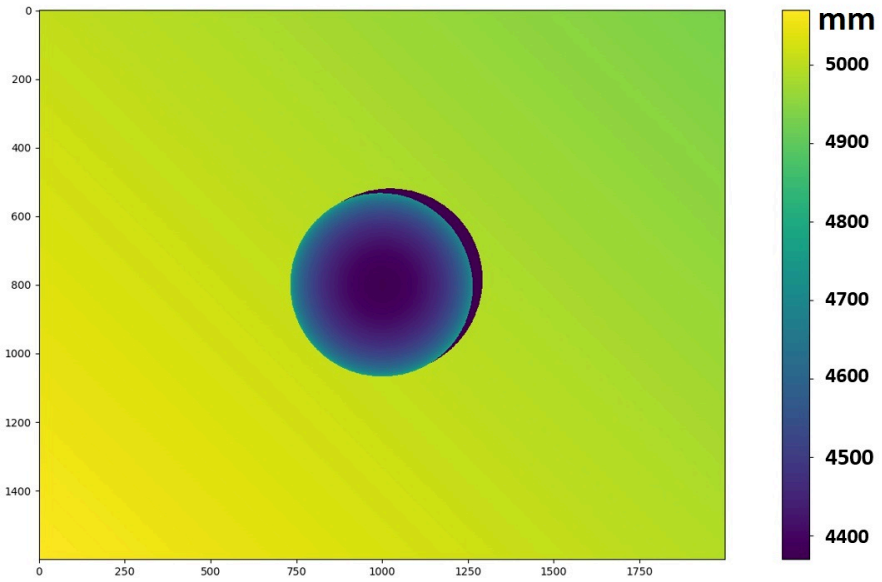


FIGURE 2.42 – Profondeur finale obtenue en sortie de la méthode

2.4 Analyse de la méthode proposée

Dans le processus, l'erreur de mesure du résultat final peut être impactée par l'estimation de l'angle du plan ainsi que l'estimation de la profondeur, mais aussi par l'estimation des coordonnées opérée grâce à la méthode proposée combinant la émthode Bi-frequency et Gray Code + Phase Shifting. Bien que le cahier des charges spécifie une erreur de mesure de l'ordre du millimètre, l'erreur de mesure utilisée pour paramétrer les plans ainsi que pour valider la méthode proposée combinant la méthode Bi-frequency et Gray-Code + Phase shifting est fixé à 2 mm. Le fait d'augmenter cette erreur permet de réduire le nombre de plan à acquérir ainsi que la taille des images de la scène et des plans de références, et ainsi de diminuer le temps de traitement de la méthode .

2.4.1 Estimations de l'angle et de l'emplacement des plans réel

Deux éléments sont analyser :

— l'erreur d'estimation de l'angle ;

— l'erreur d'estimation du placement du plan.

L'angle est calculé directement grâce à la régression. Il est donc possible de comparer l'angle du plan défini avant acquisition en simulation. Pour que la méthode d'estimation de l'angle soit plus robuste, les résultats sont analysés pour des plans positionnés à plus ou moins grande distance de la caméra.

TABLE 2.6 – Erreur d'estimation des angles de rotation pour le plan proche des données GC+PS

Distance (mm)	**Angle X (°)**	**Angle Y (°)**	**Angle Z (°)**
4250	1,140	-1,140	0
5000	1,150	-1,154	0
5750	1,144	- 1,119	0
Vérité terrain	1,145	-1,145	0

Il est possible de remarquer que, grâce au Tableau 2.6 , quelle que soit la profondeur de l'objet, l'estimation de l'angle du plan est très proche de la vérité terrain avec une erreur inférieure à 0,03 degré. Avec cette erreur de rotation, l'erreur en Z sur les extrémités du plan est égale à 0,9 mm se qui est inférieur à l'erreur maximale voulue qui est de 1 mm. L'erreur de profondeur issue de l'erreur d'estimation d'angle peut être visualisée sur la Figure 2.43

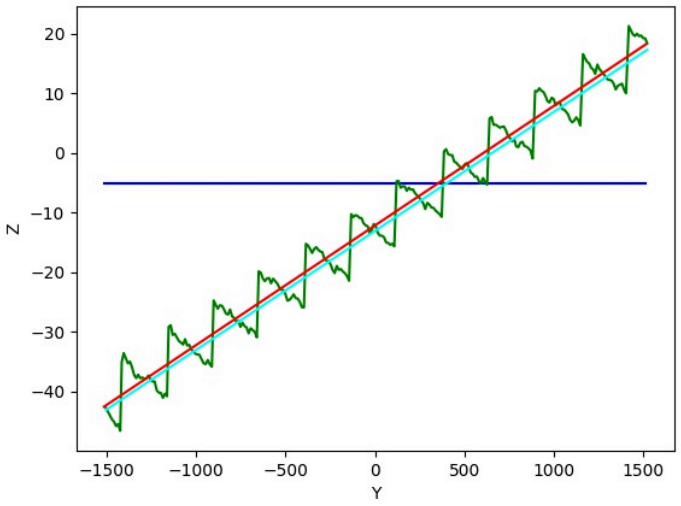


FIGURE 2.43 – Erreur de profondeur issue de l'erreur d'estimation de la rotation du plan

TABLE 2.7 – Erreur d'estimation de la translation sur l'axe Z des plans issus des données GC+PS (brut et avec le plan issu de la régression)

Distance	**Erreur entre la sortie GC+PS et la vérité terrain (mm)**	**Erreur entre le plan estimé et la vérité terrain (mm)**
4250	2,633	0,867
5000	3,977	0,844
5750	2,548	0,820

Enfin, l'erreur de translation est estimée pour déterminer les modifications que l'utilisateur doit réaliser pour être au plus près du plan à acquérir.

Dans la Figure 2.43, il est possible de remarquer que l'erreur commise par la sortie de la méthode GC+PS est importante, avec une erreur supérieure à 2,5 mm, alors que l'erreur d'estimation de la translation pour le plan calculé grâce à la régression est inférieur à 0,9 mm . Pour estimer la translation nécessaire pour que le plan estimé soit superposé au plan original, il faut ajouter au plan une constante égale à 0,84 (Tableau 2.7). Cette constante varie cependant légèrement en fonction de la distance entre le plan de référence et la caméra. Plus le plan est près de la caméra, plus l'erreur est faible. La différence de la constante entre les deux extrémités de la zone est de 0,04 mm, ce qui est très en faible. Cependant, pour être plus robuste, une étude de l'erreur moyenne à ajouter en fonction de la distance peut être réalisée.

Cette erreur est dûe à l'estimation de la profondeur grâce à la méthode Gray-Code + Phase Shifting. Pour pouvoir réduire cette erreur et ainsi réduire l'erreur de mesure de l'angle du plan et de sa translation, il est possible d'augmenter la résolution de la caméra et du projecteur, ainsi qu'augmenter l'écartement du projecteur par rapport à la caméra.

TABLE 2.8 – Erreur de la méthode proposée (Bi-F + GC PS) sur la sphère et le plan incliné de la scène

	Axe	Moyenne (mm)	Écart-Type (mm)
Plan	X	-3,29E-05	1,88E-02
	Y	5,01E-05	1,89E-02
	Z	4,98E-03	3,73E-01
Sphère	X	-2,09E-04	7,98E-02
	Y	2,43E-04	6,38E-02
	Z	2,26E-03	3,47E-01

Cependant, la dernière solution peut impacter les résultats car il y aura plus d'ombre sur la scène du fait du décalage.

La méthode d'estimation du plan pour l'acquisition de celui-ci permet de donner à l'utilisateur les valeurs des angles et translations à appliquer pour que le système soit perpendiculaire et situé à la distance voulue au dixième de millimètre près avec au maximum une erreur maximale inférieur à 1 mm.

2.4.2 Estimations de la profondeur grâce à la méthode proposée

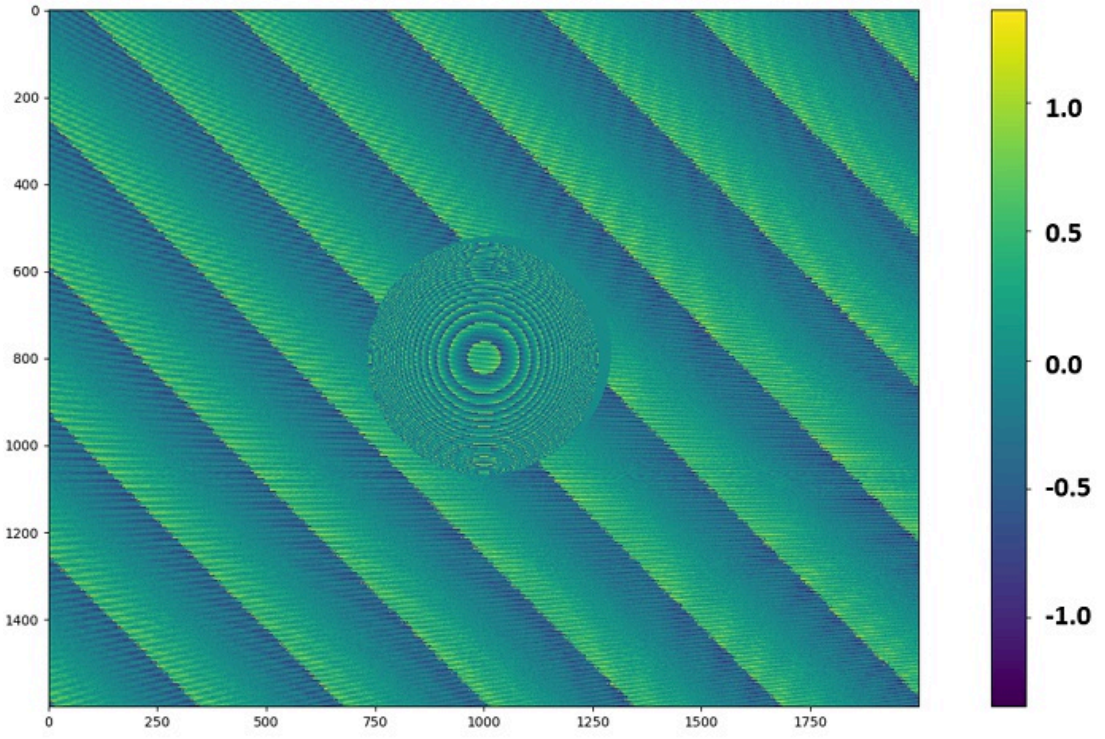
Pour s'assurer que la méthode permette d'améliorer le processus, les plans d'acquisition sont générés grâce à la simulation. Ils sont donc parfaits.

En Figure 2.44 l'erreur sur l'axe Z reste faible et comprise entre 2 et -2 mm. L'erreur maximale souhaitée ne dépasse pas l'erreur souhaitée, en chaque pixel, quelle que soit la distance de l'objet et ce selon les 3 axes.

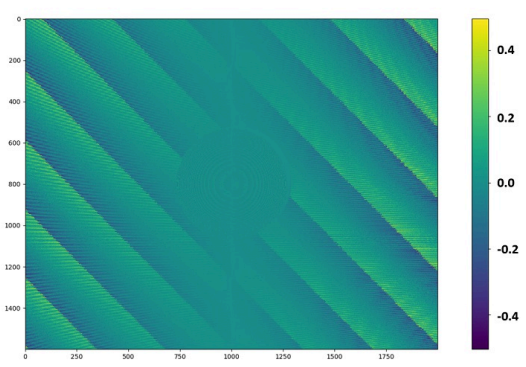
Ainsi, la méthode proposée combinant la méthode Bi-frequency et Gray Code + Phase Shifting permet d'obtenir des résultats avec des erreurs de mesure très faibles pour le plan comme pour la sphère (Tableau 2.8). En effet, les moyennes et variances sur les 3 axes des deux exemples sont en-dessous de 0,2 mm.

Par rapport aux résultats de la méthode Bi-frequency avec un seul plan, la moyenne des erreurs a été divisée par 2000 et la variance des erreurs absolues par 400, notamment sur les objets loin du plan de référence (sphère).

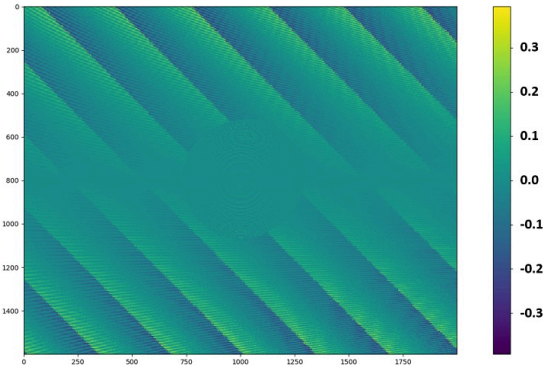
La méthode proposée diminue donc grandement l'erreur d'estimation des coordonnées 3D de la scène, et donc de la profondeur de celle-ci. De plus, elle peut fonctionner pour des erreurs de mesure différentes et très faibles. En effet, elle s'auto-calibre en fonction de ce paramètre.



(a) Erreur de l'estimation des coordonnées sur l'axe Z



(b) Erreur de l'estimation des coordonnées sur l'axe X



(c) Erreur de l'estimation des coordonnées sur l'axe Y

FIGURE 2.44 – Erreur de la méthode sur les axes X,Y et Z. La scène de test est composée d'un plan incliné et d'une sphère

2.5 Conclusion

Les travaux réalisés et présentés précédemment, montrent que la méthode proposée, c'est-à-dire la combinaison de la méthode Bi-Frequency et Gray-Code+Phase Shifting, apporte une réelle amélioration dans l'estimation des coordonnées de la scène en chaque pixel. En effet, elle permet d'obtenir une répartition homogène et dense des points de

mesure et une erreur minimale de mesure obtenue respectant l'erreur minimale souhaitée.

La méthode proposée utilise la méthode Bi-Frequency avec plusieurs plans de référence pour pouvoir estimer avec une faible erreur les coordonnées 3D. La méthode Gray-Code + Phase Shifting est utilisée pour acquérir les différents plans de références et pour limiter le nombre de plans utilisé lors de l'estimation des coordonnées 3D de la scène. Cette méthode permet de lever la problématique concernant le placement des plans de références avec une erreur faible (en dessous du millimètre) et la rotation du système pour que celui-ci puisse être perpendiculaire à ces plans (erreur en dessous du dixième de degré).

Cette méthode a été testée sur des objets non-réfléchissants et de synthèse. Cela a permis de s'affranchir de plusieurs contraintes notamment des erreurs de calibrage ou d'erreur des objets de références (plans et sphère). Plusieurs points d'attention sont donc à prendre en compte. Tout d'abord, il est donc important de prendre en compte les erreurs calibrage lors d'acquisition réelle, car elles peuvent dégrader les résultats finaux. Lors de l'acquisition des plans de références, l'environnement doit être contrôlé notamment au niveau des systèmes permettant de décaler et d'exercer une rotation sur le système de vision pour que les plans de références soient acquis avec une erreur de placement et de rotation très faible. La méthode peut être améliorée et plus robuste, notamment au niveau des erreurs liés aux réflexions spéculaires et à la luminosité de l'environnement, en ajoutant une seconde caméra.

BIBLIOGRAPHIE

[1] Yatong AN et al. “Method for large-range structured light system calibration”. In : *Appl. Opt.* 55.33 (2016), p. 9563-9572. DOI : 10.1364/AO.55.009563. URL : http://opg.optica.org/ao/abstract.cfm?URI=ao-55-33-9563.

[2] O. V. ANGELSKY et al. “Structured Light Control and Diagnostics Using Optical Crystals”. In : *Frontiers in Physics* 9 (2021). ISSN : 2296-424X. DOI : 10.3389/fphy.2021.715045. URL : https://www.frontiersin.org/article/10.3389/fphy.2021.715045.

[3] Cyrus BAMJI et Peiqian ZHAO. “Single chip red, green, blue, distance (RGB-Z) sensor”. en. Brev. US8139141B2. Mar. 2012. URL : https://patents.google.com/patent/US8139141/en (visité le 20/07/2022).

[4] Jianhui CHEN, Karim BENZEROUAL et Robert S. ALLISON. “Calibration for high-definition camera rigs with marker chessboard”. In : *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 2012, p. 29-36. DOI : 10.1109/CVPRW.2012.6238905.

[5] Rui CHEN, Jing XU et Song ZHANG. “Comparative study on 3D optical sensors for short range applications”. In : *Optics and Lasers in Engineering* 149 (2022), p. 106763. ISSN : 0143-8166. DOI : https://doi.org/10.1016/j.optlaseng.2021.106763. URL : https://www.sciencedirect.com/science/article/pii/S0143816621002335.

[6] Danick DESJARDINS et Pierre PAYEUR. “Dense Stereo Range Sensing with Marching Pseudo-Random Patterns”. In : *Fourth Canadian Conference on Computer and Robot Vision (CRV ’07)*. 2007, p. 216-226. DOI : 10.1109/CRV.2007.22.

[7] Andreas DIETRICH et al. “Realtime Ray Tracing for Advanced Visualization in the Aerospace Industry”. In : (2006).

[8] Jason GENG. “Structured-light 3D surface imaging: a tutorial”. In : *Adv. Opt. Photon.* 3.2 (2011), p. 128-160. DOI : 10.1364/AOP.3.000128. URL : http://opg.optica.org/aop/abstract.cfm?URI=aop-3-2-128.

[9] Dennis C. GHIGLIA et Louis A. ROMERO. “Minimum Lp-norm two-dimensional phase unwrapping”. In : *J. Opt. Soc. Am. A* 13.10 (1996), p. 1999-2013. DOI : 10.1364/JOSAA.13.001999. URL : http://opg.optica.org/josaa/abstract.cfm?URI=josaa-13-10-1999.

[10] Alan A GOLDSTEIN. “Convex programming in Hilbert space”. In : *Bulletin of the American Mathematical Society* 70.5 (1964), p. 709-710.

[11] Jens GUEHRING. “Dense 3D surface acquisition by structured light using off-the-shelf components”. In : (2000).

[12] Kyriakos HERAKLEOUS et Charalambos POULLIS. “3DUNDERWORLD-SLS: An Open-Source Structured-Light Scanning System for Rapid Geometry Acquisition”. In : *CoRR* abs/1406.6595 (2014). arXiv : 1406.6595. URL : http://arxiv.org/abs/1406.6595.

[13] Eli HORN et Nahum KIRYATI. “Toward optimal structured light patterns”. In : *Proceedings. International Conference on Recent Advances in 3-D Digital Imaging and Modeling (Cat. No.97TB100134)* (1997), p. 28-35.

[14] Danying HU, Daniel DETONE et Tomasz MALISIEWICZ. “Deep ChArUco: Dark ChArUco Marker Pose Estimation”. In : juin 2019, p. 8428-8436. DOI : 10.1109/CVPR.2019.00863.

[15] Nicolas HURTUBISE. “Introduction à la reconstruction 3D par lumière structurée”. fra. In : (juil. 2021). Accepted: 2021-08-25T12:47:56Z. URL : https://papyrus.bib.umontreal.ca/xmlui/handle/1866/25466 (visité le 11/05/2022).

[16] Heinz HÜGLI et Gilbert MAÎTRE. “Generation And Use Of Color Pseudo Random Sequences For Coding Structured Light In Active Ranging”. In : *Proceedings of SPIE - The International Society for Optical Engineering* 1010 (fév. 1989). DOI : 10.1117/12.949215.

[17] Wonjoo KIM et al. “A 1.5Mpixel RGBZ CMOS image sensor for simultaneous color and range image capture”. In : *2012 IEEE International Solid-State Circuits Conference*. 2012, p. 392-394. DOI : 10.1109/ISSCC.2012.6177061.

[18] J.J. LE MOIGNE et A.M. WAXMAN. “Structured light patterns for robot mobility”. In : *IEEE Journal on Robotics and Automation* 4.5 (1988), p. 541-548. ISSN : 2374-8710. DOI : 10.1109/56.20439.

[19] Yongzeng LI et al. “Simultaneous 3D surface profile and pressure measurement using phase-shift profilometry and pressure-sensitive paint”. In : *Review of Scientific Instruments* 92 (mar. 2021), p. 035107. DOI : 10.1063/5.0031036.

[20] Lukas LIPP. “Real-Time Ray Tracing”. In : (2020).

[21] Kai LIU et al. “Dual-frequency pattern scheme for high-speed 3-D shape measurement”. In : *Opt. Express* 18.5 (2010), p. 5229-5244. DOI : 10.1364/OE.18.005229. URL : http://opg.optica.org/oe/abstract.cfm?URI=oe-18-5-5229.

[22] Lilika MAKABE et al. “Shape-coded ArUco: Fiducial Marker for Bridging 2D and 3D Modalities”. In : *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2022, p. 2315-2324. DOI : 10.1109/WACV51458.2022.00237.

[23] Donald W. MARQUARDT. “An algorithm for least-squares estimation of nonlinear parameters”. In : *SIAM Journal on Applied Mathematics* 11.2 (1963), p. 431-441. DOI : 10.1137/0111030. URL : http://dx.doi.org/10.1137/0111030.

[24] Andres G. MARRUGO et al. “Method for large-scale structured-light system calibration”. In : *Opt. Express* 29.11 (2021), p. 17316-17329. DOI : 10.1364/OE.422327. URL : http://opg.optica.org/oe/abstract.cfm?URI=oe-29-11-17316.

[25] R.A. MORANO et al. “Structured light using pseudorandom codes”. In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.3 (1998), p. 322-327. DOI : 10.1109/34.667888.

[26] Johann NÜSS et al. “Effects of non-ideal display properties in phase measuring deflectometry: A model-based investigation”. In : (mai 2018), p. 35. DOI : 10.1117/12.2306463.

[27] *Objectifs adéquats pour une vision industrielle de haute précision*. fr_FR. URL : https://www.stemmer-imaging.com/fr-fr/conseil-techniqu/objectifs-adequats-pour-une-vision-industrielle-de-haute-precision/ (visité le 15/05/2022).

[28] J.L POSDAMER et M.D ALTSCHULER. “Surface measurement by space-encoded projected beam systems”. In : *Computer Graphics and Image Processing* 18.1 (1982), p. 1-17. ISSN : 0146-664X. DOI : https://doi.org/10.1016/0146-664X(82)90096-X. URL : https://www.sciencedirect.com/science/article/pii/0146664X8290096X.

[29] J QUIROGA. “Structured light-based NDT techniques of interest in the aerospace industry”. In : *Insight* 45 (jan. 2003), p. 280-285. DOI : 10.1784/insi.45.4.280.52841.

[30] Valentin REBIERE et al. “Semi-Gradient for Color Pixel Reconstruction in a RGBZ CMOS Sensor”. In : *2020 IEEE SENSORS*. 2020, p. 1-4. DOI : 10.1109/SENSORS47125.2020.9278887.

[31] Joan R. ROSELL-POLO et al. “Chapter Three - Advances in Structured Light Sensors Applications in Precision Agriculture and Livestock Farming”. In : sous la dir. de Donald L. SPARKS. T. 133. Advances in Agronomy. Academic Press, 2015, p. 71-112. DOI : https://doi.org/10.1016/bs.agron.2015.05.002. URL : https://www.sciencedirect.com/science/article/pii/S0065211315001078.

[32] J. SALVI, J. BATLLE et E. MOUADDIB. “A robust-coded pattern projection for dynamic 3D scene measurement”. In : *Pattern Recognition Letters* 19.11 (1998), p. 1055-1065. ISSN : 0167-8655. DOI : https://doi.org/10.1016/S0167-8655(98)00085-3. URL : https://www.sciencedirect.com/science/article/pii/S0167865598000853.

[33] Joaquim SALVI, Jordi PAGÈS et Joan BATLLE. “Pattern codification strategies in structured light systems”. In : *Pattern Recognit.* 37 (2004), p. 827-849.

[34] Joaquim SALVI, Jordi PAGÈS et Joan BATLLE. “Pattern Codification Strategies in Structured Light Systems”. In : *PATTERN RECOGNITION* 37 (2004), p. 827-849.

[35] Giovanna SANSONI, Matteo CAROCCI et Roberto RODELLA. “Three-dimensional vision based on a combination of gray-code and phase-shift light projection: analysis and compensation of the systematic errors”. In : *Appl. Opt.* 38.31 (1999), p. 6565-6573. DOI : 10.1364/AO.38.006565. URL : http://opg.optica.org/ao/abstract.cfm?URI=ao-38-31-6565.

[36] K. SATO. “Range imaging based on moving pattern light and spatio-temporal matched filter”. In : 1 (1996), 33-36 vol.1. DOI : 10.1109/ICIP.1996.559426.

[37] P. SCHROEDER et al. “Industrial Phase-Shifting Profilometry in Motion”. In : (2013). Sous la dir. de Joni-Kristian KÄMÄRÄINEN et Markus KOSKELA, p. 579-590.

[38] Xianyu SU et Wenjing CHEN. “Fourier transform profilometry:: a review”. In : *Optics and Lasers in Engineering* 35.5 (2001), p. 263-284. ISSN : 0143-8166. DOI : https://doi.org/10.1016/S0143-8166(01)00023-9. URL : https://www.sciencedirect.com/science/article/pii/S0143816601000239.

[39] Natalia SWOJAK, Michał WIECZOROWSKI et Michał JAKUBOWICZ. “Assessment of selected metrological properties of laser triangulation sensors”. In : *Measurement* 176 (2021), p. 109190. ISSN : 0263-2241. DOI : https://doi.org/10.1016/j.measurement.2021.109190. URL : https://www.sciencedirect.com/science/article/pii/S0263224121002086.

[40] Junyi XU et al. “An Efficient Minimum-Discontinuity Phase-Unwrapping Method”. In : *IEEE Geoscience and Remote Sensing Letters* 13.5 (2016), p. 666-670. DOI : 10.1109/LGRS.2016.2535159.

[41] Li ZHANG, Brian CURLESS et Steven M. SEITZ. “Rapid shape acquisition using color structured light and multi-pass dynamic programming”. In : *Proceedings. First International Symposium on 3D Data Processing Visualization and Transmission* (2002), p. 24-36.

[42] Zonghua ZHANG et al. “Comparison of Fourier transform, windowed Fourier transform, and wavelet transform methods for phase calculation at discontinuities in fringe projection profilometry”. In : *Optics and Lasers in Engineering* 50.8 (2012). Fringe Analysis Methods Applications, p. 1152-1160. ISSN : 0143-8166. DOI : https://doi.org/10.1016/j.optlaseng.2012.03.004. URL : https://www.sciencedirect.com/science/article/pii/S0143816612000760.

[43] W.-D ZHU et al. “Calibration of industrial cameras using asymmetric circle center projection”. In : *Guangxue Jingmi Gongcheng/Optics and Precision Engineering* 22 (août 2014), p. 2267-2273. DOI : 10.3788/OPE.20142208.2267.

[44] Chao ZUO et al. “High-speed three-dimensional shape measurement for dynamic scenes using bi-frequency tripolar pulse-width-modulation fringe projection”. In : *Optics and Lasers in Engineering* 51.8 (2013), p. 953-960. ISSN : 0143-8166. DOI : https://doi.org/10.1016/j.optlaseng.2013.02.012. URL : https://www.sciencedirect.com/science/article/pii/S0143816613000754.

[45] Chao ZUO et al. “Phase shifting algorithms for fringe projection profilometry: A review”. In : *Optics and Lasers in Engineering* 109 (mai 2018). ISSN : 0143-8166. DOI : 10.1016/j.optlaseng.2018.04.019. URL : https://www.osti.gov/biblio/1469785.

CHAPITRE 3

CNN-CA : ASSOCIATION DE MÉTHODES PERMETTANT D'AMÉLIORER LA SEGMENTATION D'IMAGES RGB-D

La segmentation d'images est une problématique très présente dans le domaine de la recherche, car elle concerne de nombreuses applications telles que l'imagerie médicale[4], l'agronomie[26], la qualité des produits industriels[9] ou le spatial[29]. La segmentation d'images peut être résumée comme un processus de regroupement de pixels ayant les mêmes caractéristiques. Plusieurs approches ont été proposées, basées soit sur les contours, soit sur les caractéristiques de régions des objets à segmenter.

Dans le cas d'une scène, les objets peuvent être de couleur, de forme, de taille différentes. De plus, certains objets existent dans plusieurs versions qui peuvent être très différentes les unes des autres. S'ajoutant à cela, des éléments extérieurs tels que l'éclairage et le positionnement de l'objet dans la scène font que l'algorithme permettant de les segmenter doit s'adapter et prendre en compte de nombreux changements.

Parmi les méthodes proposées dans la littérature, les réseaux neuronaux, et plus précisément les CNN (Convolutional Neural Network), sont les algorithmes de traitement d'images les plus performants pour ce type de tâche. Cependant, leurs résultats ne permettent pas d'avoir une segmentation sans erreur. L'amélioration des contours peut être réalisée grâce à l'approche par modèles déformables. Les modèles déformables sont flexibles et peuvent être adaptés à différentes problématiques grâce au choix de plusieurs forces paramétrables. Cependant, l'initialisation de ces modèles, qui doit être effectuée par l'utilisateur ou préprogrammée, ainsi que le réglage de ces paramètres sont difficiles. L'une des méthodes de segmentation les plus connues est celle dite des contours actifs.

L'objectif de la solution proposée est de combiner les deux méthodes (CNN et Contours actifs) pour segmenter avec moins d'erreur des objets sur des images RGB-D (couleur et profondeur) car elles permettent une meilleure compréhension de la scène et des objets. Le CNN est utilisé pour segmenter grossièrement l'image et ainsi prédire le contour des objets, tandis que le contour actif améliore ce contour sur le bord de l'objet. Cette solution combine donc une méthode qui prend en compte des données de haut niveau (CNN) avec une méthode qui travaille exclusivement sur des données de bas niveau (Contours actifs).

3.1 Réseaux de neurones convolutifs de segmentation

3.1.1 Principe

Les réseaux de neurones convolutifs (CNN) sont actuellement des algorithmes très développés et explorés du fait de leur capacité à extraire d'une image des caractéristiques non définies au préalable par l'utilisateur et d'extraire des informations haut niveau. C'est notamment le cas des CNN de classification qui prédisent le ou les objets principaux présents dans une image. Pour les CNN de segmentation, tel que la Figure 3.1, les réseaux de neurones extraient des informations haut niveau et tentent de redescendre vers l'information bas niveau pour obtenir à la fin une image segmentée.

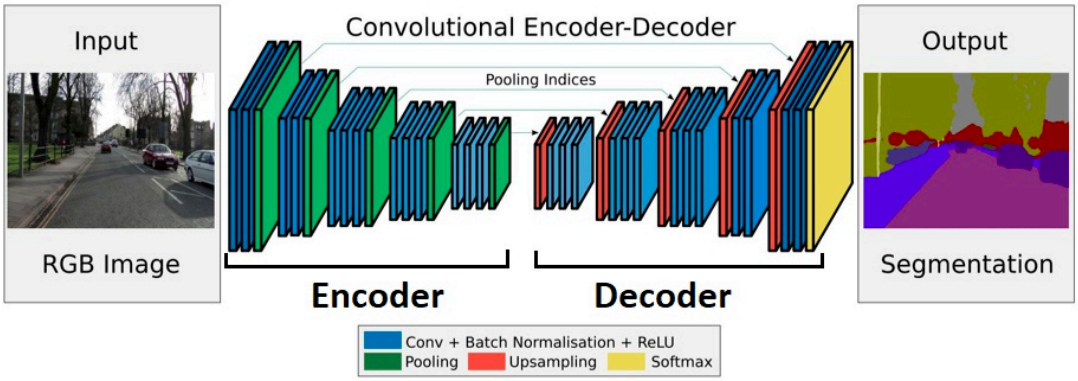


FIGURE 3.1 – CNN de segmentation prenant en entrée des images RGB et générant en sortie une image labélisée [2]

Dans l'image segmentée d'une image originale, chaque pixel contient, à la place d'une information de couleur ou de luminosité, le label indiquant à quel objet il appartient. Par exemple, dans une image d'une scène de la vie courante, le label peut désigner : bus, vélo, maison, humain, chien, etc. Comme tous les algorithmes de machine learning, les CNN doivent être paramétrés avant de pouvoir les utiliser dans l'application voulue, c'est-à-dire lors de l'inférence. La paramétrisation du CNN est composée de 2 étapes :

— le choix du modèle de CNN ;

— l'apprentissage.

3.1.2 Modèle de CNN

Un nombre considérable de modèles existent [20]. Ils sont différentiables par le type et la taille des données d'entrée, le nombre et les types de couches, la fusion de données

(lorsque plusieurs types de données sont fournis en entrée) et leurs sorties.

Les CNN prennent en général en entrée de réseau des images en couleur. Cependant, ils peuvent prendre à la place ou en plus, des données 3D [25], des images de profondeur [11], des images thermiques [22] ou encore des images issues d'une caméra multispectrale [1]. Le choix des données en entrée est opéré par l'expert en fonction du contexte de l'application et de ce qu'il souhaite segmenter. Les données d'entrée doivent apporter une information pertinente et essentielle pour déterminer l'objet et donc bien le segmenter. En fonction de la résolution souhaitée et de la limite du matériel à disposition pour les calculs (GPU), la résolution des couches du modèle et des images en entrée sont adaptées.

Les modèles sont composés d'une multitudes de couches avec, pour certains, plusieurs centaines de couches différentes.

Les premières couches des CNN de segmentation extraient des caractéristiques telles que les bords des objets ou la variation des couleurs. Ces couches sont souvent composées de couches de convolution et d'activation comme la couche ReLu.

Les couches intermédiaires synthétisent les caractéristiques les plus significatives pour pouvoir reconnaître les objets dans l'image et permettent ainsi de passer d'une information bas niveau à haut niveau. Les couches de pooling sont utilisées pour cela.

Les dernières couches, notamment les couches de unpooling et de up-convolution, visent à partitionner les pixels en catégories identifiées dans les couches intermédiaires grâce aux caractéristiques extraites par les premières couches. Elles permettent donc de passer de caractéristiques haut niveau à bas niveau.

La Figure 3.2, montre les caractéristiques extraites d'une voiture dans les différentes couches de la partie encoder d'un CNN de segmentation.

Les couches d'extractions et de synthèses des données sont rassemblées dans une partie appelée encoder. La partie permettant de remonter des données haut niveau à l'image segmentée est appelée decoder.

La profondeur du réseau est complexe à déterminer. Cependant, plus les objets à déterminer sont complexes, plus les réseaux auront tendance à être profonds. De plus, si plusieurs images de différents types de données sont en entrée du réseau, le modèle peut être découpé en plusieurs branches pour pouvoir traiter séparément les deux images. Les données des deux images seront fusionnées dès l'entrée, dans la partie extraction des caractéristiques ou dans la partie decoder du modèle. En fonction des couches où sont

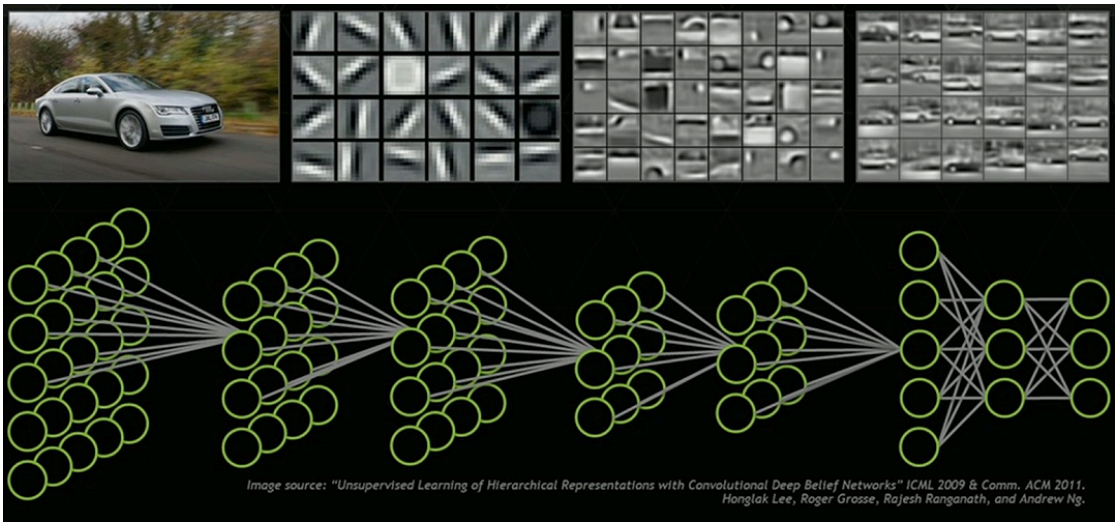


FIGURE 3.2 – Représentation des caractéristiques extraites d'images de voitures par un CNN dans ces différentes couches [21]

fusionnées les données des différentes images, les informations fusionnées permettront d'extraire des caractéristiques différentes de l'objet à analyser.

Dans la majorité des cas, l'image de labélisation est la seule sortie de ces modèles. Cependant, d'autres images de sortie, telle que l'image de profondeur, peuvent être estimées pour améliorer la labélisation issue de ce modèle [14]. Le fait d'estimer des données supplémentaires complexifie le modèle.

Le choix d'un modèle est généralement réalisé en fonction des résultats à l'issue de l'entraînement de celui-ci sur une base de données. Plus la métrique choisie pour analyser la performance et explicitée dans le chapitre 3.1.5 est haute, plus le modèle est performant.

3.1.3 Apprentissage

L'apprentissage est une étape importante dans l'utilisation des CNN, pour que ceux-ci puissent bien segmenter une scène. Le principe est de faire passer dans le réseau, selon un nombre conséquent d'itérations, la base de données complète pour pouvoir corriger à chaque passage les couches pour que celles-ci s'adaptent aux données et permettent de faire de moins en moins d'erreurs de segmentation. Dans un souci de rapidité et de limitation dû au matériel (GPU), les bases de données sont généralement découpées en batch, i.e. des petites portions de la base de données initiale. Ce sont ces batchs qui vont passer dans le réseau et non l'intégralité de la base en une seule fois. La méthode, réalisée à chaque entraînement grâce à un batch, permettant de corriger les paramètres

des couches est la backpropagation. Elle estime l'erreur qu'il faut appliquer à chaque couche du modèle pour par la suite les corriger. Lorsque les paramètres des couches sont corrigés, l'erreur diminue. L'erreur calculée à la fin du modèle (la segmentation est réussie ou non et à quel point) est remontée et calculée en fonction de l'impact des paramètres des couches dans l'erreur finale. L'apprentissage s'arrête lorsque l'erreur commise par le réseau stagne ou est nulle. En effet, si l'erreur stagne, cela signifie que le modèle n'arrive plus à apprendre des différences pour segmenter plus finement les objets. La Figure 3.3 montre que l'image d'entrée passe dans différentes couches (trait bleu) vers les couches permettant de segmenter l'image et que les erreurs calculées dans la dernière couche sont remontées jusque dans la première couche grâce à la backpropagation (trait jaune).

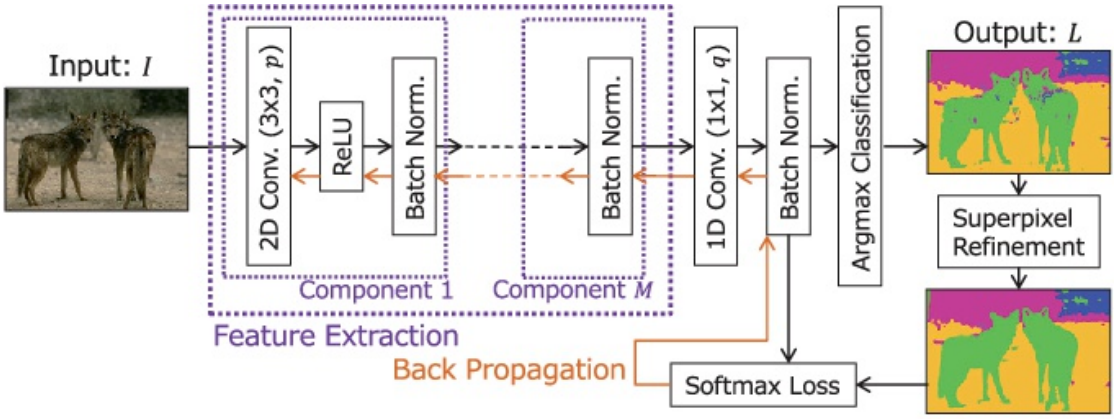


FIGURE 3.3 – CNN de segmentation représentant la segmentation d'une image RGB en image labélisée, ainsi que la backpropagation dans le modèle [15]

L'apprentissage est une étape longue et qui nécessite une base de données conséquente pour être robuste, c'est-à-dire qu'elle permettra de délivrer un résultat proche de la vérité terrain. Or, dans la majorité des cas, les applications disposent d'une base de données réduite, labélisées ou non. Ces applications utilisent généralement du transfert learning pour pouvoir paramétrer leur réseau de neurones. Le transfert learning consiste à mettre en oeuvre un réseau de neurones entraîné ou pré-entrainé sur une autre base de données de grande taille, et de poursuivre l'entraînement des couches sur la base de données spécifiques à l'application. Cela permet de commencer avec des paramètres de couches cohérents mais non spécifiques aux objets de la base de données d'application.

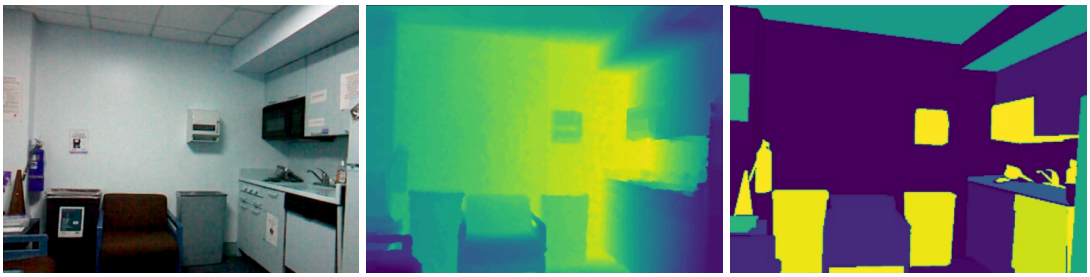
3.1.4 Bases de données

De nombreuses bases de données existent. Cependant, compte tenu des contraintes liées à l'application, il est nécessaire d'avoir une base de données comportant un ensemble

d'images de la même scène mais avec des informations différentes (images de couleur (RGB) + de profondeur + labélisées). Deux bases de données sont actuellement utilisées dans les publications utilisant des données RGB-D : SUNRGB-D et NYUV2.

3.1.4.1 SUNRGBD

La base SUNRGBD [30] contient 10 335 images (RGB + profondeur + labélisées). Plusieurs bases composent celle-ci : NYUV2, Berkeley B3DO et SUN3D. Les bases sont acquises de différentes façons et sont ainsi composées d'images n'ayant pas la même résolution. La base SUNRGBD est composée de 5285 images pour l'entraînement et de 5050 images pour le test. 37 classes sont présentes dans la base.



(a) Image RGB (b) Image Profondeur (c) Image Labélisée

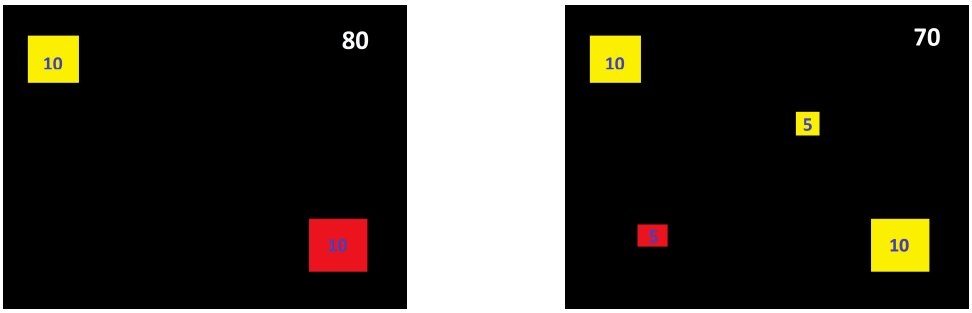
FIGURE 3.4 – Exemple d'une image de la base de données NYUV2. Dans l'image de profondeur, plus la couleur est claire, plus la profondeur est importante. Dans l'image labélisée, chaque label est codé par une couleur spécifique

3.1.4.2 NYUV2

La base NYUV2 [27] est composée de 1449 images (RGB + profondeur + labélisées) de 464 scènes différentes acquises grâce à une Kinect (Figure 3.4). Cette base de données est séparée en deux parties : la partie entraînement contenant 795 images issues de 249 scènes différentes et la partie test contenant 645 images issues de 215 scènes différentes. La résolution des images est de 640x480. 900 classes sont présentes dans la base bien que 40 classes soient gardées en général pour l'analyse des algorithmes de segmentation utilisant cette base de données. En effet, les autres classes sont peu représentées ce qui peut engendrer un déséquilibre important et donc des erreurs sur des méthodes pas assez robustes pour ces types de classes. Seules ces 40 classes seront utilisées dans les tests présentés ci-après.

3.1.5 Métriques

La performance des méthodes de segmentation est mesurée grâce à plusieurs types de métriques. Celles-ci sont présentées ci-dessous. Pour pouvoir les comparer et montrer leurs différences au niveau des résultats, la Figure 3.5 est prise comme exemple de référence.



(a) Image Vérité Terrain (b) Image Prédite

FIGURE 3.5 – Exemple d'une image labélisée représentant la vérité terrain (a), et d'une image segmentée par un modèle (prédite) (b). Ces images sont constituées de 3 types d'objet : le fond en noir, un ou plusieurs objets jaunes et un objet rouge. Les chiffres représentent le nombre de pixels contenus dans chaque composante connexe

3.1.5.1 Matrice de confusion

La matrice de confusion est un résumé des résultats de classification d'une méthode. Elle calcule le résultat pour une classe particulière et permet de visualiser la répartition des données. La donnée de base appartient ou non à la classe considérée et la prédiction est ou n'est pas la classe. Quatre situations sont ainsi définies et le nombre de cas correspondant à chacune sont calculés :

— Vrai Positif (VP) : la prédiction et la vérité terrain sont égales à la classe ;

— Vrai Négatif (VN) : ni la prédiction ni la vérité terrain ne sont égales à la classe ;

— Faux Positif (FP) : la prédiction est égale à la classe mais la vérité terrain n'est pas égale à la classe ;

— Faux Négatif (FN) : la prédiction n'est pas égale à la classe mais la vérité terrain est égale à la classe.

Cette métrique, généralement représentée comme dans la Figure 3.6, analyse une seule classe à la fois ce qui ne permet pas d'avoir un résultat global de la méthode sur

	Prédiction **Classe**	Prédiction **Non Classe**
Vérité Terrain **Classe**	Vrai Positif (VP)	Faux Négatif (FN)
Vérité Terrain **Non Classe**	Faux Positif (FP)	Vrai Négatif (VN)

FIGURE 3.6 – Matrice de confusion

un ensemble de classes. De plus, les classes prédominantes sont avantagées. La matrice de confusion est utilisée pour calculer d'autres métriques présentées ci-dessous.

Les matrices de confusion des 3 classes de l'exemple de référence sont présentées dans le Tableau 3.1

TABLE 3.1 – Matrices de confusion des classes jaune, rouge et noir

	Classe Jaune		**Classe Rouge**		**Classe Noire**	
	Prédiction Classe	**Prédiction Non classe**	**Prédiction Classe**	**Prédiction Non classe**	**Prédiction Classe**	**Prédiction Non classe**
Vérité Terrain Classe	10	0	0	10	70	10
Vérité Terrain Non classe	15	75	5	85	0	20

Les matrices de confusion ne permettent pas d'obtenir une valeur unique pour quantifier la qualité de la segmentation prédite. D'autres métriques sont utilisées et privilégiées tel que l'accuracy, F1-Score ou encore l'IoU pour quantifier cela.

3.1.5.2 *Accuracy*

Cette métrique est très utilisée pour les CNN de classification. Elle calcule le pourcentage de pixels correctement prédits par rapport au nombre total de pixels. Elle n'est toutefois pas adaptée pour la segmentation d'image car elle est dépendante de la taille des objets. En effet, si les objets sont petits, la métrique n'est pas significative car le nombre de pixels n'appartenant pas à la classe est bien supérieur aux pixels de la classe. Le résultat de cette métrique sera donc très élevé bien que la segmentation de certains objets soit mauvaise.

Pour l'exemple de référence, l'*accuracy* de l'image est calculé de la façon suivante :

$$Accuracy = \sum_{i=1}^{class} \frac{pixCorrectPredict_{classe}}{\sum pixel_{Image}} \tag{3.1}$$

$$Accuracy = \frac{70+10+0}{70+25+5} = \frac{80}{100} = 0.80 \tag{3.2}$$

Avec :

— $pixCorrectPredict_{class}$: le nombre de pixels bien prédits d'une classe

— $pixel_{Image}$: le nombre de pixels dans l'image

3.1.5.3 *F1 Score*

La troisième métrique, *F1 Score* ou *Dice Coefficient*, prend en compte la taille des objets. La métrique fait un ratio entre le nombre de pixels correctement prédits et le nombre de pixels prédits + le nombre de pixels de la vérité terrain.

$$F1Score = \frac{2VP}{2*VP+FN+FP} \tag{3.3}$$

En règle générale, la métrique est appliquée de façon séparée à chaque classe puis ensuite la moyenne est calculée.

Pour l'exemple de référence , la métrique F1 Score est calculée de la façon suivante :

$$F1Score = \frac{1}{numberClass} \sum_{i=1}^{class} \frac{2 * pixCorrectPredict_{class}}{\sum(pixPredict_{class} + pixGT_{class})} \tag{3.4}$$

$$F1Score = \frac{\frac{2*70}{80+70} + \frac{2*10}{10+25} + \frac{2*0}{10+5}}{3} = 0.5 \tag{3.5}$$

Avec :

— $numberClass$: le nombre de classes totales dans l'image

— $pixCorrectPredict_{class}$: le nombre de pixels bien prédits d'une classe

— $pixPredict_{class}$: le nombre de pixels prédits d'une classe

— $pixGT_{class}$: le nombre de pixels dans la vérité terrain d'une classe

3.1.5.4 IoU

La quatrième métrique, IoU (*Intersection over Union*), est la plus utilisée et la plus significative pour mesurer la performance des CNN de segmentation. L'intersection représente l'ensemble des pixels correctement prédits. L'union est l'ensemble des pixels à prédire ajouté aux pixels prédits (Figure 3.7). La métrique ne prend donc pas en compte deux fois les pixels correctement prédits contrairement à la métrique *F1-Score*.

$$IoU = \frac{Intersection}{Union} = \frac{VP}{VP+FN+FP} \tag{3.6}$$

Cette métrique prend en compte les pixels prédits et les pixels à prédire et non l'ensemble des pixels de l'image. Cela permet à la métrique de s'adapter à la taille des objets à prédire et donc d'être robuste à des classes qui sont sur-représentées ou sous-représentées dans l'image. Comme pour le *F1 Score*, la métrique est appliquée de façon séparée à chaque classe puis la moyenne est calculée.

Les métriques IoU et *F1 Score* sont fortement corrélées mais la métrique IoU a tendance à pénaliser plus fortement les erreurs de segmentation que *F1 Score*.

Expérimentalement, un IoU supérieur à 50% (valeur arbitraire) peut être considéré comme satisfaisant de manière générale, bien que cette métrique puisse tendre vers 100% en fonction de l'application (ex : contrôle qualité).

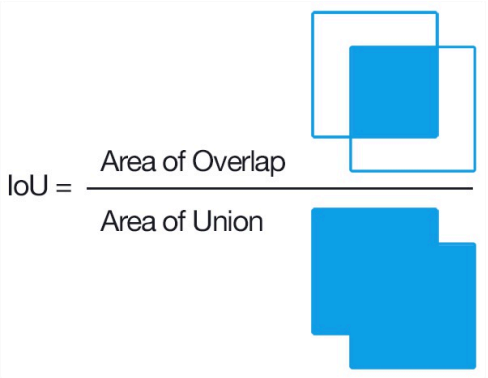


FIGURE 3.7 – Exemple illustrant le calcul de la métrique IoU [13]. Les 2 carrés, initialement non pleins, sont la prédiction et la zone de la vérité terrain. La zone pleine en haut correspond à l'endroit où la prédiction et la vérité terrain se superposent. La zone pleine en bas correspond à l'endroit où il y a la prédiction ou vérité terrain, qu'elles se superposent ou non.

Pour l'exemple de référence, l'IoU est calculé de la façon suivante :

$$IoU = \frac{\sum_{i=1}^{class} \frac{pixCorrectPredict_{class}}{pixelsGroundTruthOrPrediction_{class}}}{numberClass} \tag{3.7}$$

$$IoU = \frac{\frac{70}{70+5+5} + \frac{10}{10+10+5} + \frac{0}{10+5}}{3} = 0.425 \tag{3.8}$$

Avec :

— $numberClass$: le nombre de classes totales dans l'image

— $pixCorrectPredict_{class}$: le nombre de pixels bien prédits d'une classe

— $pixelsGroundTruthOrPrediction_{class}$: le nombre de pixels de la classe prédite ou dans la vérité terrain

Le Tableau 3.2 permet de synthétiser les métriques et notamment de montrer que la métrique IoU est la métrique la plus adéquate pour analyser les résultats d'une méthodes de segmentation.

TABLE 3.2 – Synthèse des différentes métriques

Métrique	Pénalise les petites classes	Adéquate à un problème de segmentation	Pénalise fortement les erreurs
Matrice de confusion	Oui	Non	Non
Accuracy	Oui	Non	Non
F1 Score	Non	+	Oui
IoU	Non	+	Oui (le plus)

3.1.6 Bilan

Les CNN de segmentation sont actuellement les algorithmes les plus performants pour segmenter une scène complexe. Il existe plusieurs modèles de CNN qui sont notamment adaptés aux images d'entrée, que ce soit la résolution et le type d'image, et l'image de sortie. Un apprentissage est réalisé grâce à une base de données pour pouvoir les paramétrer. Pour vérifier leur qualité de segmentation, plusieurs métriques peuvent être utilisées. Cependant la métrique IoU est privilégiée car elle est plus robuste aux classes sur-représentées et sous-représentées dans l'image. Les CNN sont des algorithmes qui permettent d'extraire des informations haut niveau de la scène avec cependant une moins bonne prise en compte des informations bas niveau en sortie. Pour

cela, une méthode travaillant exclusivement sur des informations bas niveau, tel que les contours actifs, est explorée.

3.2 Contours actifs

3.2.1 Les différents types de contours actifs

Les contours actifs font partis de l'approche issue des modèles déformables. Ils ont comme principe d'évoluer d'un modèle initial vers un modèle qui minimise son énergie. Il existe deux types de contours actifs :

3.2.1.1 Level Set

Cette méthode est basée sur la théorie de l'évolution des courbes et des surfaces implicites [28]. Une surface implicite est définie en utilisant une fonction F(x,y) : lorsque les points sont à l'extérieur de la fonction, F(x,y)>0 ; lorsque les points sont à l'intérieur de la fonction, F(x,y) <0 et lorsque la fonction F(x,y)=0, les points sont sur la limite de l'objet. La courbe se déplace de façon itérative vers la position définie par la fonction F. Elle s'arrête lorsque la courbe ne bouge plus ou lorsque le nombre d'itérations prédéfini est atteint. La fonction F peut être définie grâce aux gradients, textures, couleurs ou encore aux bruits de l'image. La méthode Level Set ne nécessite pas de paramètres durant l'évolution de la courbe : c'est un modèle non paramétrique. Cependant le choix de la fonction F est très important. Cette méthode permet la segmentation de plusieurs objets ayant les mêmes propriétés (contour fort, objet clair...) avec un seul contour initial. Les modèles basés sur une segmentation frontière utilisent une fonction F utilisant les gradients de l'image (et notamment là où il est important) alors que les modèles basés sur une segmentation région utilisent une fonction F minimisant l'énergie calculée grâce aux textures ou aux informations statistiques des régions. La principale difficulté de cette méthode est de pouvoir déterminer la fonction permettant de décrire la courbe initiale du contour. De plus, la définition de la fonction d'énergie F peut être complexe pour pouvoir bien la paramétrer.

3.2.1.2 Snake

Initialement conçus pour une courbe déterminée par une équation mathématique, les Snakes ont été adaptés à l'image en discrétisant la courbe en points[16]. Cette méthode

peut donc facilement être implémentée. C'est un modèle déformable paramétrique c'est-à-dire qu'il utilise plusieurs paramètres pour pouvoir évoluer. Ces paramètres servent à mettre plus ou moins en avant les différentes forces utilisées. Plusieurs inconvénients sont cependant à noter tels que :

— le paramétrage qui peut être complexe et fastidieux ;

— l'initialisation du contour qui doit être plus ou moins loin de l'objet pour que le contour actif puisse converger ;

— la topologie, c'est-à-dire que le contour actif est incapable de se séparer en plusieurs objets ;

— la concavité, c'est-à-dire que le contour actif a du mal à rentrer dans des zones concaves ;

— la discrétisation du contour qui impacte la qualité finale du contour ou le temps de calcul.

Les contours actifs Snake sont choisis car ils permettent une grande flexibilité et une simplicité d'implémentation. De plus, la solution proposée permet d'outrepasser ou de minimiser les inconvénients de cette méthode notamment l'initialisation, la discrétisation et la topologie.

Le contour actif est donc considéré comme un ensemble de points reliés entre eux comme le montre la Figure 3.8.

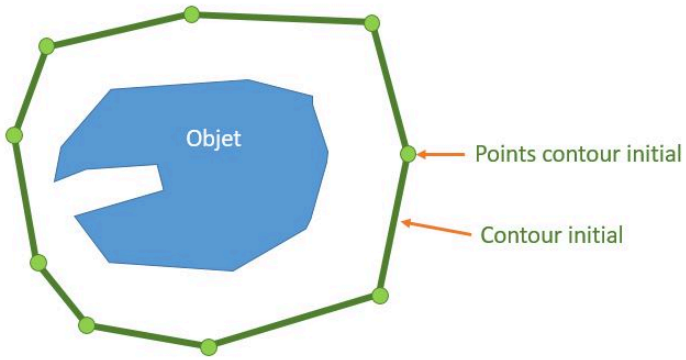


FIGURE 3.8 – Représentation d'un contour actif initialisé autour d'un objet

De façon itérative, la position de tous les points de la courbe est mise à jour grâce à des forces prédéfinies. Étant donné que les forces ne sont pas des valeurs mais des vecteurs, il est nécessaire de les convertir en énergies qui sont, elles, des valeurs numériques. Par équivalence avec un raisonnement en mécanique classique, le produit du déplacement de chaque point multiplié par la force qui s'exerce dessus est assimilé à une énergie. Étant donné que les contours actifs doivent minimiser leur modèle, ils doivent déplacer

les points du contour vers une position qui minimise l'énergie totale.
L'énergie totale est la somme des énergies définies dans le contour actif.

Les forces sont séparées en deux groupes : les forces internes et les forces externes.

3.2.2 Forces internes

Les forces internes prennent uniquement en compte les points du contour initial en entrée et non les informations issues des images (Figure 3.9). L'énergie interne associée

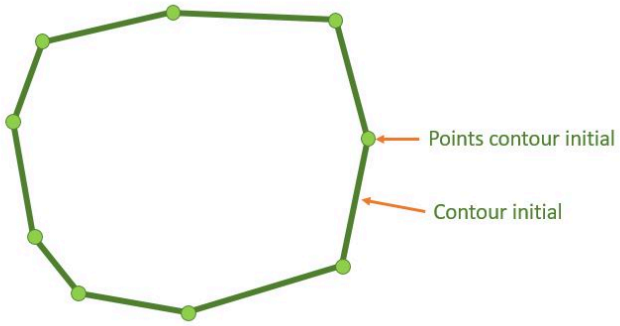


FIGURE 3.9 – Représentation d'un contour actif initialisé

aux forces est la somme des trois énergies (cohésion, raideur et ballon) présentées ci-dessous. Toutefois, l'utilisateur peut choisir de ne travailler qu'avec une, deux ou la totalité de ces trois forces.

3.2.2.1 Force cohésion

La première force interne représente la cohésion de la courbe, c'est-à-dire la distance entre un point et ses proches voisins. Plus les points sont réguliers et rapprochés, plus la force est petite. Pour cela l'équation utilisant la dérivée première de la courbe v' au point s est mise en œuvre. Le coefficient α est introduit pour pouvoir pondérer cette énergie.

$$E_{cohesion} = \int_0^1 \alpha \parallel v'(s) \parallel^2 ds \tag{3.9}$$

En discrétisant cette équation à chaque point caractéristique de la dérivée première de la courbe v', l'équation suivante est obtenue :

$$E_{cohesion} = \alpha(\parallel \overrightarrow{point_{actuel}, point_{suivant}} \parallel - distanceMoyenne_{pointscourbe})^2 \tag{3.10}$$

3.2.2.2 Force raideur

La deuxième force interne représente la raideur de la courbe, c'est-à-dire la déformation de la courbe. Plus les points forment un angle aigu, plus l'énergie est importante. L'équation utilisant la dérivée seconde de la courbe v'' au point s est mise en œuvre avec un coefficient β pour pouvoir pondérer cette énergie.

$$E_{raideur} = \int_0^1 \beta \parallel v''(s) \parallel^2 ds \tag{3.11}$$

En image, cela revient à écrire l'équation suivante :
pour chaque point, avec x représentant l'abscisse et y l'ordonnée du point dans le repère image,

$$dd_x = 2(point_{actuel}.x) - point_{suivant}.x - point_{precedent}.x \tag{3.12}$$

$$dd_y = 2(point_{actuel}.y) - point_{suivant}.y - point_{precedent}.y \tag{3.13}$$

$$E_{raideur} = \beta \frac{(dd_x^2 + dd_y^2)}{(\parallel \overrightarrow{point_{actuel}, point_{suivant}} \parallel * \parallel \overrightarrow{point_{actuel}, point_{precedent}} \parallel)^2} \tag{3.14}$$

3.2.2.3 Force ballon

Lorsque les deux premières forces sont additionnées, le contour actif a tendance à s'écrouler sur lui-même (diminution de la distance) en un cercle (raideur de la courbe minimale). Pour éviter cet affaissement, une force peut être ajoutée : la force ballon. Elle est obtenue avec n le vecteur normal unitaire à la courbe au point s. La constante κ permet de pondérer cette énergie [8].

$$E_{ballon} = \int_0^1 \kappa n(s)\, ds \tag{3.15}$$

3.2.3 Forces externes

Les forces externes prennent en compte les informations issues de l'image à segmenter. La Figure 3.10 représente ces forces externes.

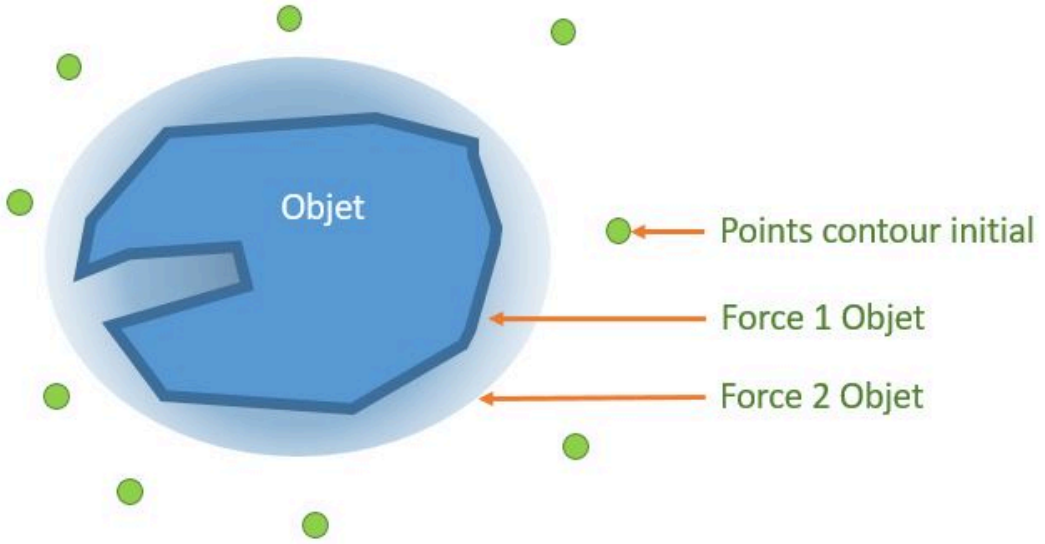


FIGURE 3.10 – Représentation d'un contour actif initialisé avec les forces issues de l'objet

Cela implique d'exécuter ces forces en chaque point. Deux méthodes permettent cela : l'approche par programmation dynamique et l'approche « greedy » ou glouton [19]. Le but est de pouvoir prendre au voisinage du point du contour les pixels de l'image, d'estimer les forces internes et externes en chaque voisin pour estimer ceux ayant l'énergie la plus faible par rapport au point initial. Il est possible d'accélérer le processus en considérant les 8 voisins proches sur un carré de 3x3 pixels, seuls les 4 pixels qui ne sont pas « en diagonale » sont pris en compte. Si un de ces 4 voisins minimise le plus l'énergie alors le point « initial » est déplacé vers cette position et les autres points voisins ne sont pas considérés. Différentes forces peuvent être considérées comme forces externes. Comme pour les forces internes, l'utilisateur sélectionne celle(s) qui répond(ent) à ses attentes. Quelques-unes sont présentées ci-après.

3.2.3.1 Force intensité image

Pour une image I, la constante I_0 permet d'effectuer un seuillage sur l'énergie et le signe $\pm$ permet de favoriser les pixels sombres ou lumineux. La constante τ permet de pondérer cette énergie.

$$E_{intensite} = \pm \int_0^1 \tau(I(v(s)) - I_0)^2 \, ds \tag{3.16}$$

En image, cela revient à écrire l'équation suivante pour chaque point s :

$$E_{intensite} = \pm\tau(I(s) - I_0)^2 \tag{3.17}$$

3.2.3.2 Force gradient image

Cette force est nommée $\nabla I(s)$ pour l'image I. Le gradient permet de prendre en compte le contour des objets. Plus le gradient est fort, plus l'énergie est faible grâce au signe moins devant l'équation. La constante σ permet de pondérer cette énergie.

$$E_{gradient} = -\int_0^1 \sigma(\nabla I(v(s))^2\, ds \tag{3.18}$$

En image, l'équation peut s'écrire comme suit pour chaque point s :

$$E_{gradient} = -\sigma(\nabla I(s))^2 \tag{3.19}$$

3.2.3.3 Force GVF

Il s'agit de la force de flux vectoriel du gradient (Gradient Vector Flow – GVF) [9]. Cette force permet d'attirer la courbe vers les bords de l'objet même si elle n'est pas près du gradient. Le champ du flux vectoriel du gradient peut être défini par un vecteur $v(x,y) = [u(x,y), v(x,y)]$ qui minimise l'énergie suivante :

$$\varepsilon = \iint \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla(\nabla I)|^2 |v - \nabla(\nabla I)|^2\, dxdy \tag{3.20}$$

Si la dérivée seconde est forte, le v minimisant l'énergie est égal à $\nabla(\nabla I)$. Au contraire, si $\nabla(\nabla I)$ est faible, l'énergie est dominée par la somme des carrés des dérivés partielles du champ de vecteur. Cela permet de forcer le champ de vecteur, pas à pas, à propager « le rayonnement » du gradient dans les zones où celui-ci est faible et à garder une valeur forte des vecteurs lorsque ceux-ci sont près des gradients forts de l'image.

En image, pour pouvoir propager l'énergie, les algorithmes itèrent le processus permettant de calculer le GVF à partir du GVF précédent. Le champ vectoriel final de

chaque itération n+1 est égal à :

$$u_{i,j}^{n+1} = (1-(\nabla I_x{}^2 + \nabla I_y{}^2) * \Delta t) u_{i,j}^n + r(u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n - 4 * u_{i,j}^n) + \nabla I_x(\nabla I_x{}^2 + \nabla I_y{}^2) * \Delta t$$

$$v_{i,j}^{n+1} = (1-(\nabla I_x{}^2 + \nabla I_y{}^2) * \Delta t) v_{i,j}^n + r(v_{i+1,j}^n + v_{i-1,j}^n + v_{i,j+1}^n + v_{i,j-1}^n - 4 * v_{i,j}^n) + \nabla I_y(\nabla I_x{}^2 + \nabla I_y{}^2) * \Delta t$$

Avec Δt, le pas de temps à chaque itération [35] est défini grâce au paramètres μ et r=0.25. Le paramètre μ est réglé en fonction du bruit dans l'image (augmente lorsqu'il y a plus de bruit). Δt est défini grâce à l'équation ci-dessous :

$$\Delta t = \Delta x * \Delta y / (r * \mu) \tag{3.21}$$

L'énergie en chaque point de la courbe est donc égale à :

$$E_{GVF} = - \parallel \overrightarrow{V(i,j)} + \overrightarrow{Point_{initial}, Point_{voisin}} \parallel \tag{3.22}$$

3.2.4 Bilan

De nombreuses forces existent et ont chacune une information différente pouvant être utilisée par le contour actif pour atteindre la position minimisant son énergie, et donc, normalement, le contour de la forme. Le contour actif de type Snake présente l'avantage de pouvoir choisir les forces voulues pour l'objectif fixé. Les forces internes sont uniquement liées aux points du contour actif et non à l'image sur laquelle le contour actif est appliqué. Ils sont liés à leurs emplacements, que ce soit leurs distances entre eux ou la forme du contour (plus ou moins arrondi). Ces forces sont utiles lorsque le résultat du contour actif doit suivre des règles qui ne sont pas liézs à l'objet traité, ou pour éviter un effondrement du résultat.

Les forces externes sont liées uniquement à l'image d'entrée et sont notamment liées à l'intensité, aux gradients et aux GVF des objets de la scène. Les forces de gradients et GVF sont deux forces spécifiques aux contours d'un objet alors que la force d'intensité est liée uniquement à la valeur des pixels. Les forces de gradients et GVF sont

privilégiées pour traiter des problématiques de frontière.

Cette méthode comporte cependant plusieurs défauts. En premier lieu, l'initialisation est très importante car si elle n'est pas placée dans l'image, il est possible que le contour actif ne trouve pas la solution ou alors pas la bonne (contours sur une autre pièce que celle attendue, contour final ne suivant pas le contour de la pièce, contour suivant le contour de deux pièces, etc.)

Un deuxième défaut réside dans le choix de la valeur des paramètres liés aux différentes forces. En fonction de ces valeurs, la courbe peut ne pas atteindre le contour de la forme ou alors dépasser celui-ci pour s'attacher à un autre contour plus fort, plus contrasté dans l'image. Plusieurs méthodes existent pour configurer les paramètres : l'apprentissage des paramètres ou la méthode par force brute qui consiste à tester toutes les valeurs possibles et trouver les valeurs qui sont les plus adéquates.

3.3 CNN segmentation combiné aux contours actifs

Il existe plusieurs stratégies associant les CNN à d'autres algorithmes de segmentation dans le but d'améliorer la segmentation de l'image. On peut citer notamment une méthode combinant un CNN et l'algorithme de partage des eaux segmentant des images de cellules humaines [18] ou encore une méthode combinant un CNN et l'algorithme des graphes pour améliorer la segmentation d'une scène [23].

Les méthodes présentées ci-dessous portent uniquement sur les CNN et les contours actifs.

En outre, les solutions utilisant les contours actifs proposées dans la littérature s'appuient sur la méthode Level Set. Pour prendre en compte les contours actifs avec le CNN, deux approches sont possibles : la modification de la loss du CNN pour prendre en compte les caractéristiques de l'algorithme ou l'ajout des contours actifs en post-processus pour améliorer la segmentation issue du CNN.

3.3.1 Modification de la loss du CNN

La solution proposée par [7] et [17] modifie la loss de leur CNN de segmentation pour améliorer leur segmentation finale. La loss, du fait qu'elle se minimise et qu'elle peut prendre plusieurs énergies, est bien adaptée pour appliquer des énergies issues des

contours actifs. Dans le premier article, la solution prend en compte l'énergie liée à la longueur de l'objet, ici des organes acquis par une imagerie à résonance magnétique (IRM), l'aire de l'organe, l'aire en dehors de l'organe ainsi que les informations sur les surfaces de niveaux (Level Set). La loss va donc minimiser l'écart entre la longueur, l'aire intérieure, l'aire extérieure et les informations sur les surfaces de niveaux (Level Set) du contour prédit par le CNN et ces mêmes caractéristiques mesurées sur la vérité terrain. Le deuxième article [17] ne prend pas directement en compte la longueur et les aires mais la carte de probabilités des classes pour chaque pixel. Il s'agit de l'ensemble des probabilités d'appartenance aux différentes classes pour chacun des pixels de l'image d'entrée.

3.3.2 Ajout des contours actifs en post traitement

La plupart de la littérature, principalement issue du domaine médical comme pour la segmentation de cellule, de tumeur ou des ventricules du coeur, utilise les contours actifs en post traitement.

Plusieurs articles doivent initialiser manuellement leur contour actif, [12] [37]. Le CNN mis en place, qui ne nécessite pas beaucoup de couches (6), estime la classe des pixels de l'image.

Les probabilités du CNN sont conservées et sont ensuite utilisées dans les contours actifs utilisant la méthode Level Set pour mettre à jour la courbe. Pour cela, les probabilités sont utilisées pour mettre à jour les variables paramétrant l'arrondi de la courbe, la direction et la magnitude de l'évolution de celle-ci. L'énergie de la courbe sera minimisée au niveau de l'information de l'image lorsque la probabilité est égale à 0,5 c'est à dire entre la classe fond (0) et la classe à analyser (1).

D'autres solutions utilisent les CNN pour initialiser le contour actif. Dans un article, [36], le CNN est utilisé pour segmenter en premier les cellules humaines à traiter. Pour cela une fenêtre se déplace dans toute l'image et propose au CNN des morceaux d'image. Un algorithme permet de réduire le nombre de morceaux pour qu'il n'y ait pas de doublon. Un cercle est généré, pour chaque cellule trouvée, grâce au centre de ces cellules. C'est l'initialisation du contour actif. Le contour actif utilisant la méthode de Level Set avec les informations de l'image ainsi que la carte de probabilités du CNN est utilisé pour améliorer la segmentation. Un deuxième article, [5], utilise la segmentation du CNN comme initialisation pour le contour actif. Il ne prend pas en compte deux régions mais quatre car les auteurs veulent segmenter deux zones accolées. Ces

régions sont retrouvées par la suite dans l'algorithme des contours actifs pour attirer ou repousser les deux contours actifs.

Dans la littérature, aucun article ne prend en compte les images RGB-D dans les contours actifs. Les images RGB-D traitent uniquement de scène du quotidien donc avec des objets complexes, c'est à dire n'ayant pas leur profondeur et/ou leur couleur de la même intensité. Il est donc difficile de paramétrer une fonction de niveau unique pour pouvoir segmenter un type d'objet en particulier. C'est notamment pour cela que la méthode Level Set est peu en adéquation avec notre problématique.

3.4 Méthode proposée

Le processus global est composé de 4 étapes décrites dans la Figure 3.11. La première étape est la segmentation d'une image avec un CNN entraîné. Le CNN traite les informations de l'image à un haut niveau. Ce traitement a l'avantage d'obtenir des informations proches du raisonnement humain et donc de détecter des classes d'objets. La deuxième étape est l'initialisation du contour actif grâce aux images étiquetées provenant du CNN. Cette étape transforme les objets composés de pixels en contours de points équidistants et calcule les forces utilisées par les contours actifs. La troisième étape est l'estimation, grâce à un CNN de classification, des coefficients des forces qui seront utilisées lors de la dernière étape. Celle-ci, c'est-à-dire le contour actif à proprement parler, déforme les points du contour pour qu'ils puissent se déplacer vers le bord de l'objet.

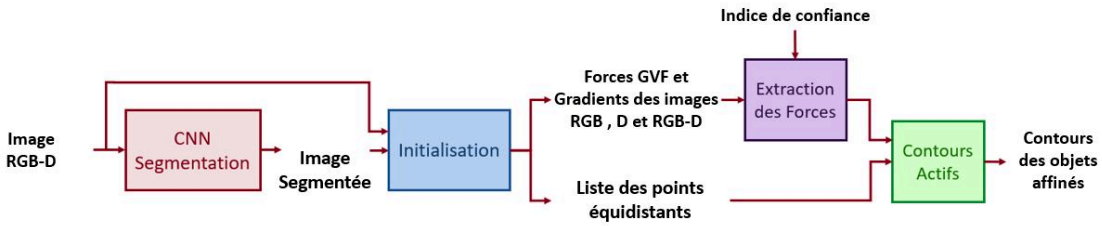


FIGURE 3.11 – Schéma global de la solution proposée

3.4.1 CNN de segmentation RGB-D

Le but de la segmentation grâce à un CNN est de pouvoir extraire une image étiquetée de la scène grâce à une image couleur et image de profondeur (Figure 3.12). L'image de couleur est utilisée par rapport à une image en niveau de gris pour pouvoir obtenir des informations sur la couleur de l'objet. Si l'objet a la même couleur,

quelque soit la situation, alors la couleur sera une des caractéristiques importante pour segmenter l'objet. A l'inverse, si les objets ne sont pas de la même couleur, alors le CNN utilisera d'autres caractéristiques, tels que les contours, pour caractériser l'objet et ainsi, le segmenter. L'image de profondeur est utilisée pour obtenir l'information en trois dimensions des objets.

Tous les CNN traitant des images RGB-D utilisent la profondeur pour estimer l'image segmentée. Deux groupes existent :

— les architectures utilisant en entrée les images RGB-D de la scène à traiter ;

— les architectures prédisant l'image de profondeur grâce à l'image RGB de la scène à traiter.

Les premiers types d'architecture utilisent la profondeur dans la partie encoder soit dans une branche à part, soit mêlée avec l'image RGB pour extraire les caractéristiques de l'image. Les deuxièmes types d'architectures prédisent l'image de profondeur en sortie, avant la prédiction de l'image labélisée. En effet, cette image de profondeur permet par la suite d'améliorer la segmentation. Pour paramétrer cette estimation de la profondeur. Ces architectures utilisent les images RGB-D de la base de données en entraînement.

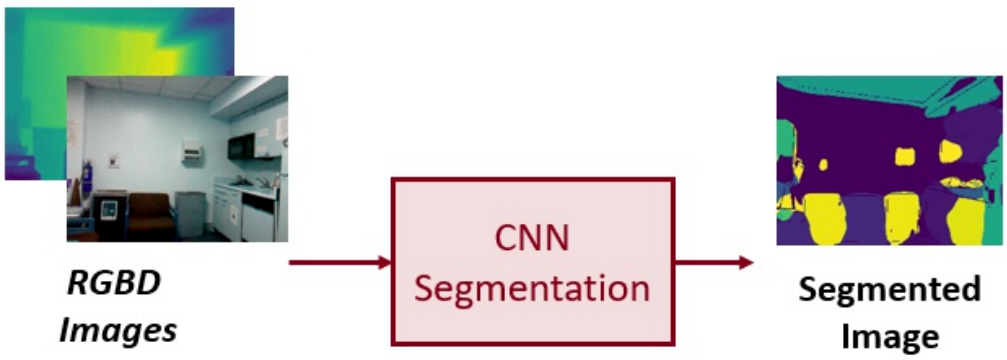


FIGURE 3.12 – CNN de segmentation prenant en entrée des images RGB-D et générant en sortie une image labélisée

Le choix de la ou des images d'entrée est fait en fonction des modèles proposés dans la littérature ainsi que de l'utilisation souhaitée. L'IoU moyen obtenu par le CNN doit être suffisamment important, c'est-à-dire au minimum égal à 50%, car le but est d'avoir une segmentation correcte, au plus près du bord, pour pouvoir en corriger les erreurs grâce aux contours actifs. Actuellement, les meilleurs CNN segmentant les bases de données RGB-D sont ceux du Tableau 3.3.

	NYUV2			SUNRGB		
Method	PixAcc	mAcc	mIoU	PixAcc	mAcc	mIoU
GAD [14]	84,8	68,7	59,6	85,5	74,9	54,5
CANet (ResNet101) [32]	77,1	64,6	51,5	85,2	x	50,6
Zig-Zag Net(ResNet152) [24]	77	64	51,2	84,7	62,9	51,8
PSD-ResNet50 [40]	77	58,6	51	84	57,3	50,6
PAP [39]	76,2	62,5	50,4	83,8	58,4	50,5
CEN(ResNet152) [33]	77,7	65	52,5	83,5	63,2	51,1
TCD(ResNet101) [38]	77,8	x	53,1	83,1	x	49,5
GLPNet(ResNet101) [6]	79,1	66,6	54,6	82,8	63,3	51,2
LWM(ResNet152) [10]	81,46	65,24	51,51	82,65	70,21	53,12
BCMFP+SA-Gate [34]	77,9	x	52,4	82,5	x	49,4

TABLE 3.3 – Résultat des CNN de segmentation RGB-D avec les meilleurs IoU(%), sur les bases de données NYUV2 et SUNRGB-D

Comme il est préférable d'avoir une initialisation aussi proche que possible de la vérité terrain, donc du contour des objets, l'architecture CNN ayant le meilleur IoU sur la base de NYUv2 est choisie. Le CNN sélectionné est GAD. Son architecture utilise une image RGB en entrée pour prédire l'image de segmentation et la profondeur de la scène. Le CNN GAD est décomposé en deux parties. La première estime une première image segmentée basse résolution et une image de profondeur. La deuxième partie utilise ces deux sorties pour fournir en sortie l'image finale de la segmentation. Son IoU moyen est de 59.6% pour la base NYUv2 et de 54.5% pour la base SUNRGBD. Ces résultats indiquent que la segmentation comporte de nombreuses erreurs. Cela peut être dû à une segmentation insuffisante de l'objet (objet sous-segmenté), ou à l'inverse, la segmentation empiète sur d'autres objets (objet sur-segmenté). Ce sont ces erreurs de segmentation que les contours actifs tentent de corriger. Les travaux sont ainsi menés avec le modèle GAD sélectionné mais la méthode et le résultat attendu (amélioration de la segmentation) restent les mêmes si un autre modèle de CNN est mis en oeuvre. L'étape suivante est l'initialisation qui convertit les pixels de l'objet en un contour. Celui-ci sera ensuite utilisé pour les contours actifs.

3.4.2 Initialisation

L'initialisation des images est importante car elles sont utilisées pour calculer les forces et l'initialisation de la courbe de l'objet qui sont les entrées du contour actif. La Figure 3.13 montre les différentes étapes de l'initialisation.

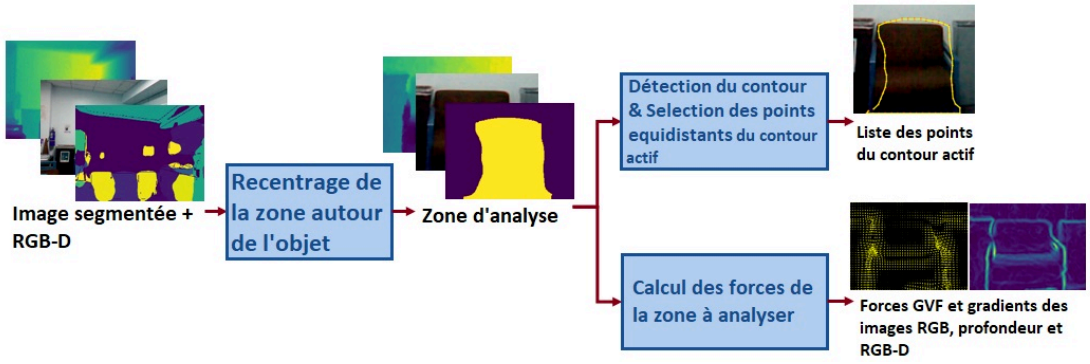


FIGURE 3.13 – Étapes de l'initialisation de la méthode proposée

La première étape consiste à recentrer les images autour de l'objet. C'est sur ces images que les différentes forces sont calculées et que la courbe initiale sera extraite. Le fait de centrer les images sur les objets permet de ne pas prendre en compte les informations des objets voisins qui pourraient être beaucoup plus contrastés et donc avoir un impact potentiel sur la segmentation de l'objet. Cette étape correspond à l'initialisation des images.
Les deux étapes suivantes à appliquer sur les images recentrées sont le choix et le calcul des forces et l'initialisation du contour.

3.4.2.1 Choix des forces externes

Les images à traiter contiennent plusieurs objets, dont les couleurs et la profondeur sont plus ou moins contrastées. La problématique derrière cette segmentation est une problématique de type frontière.

Les forces externes sont dérivées des informations de l'image. La force du gradient et la force du Gradient Vector Flow (GVF) sont retenues car elles fournissent des informations significatives pour l'amélioration des contours actifs en contours d'objets, comme indiqué dans le chapitre 3.2.4. La Figure 3.15 montre la force de gradient et GVF pour une image de profondeur d'un fauteuil.

La force du gradient est utilisée pour attirer et arrêter le contour actif soit lorsqu'il est proche du contour, soit lorsqu'il est dessus. La force du flux vectoriel du gradient (GVF) quant à elle attire le contour actif lorsqu'il est éloigné du contour. La force d'intensité n'est pas retenue car elle attire le contour actif vers les pixels blancs, c'est-à-dire vers les objets clairs, alors que les images considérées peuvent comporter des objets aussi bien clairs que foncés.

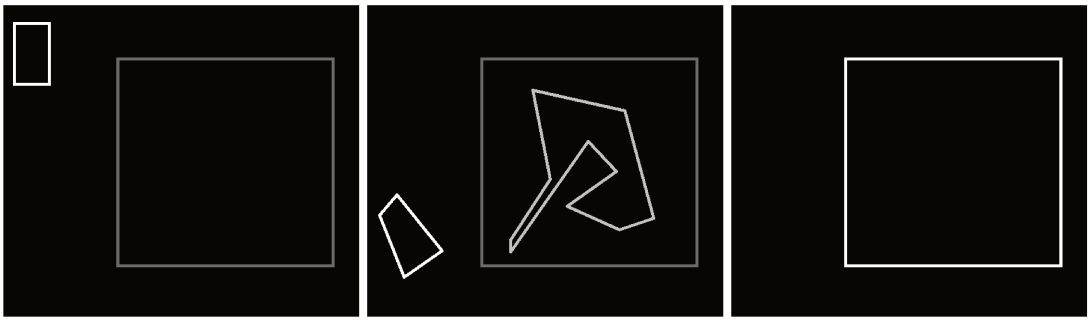
Les forces de gradient prennent leurs informations à partir de l'image en niveau de gris de la scène qui est en entrée. Deux forces de gradient ainsi que deux forces GVF sont mises en place, une pour chaque type d'image (niveau de gris et profondeur), afin de pouvoir prendre en compte les informations contenues dans ces images.

Dans la grande majorité des cas, les bords d'un objet peuvent être caractérisés par des gradients de couleur ou de profondeur. Lorsque les gradients sont importants, aucun problème n'apparaît. Cependant, étant donné que les gradients ne sont pas égaux à 0 (non présent) ou 1 (présent), il est important de faire ressortir les gradients trouvés au même endroit dans les deux images (couleur et profondeur), même faible, et de les rehausser.

Deux cas, notamment, peuvent nécessiter la création de deux nouvelles forces, c'est-à-dire le gradient et le GVF mélangeant les informations de niveau de gris extraites de l'image de couleur et de profondeur (RGB-D). Les deux cas sont représentés dans la Figure 3.14.

Le premier cas est le fait que le contour de l'objet est présent dans les deux images mais de façon peu intense. Les forces de gradient et de GVF des images de couleur et de profondeur seront très faibles et seront peu prises en compte, notamment s'il y a des contours plus marqués non loin d'eux.

Le second cas est le fait qu'un "objet saillant" au milieu de l'objet n'est contrasté que dans une des composantes alors que les contours de l'objet sont contrastés dans les deux images. L'objet "saillant" a beaucoup de chance d'attirer ou de stopper la progression du contour actif vers le bord. C'est notamment le cas d'une étiquette présente sur l'objet.



(a) Gradient de l'image de niveau de gris

(b) Gradient de l'image de profondeur

(c) Gradient de l'image RGB-D

FIGURE 3.14 – Images des gradients des 3 types d'images (niveau de gris, profondeur et RGB-D) permettant d'extraire les forces pour le contour actif

L'image RGB-D permet de rehausser les contours présents dans les deux images.

La première force envisageable est un gradient résultant de la multiplication des deux forces de gradient (couleur et gradient).
Si nous avons un dégradé sur l'image couleur et sur l'image profondeur, un gradient apparaîtra. Pour donner plus de relief à cette force, un seuillage est effectué. Si un pixel de l'image multipliant les deux types d'image (niveaux de gris et profondeur) est supérieur à 0,15 (valeur choisie arbitrairement), il est considéré comme un contour. Les pixels qui remplissent la condition seront mis à 1, tandis que les autres pixels seront mis à 0.
La deuxième force envisageable est une force GVF utilisant l'image de gradient calculée précédemment.

Dans le premier cas, les nouvelles forces exacerbent le contour de l'objet. Dans le second cas, les nouvelles forces permettent de contourner l'objet saillant et de s'accrocher à son bord.

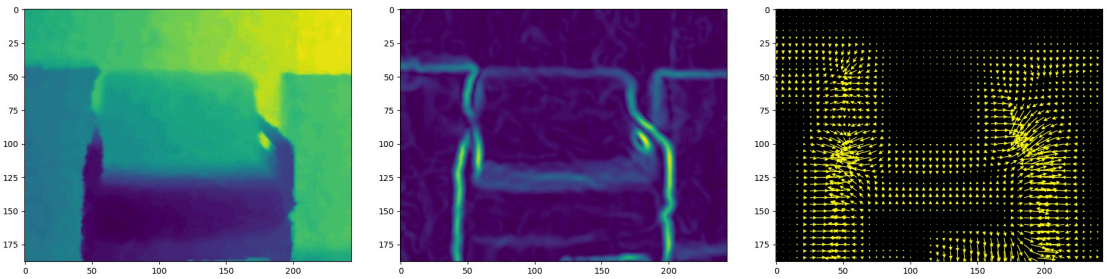


FIGURE 3.15 – Visualisation des forces gradient et GVF sur un objet de la base NYUV2 (de gauche à droite : image de profondeur de l'objet, gradient et GVF de l'objet)

3.4.2.2 Initialisation du contour

Pour éviter d'initialiser le contour actif "à la main", l'initialisation est faite par le CNN. Le CNN ne donne pas un contour initial mais un ensemble de pixels ayant la même étiquette, celle des classes. En prenant un ensemble de pixels avec la même étiquette, il est alors possible de dire qu'il y a un objet. Les CNN prédisent leur segmentation pixel par pixel, indépendamment des pixels environnants, grâce aux caractéristiques des images RGB ou RGB-D. Dans la grande majorité des cas, les CNN n'ont aucune contrainte de localisation. Cela génère des erreurs de prédiction qui n'ont pas de sens. En effet, en considérant une image d'un salon par exemple, il est possible qu'au milieu d'un tableau segmenté se trouvent quelques pixels associés à un canapé. Ces pixels sont

donc mal prédits. Ils nuisent à l'IoU final et impactent l'initialisation de l'objet. Pour éviter cette situation, une fermeture est effectuée sur l'image étiquetée issue de la sortie du CNN, afin de supprimer ces pixels mal prédits. Une fermeture est une transformation morphologique, dilatant puis érodant une image. La dilatation ferme les trous et l'érosion supprime le surplus résultant de la dilatation précédente.

Le contour d'un groupe de pixels forme un premier contour utilisé pour initialiser le contour actif. La Figure 3.16 montre la fermeture puis l'extraction des points du contour de l'objet de gauche.

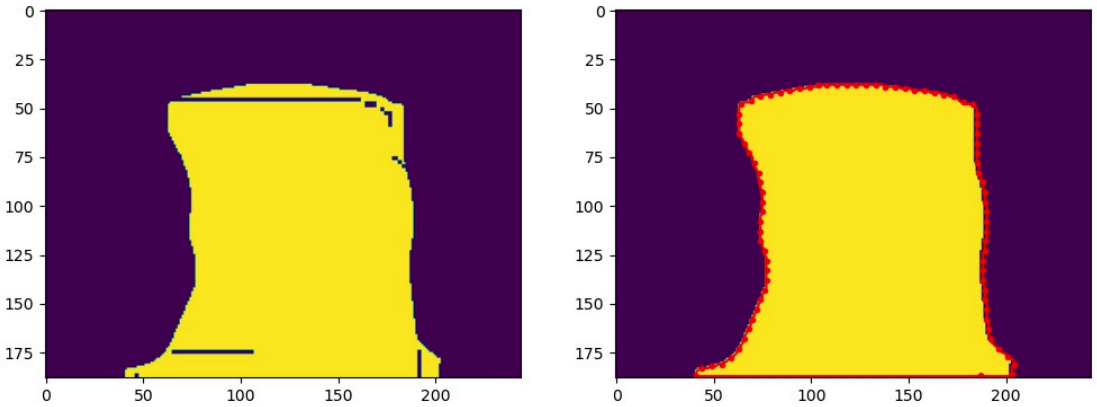


FIGURE 3.16 – Visualisation de la sortie du CNN (à gauche) et l'initialisation réalisée (à droite), c'est à dire la fermeture puis l'extraction des points du contour de l'objet (en rouge)

Afin de corriger ces problèmes de segmentation dans les objets, il est nécessaire de prendre les points de manière équidistante pour pouvoir ajuster la segmentation à chaque endroit de l'objet. Pour extraire l'acquisition de tous les points du contour de l'objet, une méthode de suivi de contour proposée dans la littérature [31] est utilisée. Il est nécessaire de mettre en entrée une image binaire (en affectant 1 aux pixels de la classe et 0 aux autres). Il est important d'avoir des points initiaux répartis de façon homogène autour de l'objet afin de ne laisser aucune zone sans amélioration du contour. Si tous les points du contour sont pris en compte, cela conduit à un grand nombre de calculs. Cependant, si trop peu de points sont pris en compte, les zones s'arrondissent et dégradent le contour initial. Cela peut alors engendrer des approximations dès le début du processus.

La méthode privilégiée est donc l'utilisation de tous les points du contour initial. Pour régler la problématique du temps de traitement, les énergies des forces sont calculées grâce au GPU de l'ordinateur. En effet, étant donné que les GPU peuvent exécuter en parallèle une tâche sur un ensemble de pixels, le temps de traitement sera fortement

diminué. Pour cela la librairie Pytorch, librairie spécialisée dans les réseaux de neurones et notamment dans les calculs sur GPU, sera utilisée.

Cependant, si le nombre de points de l'objet à analyser est trop important et/ou que le temps de traitement nécessite d'être raccourci par rapport à la première solution, la deuxième solution pourra être utilisée en complément de la première solution. Cette étape consiste donc à réduire le nombre de points du contour à un pourcentage choisi arbitrairement. Pour cela, la somme des distances entre les pixels est calculée. Les pixels sélectionnés pour le contour réduit sont choisis en fonction de la distance entre les autres points. Ils ne sont pas strictement équidistants mais aussi proches que possible du fait que les pixels diagonaux ne sont pas à la même distance($\sqrt{2}$) que les pixels verticaux ou horizontaux (1). Afin de limiter l'impact du pourcentage de points pris au début, la différence d'IoU entre le nouveau contour et le contour initial est calculée. Si elle est supérieure à 0,01, la distance est réduite et davantage de points sont pris. Le processus est répété jusqu'à ce que la condition soit validée. De la sorte, un contour composé d'un moindre nombre de points est obtenu. La forme initiale de contour actif qui en résulte ne change pas ou peu par rapport à celle constituée par l'intégralité des points. Les points constitutifs sont équidistants les uns des autres.

3.4.3 Extraction des énergies paramétrées des forces externes

Les forces extraites dans la partie Initialisation permettent d'acquérir les gradients et les GVF de la scène à traiter. Cependant, ces forces ne peuvent pas être utilisées en l'état car elles peuvent être déséquilibrées en fonction de l'objet analysé. Par exemple, si un objet est très contrasté en termes de profondeur alors que les autres objets le sont peu, alors les GVF ne seront pas bien paramétrés car ils seront impactés par cet objet. C'est notamment pour cela que dans l'initialisation est réalisée un recadrage de l'image sur l'objet à analyser. Cependant, cela ne suffit pas. Il est également nécessaire d'adapter l'énergie des forces pour que celles-ci soient plus équilibrées entre elles. Par ailleurs, des coefficients sont ajoutés à chaque énergie pour pouvoir paramétrer finement ces forces.

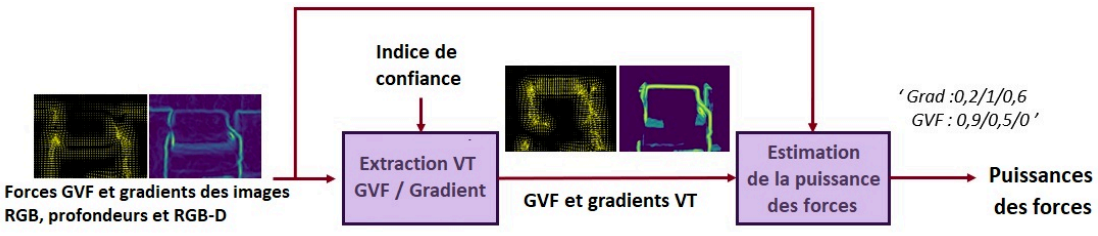


FIGURE 3.17 – Étapes permettant l'extraction des paramètres des forces

3.4.3.1 Calcul des énergies

Le calcul de l'énergie totale est une partie critique de la méthode car c'est à partir de cette information que les positions des points sont modifiées ou non.

Les énergies externes sont calculées de deux façons différentes en fonction des forces utilisées :

Les forces de gradient utilisent la valeur du pixel de l'image de force normalisée entre 0 et 1.

$$EnergyGrad = \frac{Grad_{Original} - min_{Grad}}{max_{Grad} - min_{Grad}} \tag{3.23}$$

Cela permet de garder la puissance du gradient et de rehausser les contours faibles qui ne sont pas assez marqués dans l'image à traiter.

L'énergie totale des gradients est simple à calculer car c'est la somme des énergies des gradients de chaque type d'images.

$$\begin{aligned} EnergyGrad_{Total} = Coeff_{Grad-RGB} * Grad_{RGB} + Coeff_{Grad-D} * Grad_{D} \\ + Coeff_{Grad-RGBD} * Grad_{RGBD} \end{aligned} \tag{3.24}$$

Avec :

— $Grad_X$: le gradient de l'image X (RGB, D ou RGB-D)

— $Coeff_{Grad-X}$: le coefficient de la force de gradient de l'image X (RGB, D ou RGB-D)

Les forces GVF sont représentées par des vecteurs en chaque pixel. L'amplitude de la force GVF est une des informations essentielles car elle permet de dire si le gradient est loin du pixel considéré. Pour cela, la norme du vecteur GVF est calculée. Elle est divisée par la norme maximale de la bounding box de l'image pour pouvoir augmenter le contraste.

$$Norme_{GVF-X} = \frac{\| GVF_X \|}{max_{GVF-X}} \tag{3.25}$$

Avec :

— GVF_X : le GVF de l'image X (RGB, D ou RGB-D)

— $Norme_{GVF-X}$: la norme de la force GVF de l'image X (RGB, D ou RGB-D)

Pour calculer l'énergie GVF d'un pixel, il est de plus nécessaire, de savoir quel est le pixel le plus proche qui va dans la même direction que lui. Pour cela, la projection du vecteur $centre - pixel_{voisin}$ sur le vecteur GVF du point à traiter est réalisée.
Les valeurs vont donc de 1 ($\overrightarrow{GVF}$ allant dans la même direction que le vecteur $\overrightarrow{centre - pixel_{voisin}}$) à -1 ($\overrightarrow{GVF}$ allant dans le sens contraire au vecteur $\overrightarrow{centre - pixel_{voisin}}$). Pour garder la même variation que pour les gradients, la valeur est mise entre 0 et 1.

Cependant, l'énergie totale des GVF n'est pas la somme des énergies des GVF calculés séparément. Il faut tout d'abord calculer la somme des vecteurs GVF qui sont auparavant multipliés par leurs amplitudes et leurs coefficients.

$$\begin{aligned}\overrightarrow{GVF_{Total}} = Coeff_{GVF-RGB} * Norme_{GVF-RGB} * \overrightarrow{GVF_{RGB}} \\ + Coeff_{GVF-D} * \overrightarrow{GVF_D} * Norme_{GVF-D} \\ + Coeff_{GVF-RGBD} * \overrightarrow{GVF_{RGBD}} * Norme_{GVF-RGBD}\end{aligned} \quad (3.26)$$

Avec :

— GVF_{Total} : le vecteur GVF final

— $Coeff_{GVF-X}$: le coefficient GVF de l'image X (RGB, D ou RGB-D)

— $Norme_{GVF-X}$: la norme de la force GVF de l'image X (RGB, D ou RGB-D)

L'énergie finale du GVF est la projection de ce vecteur sur les vecteurs $centre - pixel_{voisin}$ qui sont multipliés par la norme du vecteur GVF final.

$$EnergyGVF_{Total} = Projection(\overrightarrow{GVF_{Total}}, \overrightarrow{centre - pixels_{voisin}}) * \| GVF_{Total} \| \quad (3.27)$$

Avec :

— $EnergyGVF_{Total}$: l'énergie GVF finale

— $\overrightarrow{centre - pixels_{voisin}}$: le vecteur défini par le pixel où est situé le point du contour actif comme origine et un point voisin qui est la destination

— $\overrightarrow{GVF_{Total}}$: le vecteur GVF final

— GVF_{Total} : la norme de la force GVF de l'image X (RGB, D ou RGB-D)

Le fait de ne pas prendre séparément les vecteurs permet de prendre plus en compte la direction finale des GVF ainsi que de ne pas trop diminuer l'énergie finale des GVF lors de la projection. En effet, la projection n'est égale à 1 que lorsque le vecteur GVF est exactement dans la direction de l'un des vecteurs $centre - pixel_{voisin}$.

3.4.3.2 Extraction des coefficients des forces externes

La méthode proposée est composée de nombreuses forces à paramétrer (9) qui peuvent avoir des valeurs différentes pour chaque objet. Plusieurs méthodes de réglage des paramètres sont proposées dans la littérature. La première approche est la méthode brut de force. C'est la plus simple à mettre en place. Elle consiste à réaliser toutes les combinaisons possibles de forces et les appliquer sur les objets. La combinaison ayant les meilleurs résultats est retenue. Cette méthode peut s'adapter à différentes classes ou à un ensemble de la base. Cependant, étant donné que chaque paramètre peut prendre n'importe quelle valeur dans l'intervalle [0 - 1], il peut exister une infinité de solutions. Cette méthode n'est ainsi pas optimisée pour des bases de données complexes et importantes.

Une autre méthode consiste à utiliser des algorithmes génétiques [3]. Cette méthode utilise une approche biomimétique qui copie la sélection naturelle. Le principe des algorithmes génétiques est qu'une population va évoluer et créer des générations de plus en plus optimisées à leur environnement. La population, ici les combinaisons possibles des forces, est composée d'individus (une combinaison) qui contiennent des chromosomes (une force). La population va muter, fusionner et sélectionner les chromosomes les plus avantageux pour solutionner le problème. Une fonction, appelé "fitness", est utilisée pour estimer si la population s'adapte de mieux en mieux à l'environnement. Les limites de ce type d'algorithme sont :

— le temps de calcul et de convergence vers la solution est long ;
— la fonction d'analyse de la population, fitness, doit être faible en coût de calcul car elle va être utilisée très fréquemment mais elle doit être assez complexe pour pouvoir prendre en compte tous les aspects du problème ;
— le paramétrage de la taille de la population et du taux de mutations sont complexes ;
— la solution peut ne pas être la solution la plus optimale.

Étant donné que les deux méthodes précédentes demandent un temps de traitement important, une troisième méthode qui est l'utilisation des réseaux de neurones pour pouvoir extraire les forces de l'image est explorée. En effet, il est nécessaire de traiter les objets avec leurs caractéristiques bas niveau, c'est-à-dire leurs contrastes et leurs contours. Cette méthode n'est actuellement pas encore testée en littérature et sera donc développée pour vérifier si elle est plus adéquate que les autres solutions.

L'extraction des forces permet d'extraire de façon automatique les forces à utiliser pour obtenir le résultat escompté. En entrée, les contours actifs reçoivent les images de

gradient et de GVF des images RGB, de profondeur et RGB-D. Ces images sont les informations qui caractérisent les objets de l'image. Le contour des objets et les forces qui les attirent sont donc intégralement codés par ces images.

Il est donc nécessaire d'estimer l'impact des forces pour équilibrer celles-ci et ainsi pouvoir faire bouger le contour actif lorsque celui-ci n'est pas sur le contour de l'objet à segmenter et le stopper lorsqu'il est bien sur ce contour. Les données d'entrée de ce processus d'apprentissage sont les représentations des gradients et des GVF des objets segmentés (RGB,D et RGB-D).

Le processus est découpé en 4 parties :

— définition de la zone d'extraction de la vérité terrain estimée ;

— extraction et équilibrage des gradients de l'image ;

— extraction et équilibrage des forces GVF ;

— équilibrage des forces de gradient et des forces de GVF paramétrées entre elles.

3.4.3.3 Définition de la zone d'extraction de la vérité terrain estimée

Bien que la vérité terrain soit connue au niveau de l'apprentissage, il est impossible de la connaître lorsque le processus est en inférence, c'est-à-dire lorsqu'il ne connaît pas la scène et les objets qu'il traite. Il est donc nécessaire d'estimer les contours "potentiels" des objets sans en avoir une connaissance préalable. Pour cela, la sortie du CNN sera utilisée pour délimiter une zone de présence potentielle du contour (Figure 3.18). Cette zone est définie grâce à la "fiabilité" de la sortie du contour actif. La métrique utilisée pour initialiser cette fiabilité peut être l'IoU de la classe ou une métrique définie en fonction de la scène ou des caractéristiques de l'objet. Dans le cas de la base NYUv2, l'IoU de la classe est choisi. Il a l'avantage de prendre en compte la taille de l'objet tout en pénalisant fortement les contours qui segmentent trop loin de la vérité terrain.

Lorsque l'IoU est faible alors la zone de recherche des contours est grande et inversement plus l'IoU est proche de 100%, plus la zone de recherche des contours est réduite. Cette zone est définie par la distance calculée ci-dessous :

$$distance_{zone} = \frac{(100 - IoU) * \sqrt{width^2 + height^2}}{100 \times 10} \tag{3.28}$$

Avec :

— width : la taille en largeur de l'image ;

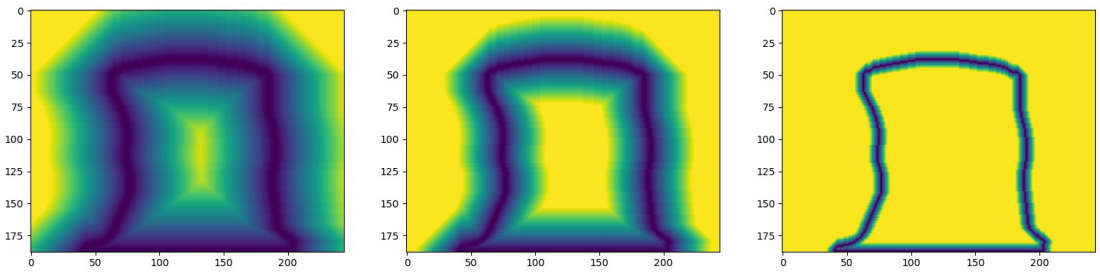


FIGURE 3.18 – Zone finale des distances avec, de gauche à droite, un indice de confiance de 0,1, 0,5 et 0,9. La zone en jaune est la zone où les gradients et les GVF ne sont pas pris en compte.

— height : la hauteur en largeur de l'image ;

— IoU : la valeur en IoU de la classe en sortie de CNN ;

— $distance_{zone}$: la distance autour de l'initialisation du CNN .

La zone finale est définie grâce à une carte des distances réalisée par plusieurs érosions successives. Cette zone servira pour l'extraction des gradients et des GVF de la vérité terrain estimée.

3.4.3.4 Extraction et équilibrage des gradients de l'image

Tout d'abord, il convient d'extraire les gradients de la vérité terrain estimée, c'est-à-dire les emplacements très probables des contours de la zone qui doit être finement segmentée. La zone dans laquelle ces contours sont extraits est définie au préalable autour de la sortie du CNN. La valeur, en chaque pixel de la zone d'extraction, des gradients de la vérité terrain estimée est égale à la valeur maximale entre les trois types de gradients (RGB, D et RGB-D). Le reste de l'image est mis à 0. Cependant, la plupart des gradients sont assez faibles dans les images de gradient originales ce qui va impacter la puissance de GVF de la vérité terrain. Une fonction est créée pour rehausser cela. Elle prend en entrée deux paramètres : une valeur de seuil et un coefficient de contraste. Le seuil permet de ne prendre en compte que les gradients supérieurs à la valeur spécifiée. Le coefficient quant à lui permet de paramétrer l'impact de la fonction sur les valeurs proches du seuil. Dans le cas de la méthode, le seuil est paramétré à 0,1 et le coefficient à 50.

$$Grad_{Modifie} = \frac{\ln\left(coeff * (Grad_{MaxImage}) + 1\right)}{\ln\left(coeff * (1 - seuil) + 1\right)} \tag{3.29}$$

Avec :

— $Grad_{Modifie}$: le gradient final rehaussé ;

— coeff : un paramètre permettant de configurer l'augmentation des gradients ;

— $Grad_{MaxImage}$: le gradient maximal trouvé dans une des trois types d'images de gradient (RGB, D, RGB-D) ;

— seuil : un paramètre permettant de ne pas augmenter les gradients plus petits que la valeur de celui-ci.

La fonction sur l'ensemble de l'intervalle des valeurs des gradients est représentée ci-dessous :

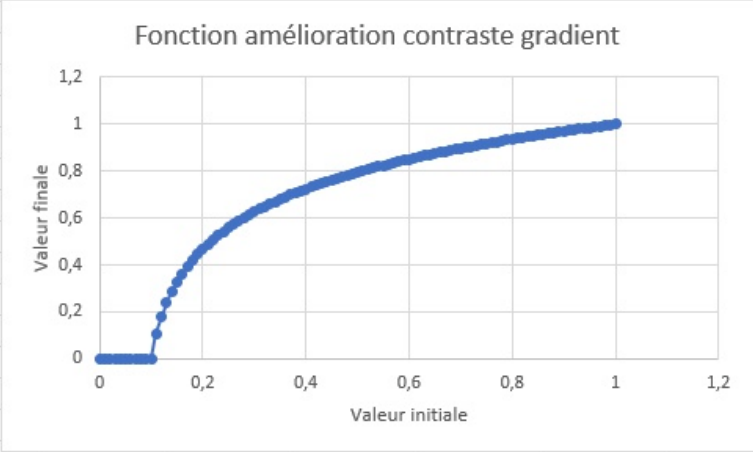


FIGURE 3.19 – Fonction de rehaussement des gradients (seuil = 0,1 et coeff = 50)

Maximiser les gradients puis rehausser les contrastes permet de faire ressortir les contours proches de l'objet à analyser. La Figure 3.20 montre un exemple de ce rehaussement sur un fauteuil.

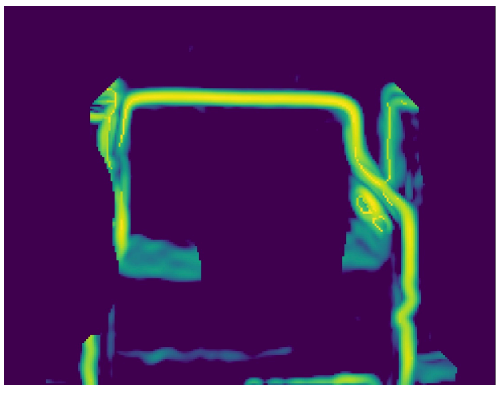


FIGURE 3.20 – Gradient pris en compte pour l'estimation de la vérité terrain estimé à partir de la sortie du CNN et de l'indice de confiance (ici égal à 0,7)

L'étape suivante vise à identifier la combinaison des forces de gradient permettant de maximiser les gradients estimés précédemment et de minimiser les autres gradients

de l'image. Maximiser les gradients du contour de l'objet doit permettre de "retenir" les contours actif sur ces gradients. De même, minimiser les autres gradients de l'image permet d'éviter que le contour actif soit retenu sur des endroits non voulus. Pour pouvoir réaliser cela, une régression est mise en oeuvre. Il s'agit d'un modèle mathématique qui permet de déterminer la meilleure combinaison de paramètres pour que l'entrée du modèle soit la plus proche de la sortie voulue.

Ce modèle utilise la backpropagation pour pouvoir corriger ces paramètres et ainsi trouver la meilleure combinaison. Ce réseau prend en entrée les images des gradients et GVF issues des images de couleur et profondeur originales ainsi que les gradients et GVF de la "vérité terrain estimée". Les paramètres à régler, c'est-à-dire la combinaison des forces à ajuster et une fonction de coût, sont choisis pour pouvoir estimer la meilleure combinaison. La fonction de coût, ou loss, est une fonction permettant de calculer une métrique. Celle-ci mesure l'écart entre la prédiction et la vérité terrain. De la sorte, les "mauvais" coefficients qui impactent le plus le résultat final sont modifiés.

Pour estimer les forces des gradients, la régression prend en entrée la vérité terrain et les gradients des images. La fonction de coût fait tendre les gradients des images qui sont multipliés par leurs coefficients respectifs, eux-mêmes modifiés à chaque itération de la régression, vers les gradients de la vérité terrain.

Étant donné que le nombre de points du contour est très inférieur au nombre de points à l'intérieur et à l'extérieur, un coefficient permettant d'équilibrer les classes est intégré. Il permet de prendre beaucoup plus en compte les erreurs dues aux pixels du contour que celles dues aux autres pixels (classWeightGrad).

Deux types de calcul d'erreur pour la loss peuvent être réalisés : l'erreur moyenne absolue (Mean Absolute Error - L1) ou l'erreur moyenne au carré (Mean Square Error). Le choix de la méthode de calcul de l'erreur est opéré dans le prochain chapitre 3.5.2 en s'appuyant sur différents tests. L'erreur moyenne absolue (Mean Absolute Error - L1) est choisie car elle permet de prendre en compte aussi bien les pixels du contour que ceux qui n'y sont pas, sans trop pénaliser les valeurs éloignées de la vérité terrain.

La fonction de coût pour les gradients est définie comme suit :

$$Loss_{Grad} = \frac{\sum |(Grad_{Input} - Grad_{VT})| * classWeightGrad}{nombrePixelImage} \tag{3.30}$$

Avec :

— $Loss_{Grad}$: la valeur de la fonction de coût pour les gradients ;

— $Grad_{Input}$: énergie des gradients des images ;

— $Grad_{VT}$: énergie des gradients des images utilisées pour la vérité terrain ;

— $classWeightGrad$: coefficient multiplicateur pour compenser les petites classes ;

— nombrePixelImage : nombre de pixels contenus dans l'image .

3.4.3.5 Extraction et équilibrage des forces GVF

Les GVF de la vérité terrain estimée, c'est-à-dire les forces amenant aux gradients de la vérité terrain estimée précédemment, sont extraits. Grâce à ces gradients, il est possible de calculer la carte des GVF associés, comme représenté dans la Figure 3.21.

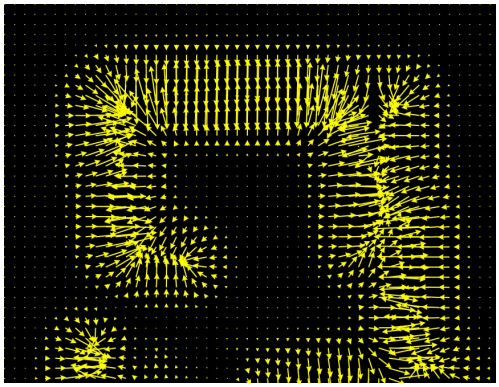


FIGURE 3.21 – GVF pris en compte pour l'estimation de la vérité terrain estimée à partir de la sortie du CNN et de l'indice de confiance (ici égal à 0,7)

Ensuite, l'apprentissage des combinaisons des forces GVF est réalisé. Cette étape permet de faire tendre les vecteurs finaux des GVF vers les vecteurs GVF de la vérité terrain, aussi bien en termes d'orientation que d'amplitude. Les vecteurs finaux des GVF sont obtenus en additionnant les vecteurs des GVF des images d'entrée, multipliées chacune par leurs coefficients respectifs. La fonction de coût est définie comme ci-dessous :

$$Loss_{GVF} = \frac{\sum |(GVF_{input} - GVF_{VT})|}{nombrePixelImage} \tag{3.31}$$

Avec :

— $Loss_{GVF}$: la valeur de la fonction de coût pour les GVF ;

— $GVFInput$: énergie des GVF des images ;

— $GVFVT$: énergie des GVF des images utilisées pour la vérité terrain ;

— nombrePixelImage : nombre de pixels contenus dans l'image .

3.4.3.6 Équilibrage entre les forces des gradients et les forces de GVF

Enfin la troisième partie est l'équilibrage des forces de gradient et des forces de GVF paramétrées entre elles.

La fonction d'apprentissage prend en entrée toutes les données car le but est de reproduire ce que devrait faire le contour actif, c'est-à-dire avoir une énergie de gradient supérieure à l'énergie de GVF sur les pixels du contour de l'objet et inversement, avoir une énergie de GVF supérieure à l'énergie de gradient sur les pixels n'étant pas sur le contour. Les coefficients $Coeff_{Grad}$ et $Coeff_{GVF}$ sont estimés. Le premier est appliqué aux forces de gradient tandis que le deuxième est appliqué aux GVF.

$$Energy_{Input} = Coeff_{Grad} * Grad_{Input} - Coeff_{GVF} * GVF_{Input} \tag{3.32}$$

$$Loss_{GVF-GRAD}^{Contour} = \frac{\sum |Energy_{Input}^{Contour} - Energy_{VT}^{Contour}|}{Nombre_{VT}^{Contour}} \tag{3.33}$$

$$Loss_{GVF-GRAD}^{NonContour} = \frac{\sum |Energy_{Input}^{NonContour} - Energy_{VT}^{NonContour}|}{Nombre_{VT}^{NonContour}} \tag{3.34}$$

$$Loss_{GVF-GRAD} = Loss_{GVF-GRAD}^{Contour} + Loss_{GVF-GRAD}^{NonContour} \tag{3.35}$$

Avec :

— $Loss_X$: la valeur de la fonction de coût pour une énergie X ;

— Contour / NonContour : points situés sur le contour / n'étant pas situés sur le contour ;

— $Coeff_X$: coefficient d'une énergie X ;

— $Energy_X$: énergie d'un nombre de pixels contenus dans l'image ;

— $Grad_X$: Gradient d'une image X ;

— GVF_X : GVF d'une image X .

La solution finale permet donc d'estimer la combinaison des forces externes utilisées par la suite par le contour actif. Cette combinaison est paramétrée grâce aux images d'entrée de gradient et de GVF des images RGB, D et RGB-D ainsi qu'à la segmentation de sortie du CNN.

3.4.4 Contours actifs

Les contours actifs sont utilisés pour mettre à jour les points de la courbe initiale avec les forces précédemment calculées et améliorées grâce à l'étape d'extraction des coefficients des forces. Le principe est simple puisqu'il s'agit de calculer l'énergie totale des forces autour de chacun des points du contour. Le point se déplace vers le point le plus proche ayant la plus petite énergie. Plusieurs itérations sont nécessaires pour aboutir à une solution stable. La Figure 3.22 représente cette étape là.

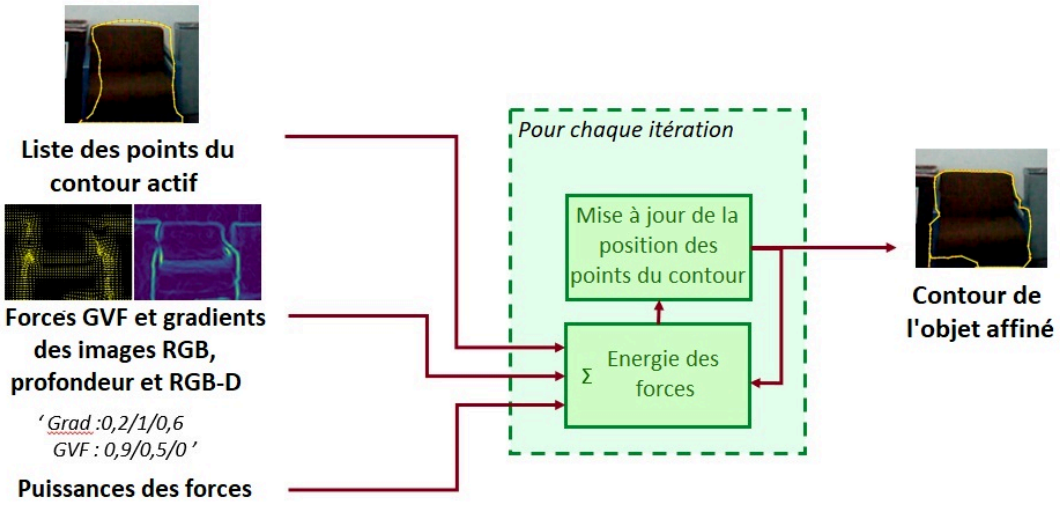


FIGURE 3.22 – Étapes présentes dans le contour actif permettant de modifier la position des points du contour initial grâce aux forces et à leurs paramètres

3.4.4.1 Forces internes

Le contour initial est issu du contour d'un objet segmenté par un CNN. Les informations bas niveau de ce contour sont régies uniquement par les informations de gradient. En effet, dans un objet, il est possible de trouver des zones où l'image est continue et lisse alors que d'autres endroits peuvent présenter des discontinuités et l'objet peut ne pas être lisse. De plus, les contours issus des CNN peuvent se trouver aussi bien à l'intérieur de l'objet qu'à l'extérieur. Dans ces deux cas, les forces internes, c'est-à-dire celles de cohésion, de raideur et de ballon, ne sont généralement pas prises en compte. Elles sont donc mises à 0.
En effet, elles ne sont pas utiles pour modifier le contour actif vers le bord de l'objet et peuvent engendrer des erreurs en forçant certains points de ce contour à ne pas se déplacer du fait des forces gérant l'aspect arrondi du contour et l'espacement entre ces

points du contour.

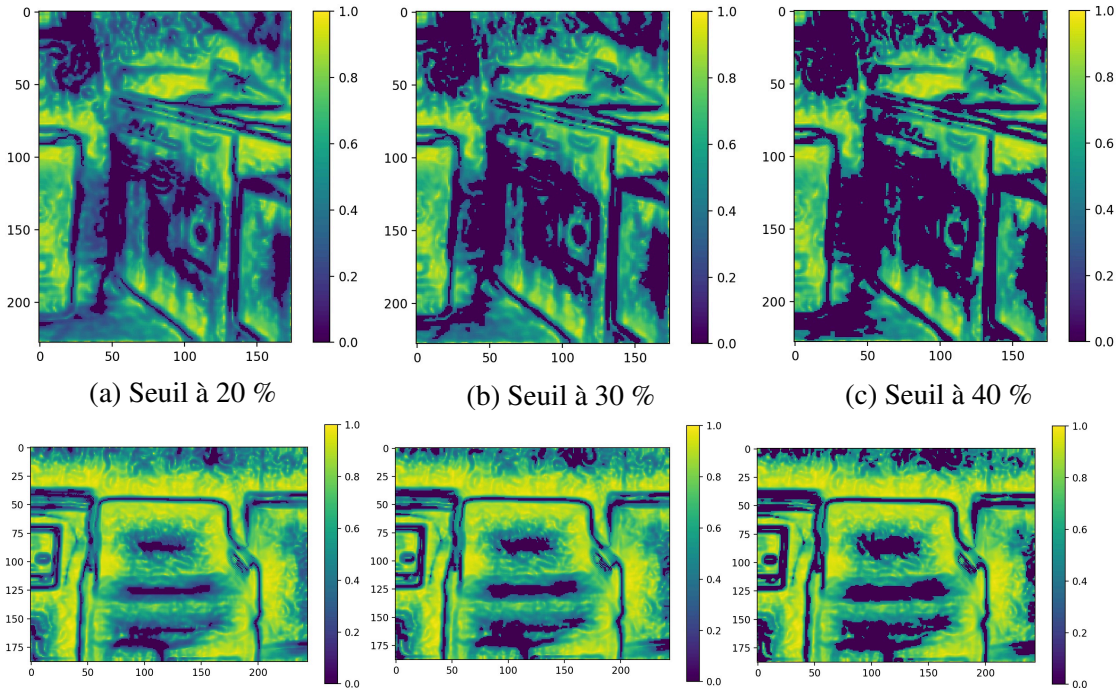
3.4.4.2 Carte de déplacement

Étant donné que les forces ont été estimées précédemment grâce à l'extraction des coefficients des forces externes d'une part et par la mise à 0 des forces internes d'autre part, la carte de déplacement peut être calculée. Pour cela, plusieurs informations sont nécessaires comme le nombre de pixels voisins (initialement égal à 8) ou encore le pas de déplacement, qui peut être paramétré de façon arbitraire ou en fonction de la taille de l'image. La carte est issue de la somme de l'$EnergyGrad_{Total}$ et $EnergyGVF_{Total}$ en chaque pixel. La carte de déplacement permet d'accélérer très fortement le processus car les GPU permettent de paralléliser facilement le déplacement des points du contour. Pour cela, en chaque pixel, un max unpooling est utilisé sur l'$Energy_{Total}$ pour trouver l'index du pixel minimisant l'énergie locale et donc la prochaine direction du déplacement si le contour est sur ce pixel.

Enfin, cette carte de déplacement doit être légèrement modifiée car comme les points du contour vont sur le pixel local ayant l'énergie la plus faible, ces pixels vont sur les contours les plus proches puis continuent à progresser au sein du contour pour trouver le maximum local. Cela va générer des erreurs de segmentation car les points se regroupent au même endroit et ne sont plus répartis uniformément autour de l'objet. Pour limiter cet impact, le pourcentage de l'$EnergyGVF_{Total}$ sur l'$Energy_{Total}$ est utilisé. L'$Energy_{Total}$ étant principalement influencée par les gradients, le déplacement est mis à 0 en dessous d'une certaine valeur de pourcentage.

Dans la Figure 3.23, deux exemples de résultat du contour actif avec une carte de déplacement modifiée par un seuil d'énergie de GVF spécifiques sont présentés. Pour un seuil de 20%, le contour actif ne permet pas d'avoir des points dans toutes les parties des objets. C'est notamment le cas de l'armoire (3.24a) dans le coin en bas à droite, ainsi que du fauteuil (3.24f) au niveau de l'accoudoir de gauche, où la majorité des points sont très regroupés et ne sont pas arrêtés sur le premier gradient qu'ils rencontrent. Les points du contour actif se déplacent au sein même du contour de l'objet pour trouver le minimum local.

Pour un seuil de 40%, les points sont bien mieux répartis et sont arrêtés sur l'arête de l'objet. Cependant, il est à noter que les points sont trop retenus à l'intérieur de l'objet (à gauche du fauteuil et sur le coté droit du meuble) en raison du bruit de l'image même s'il n'existe pas de véritable gradient visible.



(a) Seuil à 20 % (b) Seuil à 30 % (c) Seuil à 40 %

(d) Seuil à 20 % (e) Seuil à 30 % (f) Seuil à 40 %

FIGURE 3.23 – Image des pourcentages d'importance des forces GVF dans l'énergie totale du déplacement. En dessous du seuil (ici entre 20, 30 et 40 %), le déplacement est mis à 0.

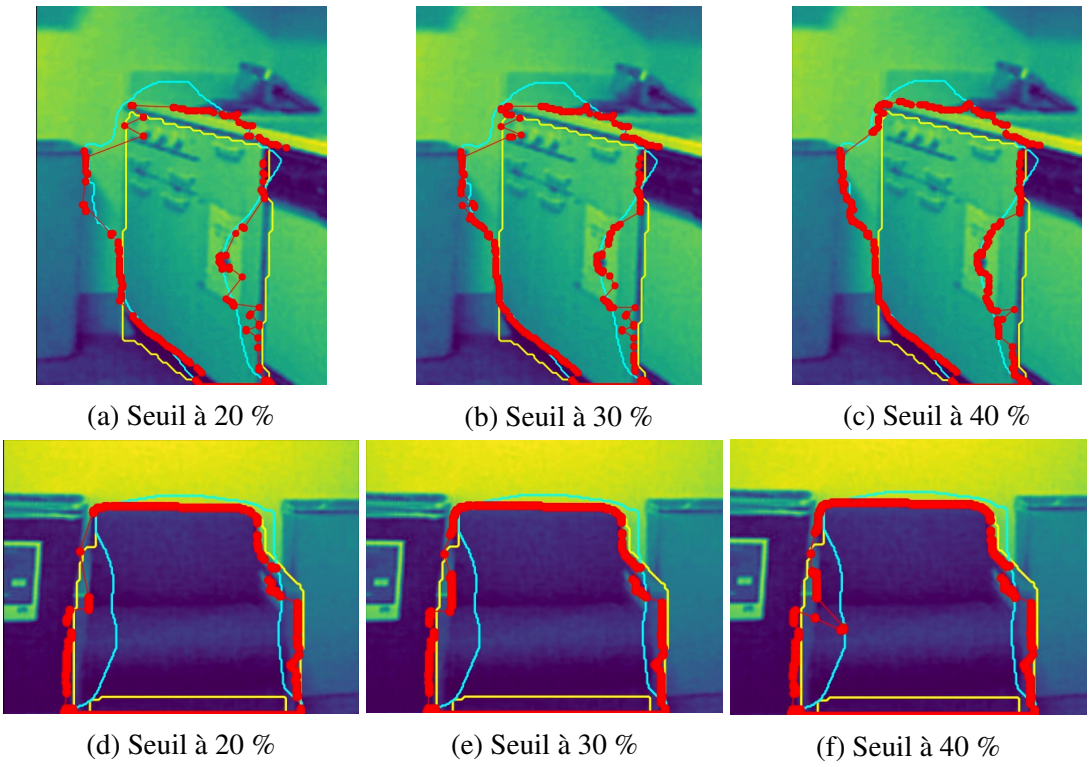
Le seuil de 30% permet de minimiser les points qui sont accrochés par les gradients à l'intérieur de l'objet. De plus, les points sont arrêtés sur l'arête de l'objet. Cette valeur de seuil est choisie pour la suite.

3.4.4.3 Amélioration des contours actifs

Étant donné que cette limitation peut induire que les contours actifs puissent ne pas aller jusqu'au milieu du contour, une dernière étape est ajoutée. Cette étape est le déplacement des points du contour sans restriction de pourcentage de GVF, mais pour très peu d'itérations. Dans notre cas, cinq itérations seront utilisées.

3.5 Analyse

Pour analyser la méthode proposée, il est nécessaire d'obtenir les données initiales de plusieurs éléments, comme la base de données, le CNN et la base de test. La base de données test, NYUV2, composée de plus de 1000 images, elles-mêmes composées d'une dizaine d'objets, est utilisée. Cependant, cette base de test sera filtrée pour réduire



(a) Seuil à 20 % (b) Seuil à 30 % (c) Seuil à 40 %

(d) Seuil à 20 % (e) Seuil à 30 % (f) Seuil à 40 %

FIGURE 3.24 – Images du résultat du contour actif en fonction du seuil du pourcentage de l'énergie GVF (ici entre 20, 30 et 40 %) limitant le déplacement de ces points (ici entre 20, 30 et 40 %)

le nombre de tests à réaliser et pour ne prendre que les objets bien définis et exploitables.

3.5.1 Base de données de test

Le premier élément est la base de données contenant des images couleur, de profondeur et labélisées. La base NYUv2 a été choisie car elle est uniforme en termes de résolution d'images et de qualité au niveau des images de profondeur. En effet, l'ensemble de la base a été réalisé avec le même capteur (voir 3.1.4.2). Le deuxième élément est le choix du CNN. Ce choix s'appuie sur l'IoU moyen de celui-ci. Comme précisé précédemment, le CNN GAD est choisi car il permet d'obtenir l'IoU le plus haut sur cette base d'images (voir 3.4.1). Les images de sortie du CNN sont enregistrées et serviront pour les contours actifs.
Les contours actifs traitent chaque objet de manière indépendante. Étant donné qu'une dizaine d'objets sont généralement contenus dans une image et qu'un ensemble de forces va être testées et analysées sur chaque objet, une sous base de données est préparée. Un premier échantillon de 10 objets par classe, pour au total 37 classes, est sélectionné. Les objets sont les 10 premiers rencontrés dans la base qui correspondent

réellement à l'objet que l'on veut segmenter. En effet, il ne s'agit pas d'analyser les erreurs du CNN mais le gain qu'apporte la solution aux objets segmentés par le CNN. Pour cela, le calcul de l'IoU de la zone de l'objet segmenté par le CNN et la vérité terrain est réalisé. Les objets ayant + de 40% sont gardés pour l'analyse finale.

Cet échantillon est réduit pour prendre en compte plusieurs contraintes :

1. L'image de profondeur et de couleur correspond bien au même objet que celui analysé par l'algorithme (3.25a) ;
2. L'objet doit être reconnaissable dans l'image traitée par le contour actif(3.25b) ;
3. La zone de l'objet de la vérité terrain ne doit pas être coupée en deux par le CNN car cela engendre des erreurs d'initialisation très importantes qui ne reflètent pas l'amélioration du réseau (3.25c) ;
4. La vérité terrain doit être correcte, c'est-à-dire proche du bord de l'objet et/ou n'englobant pas différents types d'objets. Plus l'objet est grand, plus la tolérance sera importante car peu de pixels seront concernés par rapport au nombre total de pixels de l'objet (3.25d).

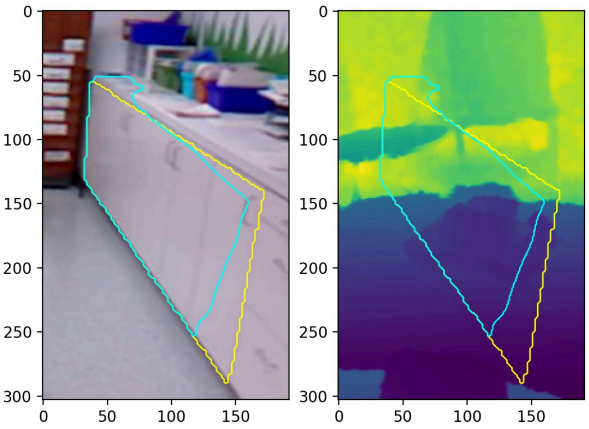
Toutes ces contraintes permettent d'éliminer un maximum d'objets "corrompus", c'est-à-dire ceux qui, à cause d'une erreur humaine ou du CNN, ne seront pas représentatifs de la méthode proposée combinant CNN et contours actifs. La Figure 3.25 illustre tous les types d'exemples qui ne correspondent pas aux contraintes de la base de données d'analyse.

3.5.2 Méthodologie de l'analyse des résultats

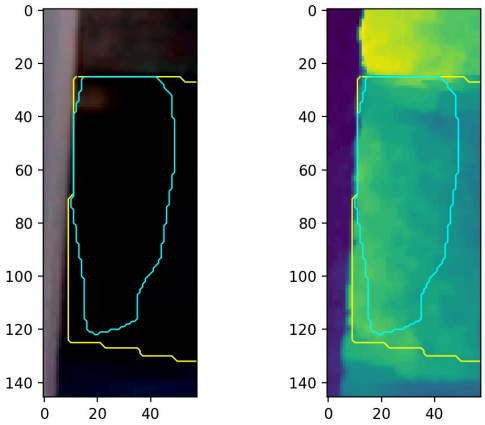
Le but de l'analyse est de savoir si la méthode permet d'améliorer ou non la segmentation finale de l'objet. Pour cela, les données pouvant être récupérées sont l'IoU de la zone de sortie de CNN par rapport à la vérité terrain, ainsi que l'IoU de la zone de sortie de la méthode proposée (CNN+CA) par rapport à la vérité terrain. Plusieurs cas sont présents :

1. Un IoU CNN élevé avec un IoU CNN+CA plus faible
2. Un IoU CNN élevé avec un IoU CNN+CA plus élevé
3. Un IoU CNN moyen avec un IoU CNN+CA plus faible
4. Un IoU CNN moyen avec un IoU CNN+CA plus élevé

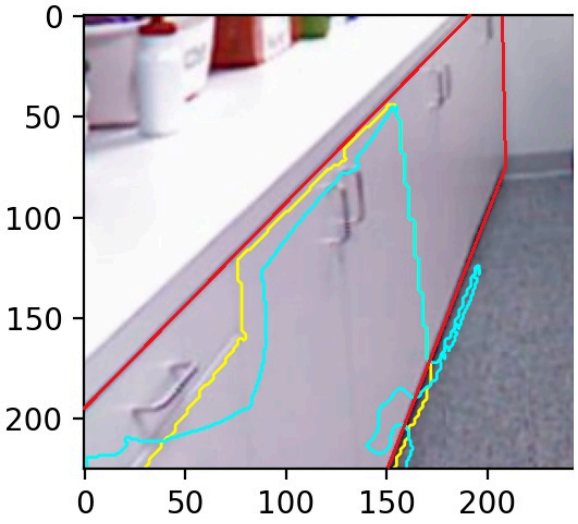
Des exemples de ces cas sont visibles dans le Tableau 3.4 et illustrés dans la Figure 3.26.



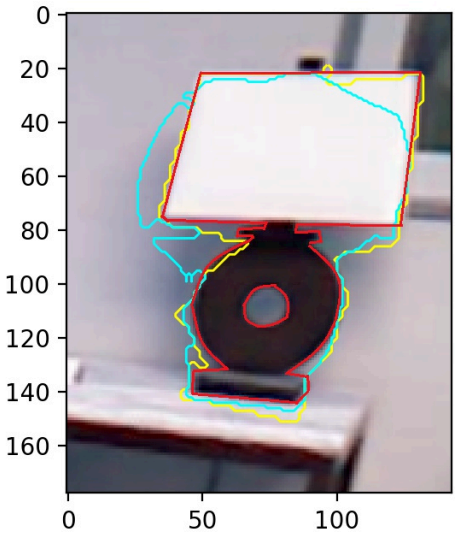
(a) Images couleur et profondeur décorrélées (profondeur représentant une table et deux chaises)



(b) Objet non reconnaissable (ici, un meuble)



(c) Vérité terrain coupée en plusieurs objets (CNN)



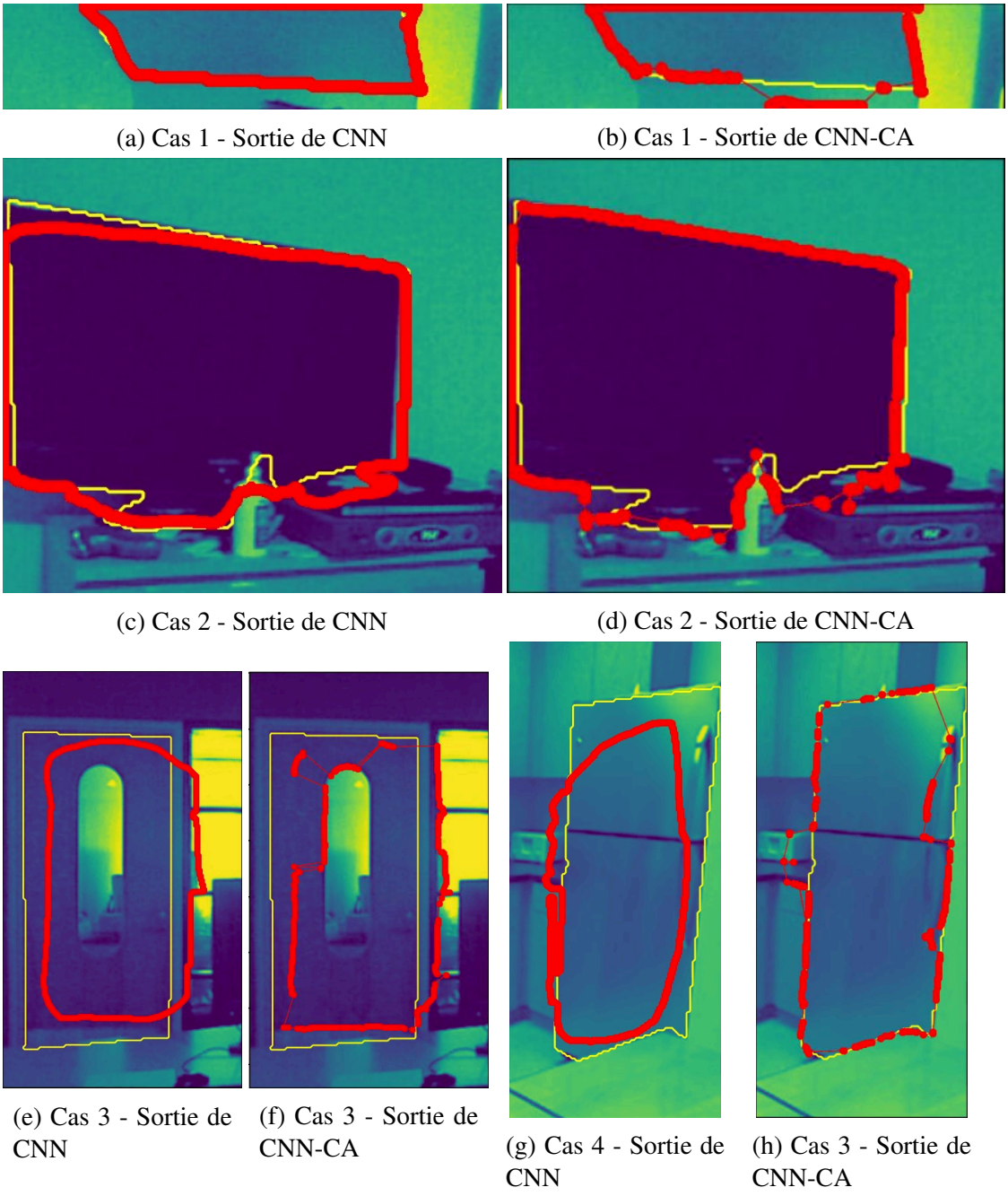
(d) Vérité terrain contenant des erreurs (la vérité terrain n'est pas superposée aux contours de l'objet)

FIGURE 3.25 – Exemples d'images ne correspondant pas aux contraintes de la base de données d'analyse. La courbe jaune représente la vérité terrain, la courbe bleue, la sortie du CNN et la courbe rouge, la segmentation modifiée au plus proche du contour de l'objet de la scène.

TABLE 3.4 – IoU des exemples d'objets en sortie de CNN et de la méthode CNN-CA

Cas	1	2	3	4
IoU CNN	90	80	60	55
IoU CNN-CA	70	85	55	85
Gain / Perte CNN-CA	-20	5	-5	20

Étant donné qu'il est compliqué de prendre directement l'IoU final car l'IoU en lui-même ne permet pas de savoir si la méthode fonctionne et améliore la segmentation, la



(a) Cas 1 - Sortie de CNN

(b) Cas 1 - Sortie de CNN-CA

(c) Cas 2 - Sortie de CNN

(d) Cas 2 - Sortie de CNN-CA

(e) Cas 3 - Sortie de CNN

(f) Cas 3 - Sortie de CNN-CA

(g) Cas 4 - Sortie de CNN

(h) Cas 3 - Sortie de CNN-CA

FIGURE 3.26 – Exemples des différents cas rencontrés de résultat d'IoU en sortie de CNN et de la méthode CNN-CA. Le trait jaune représente la vérité terrain et en rouge la sortie du CNN ou de la méthode CNN-CA.

différence entre l'IoU issu de la sortie du CNN et l'IoU issu de la sortie de la méthode CNN-CA est choisie. Cette différence est le gain, ou la perte, entre les deux sorties.

La première métrique utilisée est le pourcentage d'objets améliorés grâce à la méthode proposée. Pour cela, on compte le nombre d'objets où la différence d'IoU est positive et

TABLE 3.5 – Analyse résultat

	IoU objets améliorés			IoU objets dégradés			IoU objets tous		
Forces	Moyenne	Max	Ecart Type	Moyenne	Min	Ecart Type	Moyenne	Ecart type	Pourcentage objet amélioré
VT	7,31	25,77	6,00	-9,11	-40,44	8,49	1,03	10,65	61,8%
CNN	6,31	24,21	5,42	-6,00	-30,03	6,62	1,98	8,29	64,8%

l'on divise par le nombre total d'objets.

La moyenne et l'écart type de l'ensemble des objets analysés permettent de savoir si, en général, la méthode améliore l'IoU et si les gains sont proches de cette moyenne ou si, au contraire, les résultats peuvent être fortement différents.

Toutefois, il n'est pas possible avec ces métriques de savoir précisément comment la méthode CNN+CA réagit lorsque celle-ci améliore ou dégrade la segmentation des objets. Pour cette raison, les objets sont scindés en deux groupes pour être analysés séparément : les objets ayant leur segmentation améliorée par la méthode proposée et ceux dont la segmentation est détériorée. La moyenne et la variance pour chaque groupe sont calculées ainsi que la valeur de gain maximal ou de perte maximale quand la segmentation est respectivement améliorée ou détériorée.

3.5.3 Analyse de la similarité des résultats des forces issues de la régression des données du CNN et de la vérité terrain

La première étape de l'analyse est de vérifier que l'estimation des forces grâce à la régression des données issues de la sortie du CNN est proche de l'estimation des forces grâce à la régression de la vérité terrain issue du label de la base de données. La régression des forces grâce à l'image labélisée estime les coefficients des forces de gradient et de GVF qui amènent au maximum vers le contour de l'objet traité. La régression des forces grâce à la sortie du CNN estime les coefficients des forces de gradient et de GVF trouvés autour de cette sortie. La deuxième solution est donc moins spécifique à l'objet car elle peut prendre des gradients d'objets annexes ou de zones saillantes comme vérité terrain de l'objet.

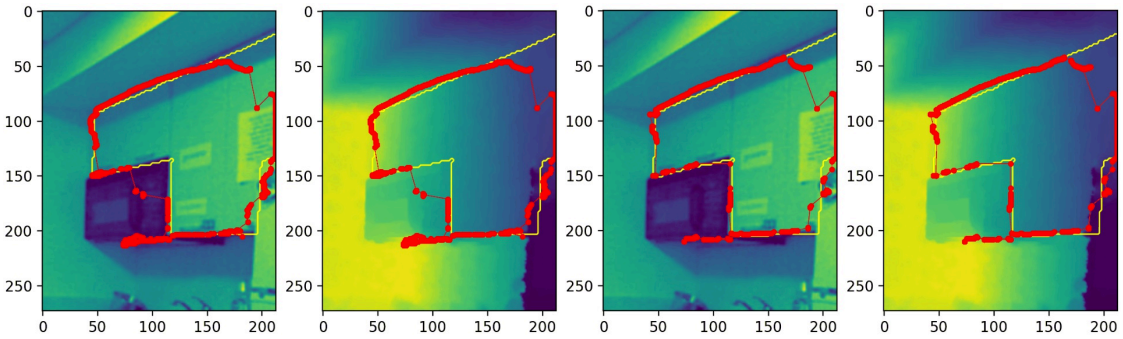
Le Tableau 3.5 est un récapitulatif des résultats obtenus après la méthode (CNN+CA) avec les forces estimées avec la régression issue du CNN et de l'image labélisée. Plusieurs éléments définis ci-après montrent que la régression issue du CNN permet d'ob-

tenir sensiblement le même résultat. Le premier élément à noter est que le pourcentage d'objets mieux segmentés avec la méthode est sensiblement le même avec 62% pour la régression issue de l'image labélisée et 65% pour la régression issue du CNN. De plus, l'IoU VT est supérieur de 1 % par rapport à l'IoU CNN.
Ces deux caractéristiques peuvent s'expliquer aisément. En effet, les forces VT sont issues de ce que l'on veut obtenir. Cela permet de contraindre fortement la régression à maximiser le résultat des forces, d'où un IoU et un maximum supérieur pour les objets bien segmentés de la VT. Cependant, étant donné que la labélisation n'est pas toujours parfaite, les forces peuvent apprendre sur les mauvais contours. Cela implique que les forces ne seront peut-être pas les bonnes. Cela impacte donc le nombre d'objets améliorés, ainsi que l'IoU moyen des objets dégradés dont le minimum est beaucoup plus bas (-40,44) par rapport aux CNN (-30,03).
Concernant les résultats issus du CNN, les forces sont calculées grâce à la zone de confiance autour de la zone prédite par le CNN. Cela implique que des gradients n'étant pas le contour de l'objet sont considérés comme le contour de l'objet. Le gain d'IoU n'est donc pas optimisé pour obtenir la segmentation finale, mais permet de prendre les contours réels et donc de s'affranchir des erreurs dues à une labélisation erronée. Pour cette raison, il est possible de remarquer que le maximum et la moyenne du différentiel d'IoU des objets améliorés par la méthode sont inférieurs à ceux de la VT. Et même, au contraire, la moyenne du différentiel d'IoU des objets détériorés par la méthode est supérieure à la vérité terrain. Le résultat de la segmentation sera d'autant plus proche de la vraie vérité terrain que la sortie du CNN sera près de celle-ci et que l'indice de confiance, c'est-à-dire l'IoU de la classe, sera haut.

L'estimation des forces grâce à la régression utilisant en entrée l'image de sortie du CNN permet d'obtenir un résultat cohérent et assez performant qui tend à être amélioré.

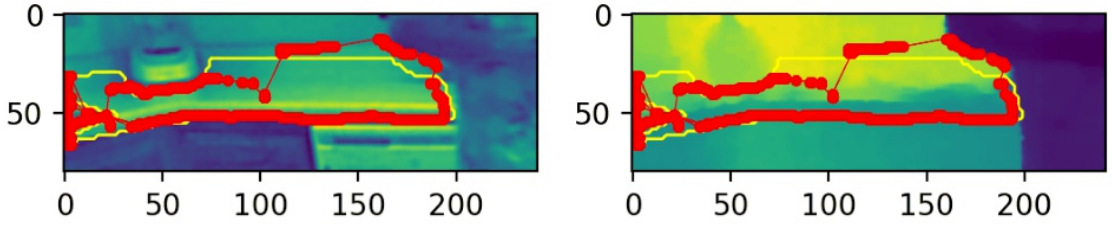
Dans la Figure 3.27, les deux exemples illustrent bien les cas expliqués précédemment. Dans le premier exemple, la vérité terrain est bien définie (3.27b) ce qui permet une segmentation finale meilleure que celle prédite par la méthode avec le CNN (3.27a). Dans le cas des forces issues du CNN, la courbe s'attache au gradient proche (ici un gradient issu de la profondeur), alors que pour les forces issues de la VT, la courbe s'attache plutôt aux gradients de couleur (RGB).

Dans le deuxième exemple, la vérité terrain n'est pas sur le contour de l'objet (3.27d). Cela induit une moins bonne segmentation finale. En effet, la vérité terrain n'est pas sur le contour du mur, qui est défini par un gradient de couleur. Les forces de la vérité terrain n'ont donc pas suffisamment pris en compte les gradients de couleur mais beaucoup plus les gradients de profondeur contrairement aux forces issues du

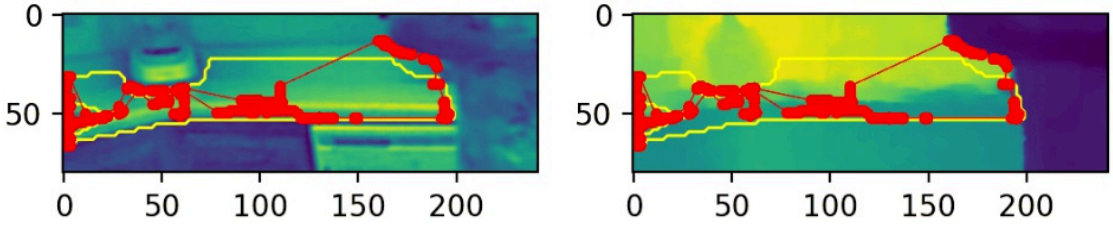


(a) Forces issues du CNN

(b) Forces issues de la VT



(c) Forces issues du CNN



(d) Forces issues de la VT

FIGURE 3.27 – Deux exemples d'objets testés avec les forces issues de la vérité terrain (VT) et du CNN. La courbe jaune représente la vérité terrain et la courbe rouge, la segmentation en sortie de la méthode en fonction des forces utilisées.

CNN (3.27c).

3.5.4 Analyse des résultats de la méthode grâce aux forces issues de la méthode brut de forces et de la méthode par régression(CNN)

La méthode retenue est l'estimation des forces grâce à une régression. Cependant, comme vu dans l'état de l'art 3.4.3.2, une autre méthode peut être utilisée : la méthode brut de force. Dans celle-ci, deux combinaisons de forces sont prises en compte :

— la combinaison activant avec le même coefficient toutes les forces externes ;

— la combinaison activant avec le même coefficient toutes les forces externes sauf la force GVF de l'image RGB-D.

TABLE 3.6 – Analyse résultat CNN et Brut de Forces

	IoU objets améliorés			IoU objets dégradés			IoU objets tous		
Forces	Moyenne	Max	Ecart Type	Moyenne	Min	Ecart Type	Moyenne	Ecart type	Pourcentage objet amélioré
BF 1 (tout)	6,07	22,76	5,12	-4,42	-21,89	-0,02	3,22	6,82	72,8%
BF 2 (tout sauf GVF RGBD)	6,12	22,74	5,27	-4,83	-25,06	-0,01	3,14	7,15	72,8%
CNN	6,31	24,21	5,42	-6,00	-30,03	-0,05	1,98	8,29	64,8%

Les forces sélectionnées en brut de forces (BF) ont été choisies sur plusieurs centaines de combinaisons exécutées sur un important panel d'objets de la base de données. Comme annoncé par l'état de l'art, ce processus est très long.

A la différence des forces issues du CNN, ces forces ne sont pas estimées mais sont les forces qui permettent d'avoir le meilleur résultat. Elles ont donc un lien direct avec l'IoU final de la méthode sur une base de données bien définie.

Le tableau ci-dessus (Tableau 3.6) montre tout d'abord que les forces BF permettent d'améliorer plus d'objets par rapport aux forces issues des CNN (8% en plus). La moyenne et le maximum des objets améliorés de ces forces sont légèrement plus faibles. En effet, les forces BF ne sont pas spécifiques à un objet en particulier mais sont spécifiques pour une base (comme dans ces tests) ou une classe. Cela implique que ces forces ne peuvent pas maximiser le gain d'IoU pour certains objets et minimiser la perte pour d'autres. La moyenne de l'IoU des objets dégradés confirme cela.

En conclusion les forces utilisant la méthode brut de force permettent d'améliorer plus d'objets que les forces issues de la régression, dans cette bas de données. Cependant, en contrepartie, ils ne permettent pas d'améliorer de façon optimale la segmentation de ces objets. Cette méthode demande énormément de ressources en amont et ne permet pas une mise en place rapide d'un système. En effet, il faut au préalable tester toutes les combinaisons, lesquelles sont généralement limitées, pour éviter un trop grand nombre de calculs, sur une multitude d'objets et ce afin d'identifier la ou les forces qui permettent de maximiser le résultat.

En fonction de la méthode d'estimation des forces utilisées lors de la régression, le nombre d'objets améliorés ainsi que les moyennes peuvent évoluer et tendre vers la moyenne trouvée avec les forces BF.

TABLE 3.7 – Analyse résultat type fonction de coût et annulation partie GVF des forces issues du CNN

	IoU objets améliorés			IoU objets dégradés			IoU objets tous		
Forces	Moyenne	Max	Ecart Type	Moyenne	Min	Ecart Type	Moyenne	Ecart type	Pourcentage objet amélioré
L1 75% GVF	6,17	23,42	5,23	-5,45	-32,55	-0,08	2,28	7,96	66,5%
L1 100% GVF	6,25	23,38	5,32	-5,65	-27,53	-0,07	2,49	7,93	68,4%
MS 75% GVF	6,54	25,01	5,57	-6,40	-29,51	-0,03	1,61	8,65	61,7%
MS 100% GVF	6,30	25,02	5,56	-6,51	-30,54	-0,02	1,54	8,62	62,6%

3.5.5 Analyse des différentes méthodes de régression

Comme présenté dans le paragraphe précédent (3.4.3.2), deux types de fonction de coût sont testées. Les fonctions de coût permettent de calculer l'erreur et ainsi d'impacter la variation des forces pour tendre vers la vérité terrain. De plus, l'annulation ou non des GVF trop faibles (les 75% les plus faibles) est également testée.

Le tableau 3.7 montre que les fonctions de coût L1 permettent d'obtenir un pourcentage plus élevé d'objets où la segmentation est améliorée à l'issue de la méthode proposée. Cela provient du fait que, par rapport à la métrique *Mean Square* (MS), cette métrique pénalise peu les erreurs et prend plus en compte les bons résultats. Il faut cependant noter que les forces utilisant la fonction de coût *Mean Square* obtiennent, pour les objets améliorés, un IoU moyen et maximal supérieur aux forces utilisant la fonction de coût L1. Cela vient du fait que lorsque l'objet est simple et/ou que l'initialisation est proche de la vérité terrain, cette métrique permet de pénaliser fortement les quelques erreurs présentes et ainsi maximiser les forces pour que le contour actif converge plus vers la vérité terrain.

L'annulation des GVF les plus faibles, c'est à dire les GVF qui sont les plus éloignés d'un contour, est surtout utile avec la métrique Mean Square car elle permet de limiter l'impact de cette métrique au GVF les plus proches. Cela se voit avec l'IoU moyen des objets améliorés (+6,54%) qui est supérieur à celui prenant tous les GVF (+6,30%). Cependant, en général, il ne permet pas de segmenter plus d'objets que la métrique L1.

La solution finale proposée dans la méthode est de pouvoir combiner deux de ces quatre types de forces. Lorsque l'IoU de la classe est faible (inférieur à 90%), la fonction de coût choisie est L1 avec 100% des GVF. Cela permet de maximiser le nombre d'objets améliorés ainsi que le gain de l'IoU final. Si l'IoU de la classe n'est pas connu et que l'indice de confiance est mis aléatoirement, il faut utiliser cette combinaison.

Lorsque l'IoU de la classe est supérieur à 90%, la fonction de coût *Mean Square* est choisie avec 75% des GVF. Étant donné que les sorties du CNN sont proches de la vérité terrain et que la zone de confiance est peu large, cette combinaison est choisie pour maximiser le gain et arriver au plus proche de la vérité terrain de l'objet.

Cet ensemble de combinaisons en fonction de l'IoU de la classe n'a cependant pas pu être testé car l'IoU des résultats du CNN était trop faible (inférieur à 70%).

3.5.6 Exemples de résultats de la méthode proposée

Les résultats de la méthode choisie sont représentés en partie pour quelques objets dans la Figure 3.29 . Ils montrent bien l'amélioration de la segmentation grâce à la méthode par rapport à la segmentation initiale issue du CNN.

3.6 Conclusion et Perspectives

Les travaux réalisés et présentés précédemment montrent que la méthode proposée, c'est-à-dire l'association d'un CNN suivi d'un contour actif utilisant tous les deux des images RGB-D, apporte une réelle amélioration dans la segmentation d'objets par rapport à l'image segmentée obtenue en sortie de CNN. En effet, la méthode utilise la sortie des CNN de segmentation RGB-D pour pouvoir initialiser la courbe du contour actif, lequel permet d'améliorer cette segmentation. Le processus global est représenté dans la Figure 3.28.

Le contour actif s'appuie sur les images couleur (RGB) et de profondeur (D) pour pouvoir améliorer au mieux la segmentation. La méthode montre que le processus est autonome et ne nécessite pas d'action de la part d'un utilisateur pour obtenir une segmentation fine. Cependant, il est à noter que cette méthode est très dépendante des objets à analyser ainsi que de l'initialisation de sortie du CNN. Les objets complexes, c'est-à-dire contenant peu de gradients ou alors étant composés de plusieurs régions très contrastées, tant en termes de profondeur que de couleur, peuvent limiter l'impact de la méthode voire, dans certains cas, détériorer le résultat final. Enfin, l'initialisation issue du CNN peut aussi impacter le résultat de la méthode car si l'initialisation est trop loin de l'objet, le contour pourra être attiré ou stoppé par d'autres objets.

Cette méthode est donc très efficace pour améliorer la segmentation des objets relativement simples en termes de caractéristiques bas niveau (gradient, texture) mais aussi dans des cas complexes. Cette méthode a aussi l'avantage de pouvoir s'adapter à des

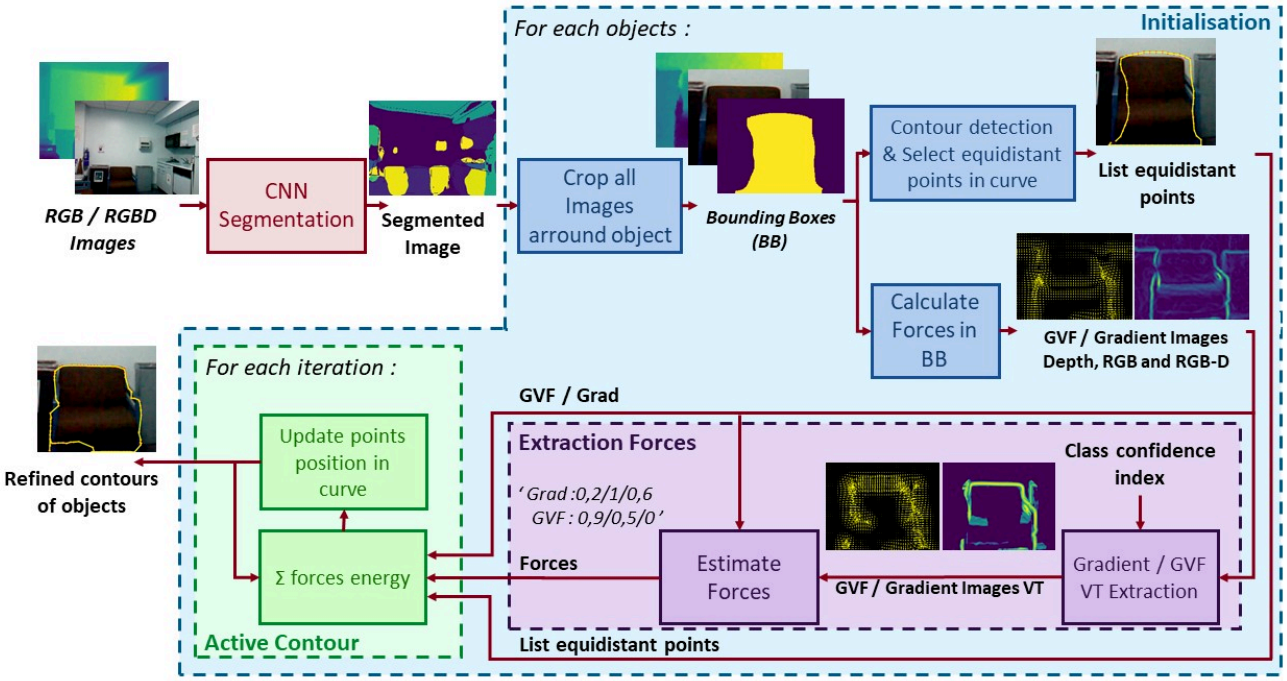
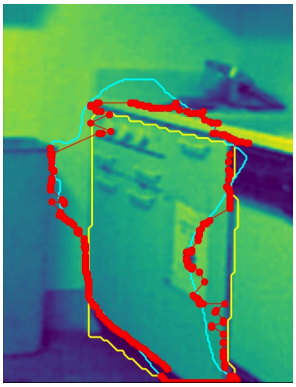


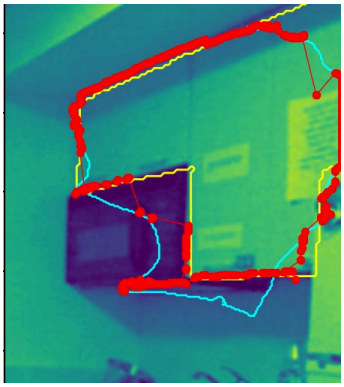
FIGURE 3.28 – Schéma global détaillé de la solution proposée

images haute résolution ce qui sera exploité dans le chapitre 5.1, lors de son application sur des images de pièces de bus.

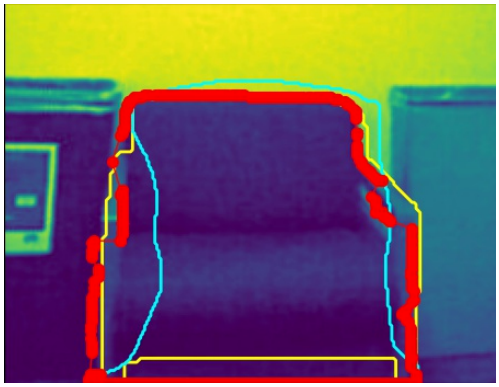
Plusieurs pistes d'améliorations sont à noter, comme la fusion de l'ensemble des segmentations améliorées pour pouvoir recomposer une seule image labélisée ou alors l'utilisation de réseaux de neurones pour apprendre les forces à extraire sans passer par une régression qui ne prend pas en compte les expérimentations précédentes.



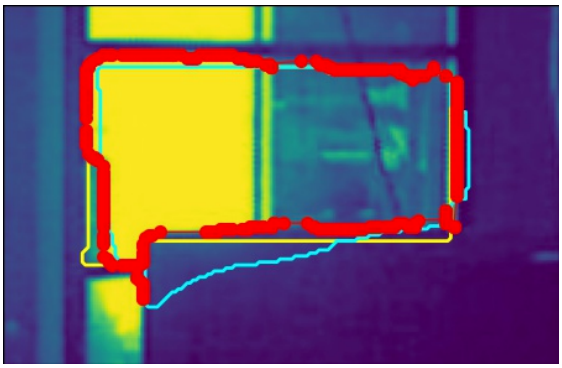
(a) Meuble



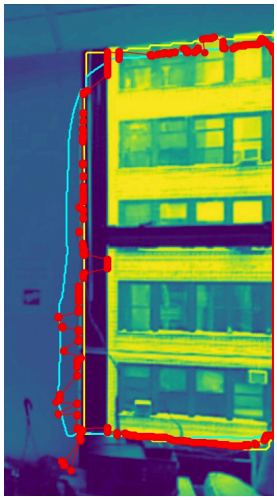
(b) Meuble



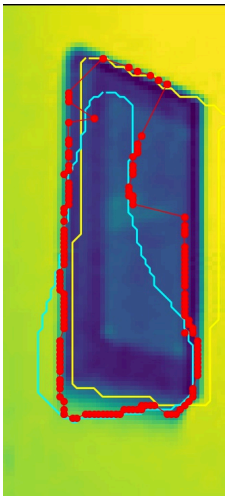
(c) Fauteuil



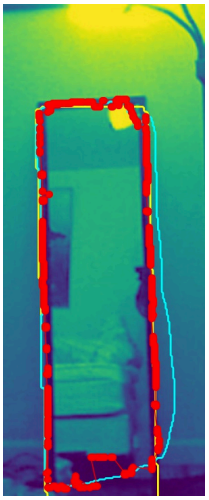
(d) Vitre



(e) Vitre



(f) Store



(g) Mirroir

FIGURE 3.29 – Résultat de la méthode (CNN+CA). La courbe en jaune représente la vérité terrain, la courbe bleue le résultat du CNN et la courbe rouge le résultat issu de la méthode.

BIBLIOGRAPHIE

[1] Majedaldein ALMAHASNEH et al. "MLMT-CNN for Object Detection and Segmentation in Multi-layer and Multi-spectral Images". In : *Machine Vision and Applications* 33 (jan. 2022). DOI : 10.1007/s00138-021-01261-y.

[2] Vijay BADRINARAYANAN, Alex KENDALL et Roberto CIPOLLA. "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation". In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.12 (2017), p. 2481-2495. DOI : 10.1109/TPAMI.2016.2644615.

[3] Lucia BALLERINI. "Genetic snakes: Active contour models by genetic algorithms". In : t. 8. Jan. 2007, p. 177-194.

[4] Walid BARHOUMI et Ezzeddine ZAGROUBA. "Evaluation de la segmentation des images médicales : une méthode supervisée utilisant conjointement l'information région et contour". In : jan. 2006, p. 43-46.

[5] Erik CARBAJAL-DEGANTE et al. "Active contours for multiregion segmentation with a convolutional neural network initialization". In : (avr. 2020), p. 6. DOI : 10.1117/12.2556928.

[6] Sihan CHEN et al. "Global-Local Propagation Network for RGB-D Semantic Segmentation". In : *CoRR* abs/2101.10801 (2021). arXiv : 2101.10801. URL : https://arxiv.org/abs/2101.10801.

[7] Xu CHEN et al. "Learning Active Contour Models for Medical Image Segmentation". In : (2019), p. 11624-11632. DOI : 10.1109/CVPR.2019.01190.

[8] Laurent D. COHEN. "On active contour models and balloons". In : *CVGIP: Image Understanding* 53.2 (1991), p. 211-218. ISSN : 1049-9660. DOI : https://doi.org/10.1016/1049-9660(91)90028-N. URL : https://www.sciencedirect.com/science/article/pii/104996609190028N.

[9] Simen Keiland FONDEVIK et al. "Image Segmentation of Corrosion Damages in Industrial Inspections". In : *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*. 2020, p. 787-792. DOI : 10.1109/ICTAI50040.2020.00125.

[10] Zhangxuan GU et al. "Hard Pixel Mining for Depth Privileged Semantic Segmentation". In : *IEEE Transactions on Multimedia* 23 (2021), p. 3738-3751. DOI : 10.1109/TMM.2020.3035231.

[11] Caner HAZIRBAŞ et al. "FuseNet: Incorporating Depth into Semantic Segmentation via Fusion-Based CNN Architecture". In : (nov. 2016). DOI : 10.1007/978-3-319-54181-5_14.

[12] Assaf HOOGI et al. "Adaptive Estimation of Active Contour Parameters Using Convolutional Neural Networks and Texture Analysis". In : *IEEE Transactions on Medical Imaging* 36.3 (2017), p. 781-791. DOI : 10.1109/TMI.2016.2628084.

[13] *Intersection over Union (IoU) for object detection*. en-US. Nov. 2016. URL : https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/ (visité le 07/07/2022).

[14] Jianbo JIAO et al. "Geometry-Aware Distillation for Indoor Semantic Segmentation". In : (2019), p. 2864-2873. DOI : 10.1109/CVPR.2019.00298.

[15] Asako KANEZAKI. "Unsupervised Image Segmentation by Backpropagation". In : *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2018), p. 1543-1547.

[16] Michael KASS, Andrew WITKIN et Demetri TERZOPOULOS. "Snakes: Active contour models". In : *INTERNATIONAL JOURNAL OF COMPUTER VISION* 1.4 (1988), p. 321-331.

[17] Youngeun KIM et al. "CNN-Based Semantic Segmentation Using Level Set Loss". In : (2019), p. 1752-1760. DOI : 10.1109/WACV.2019.00191.

[18] Adrian KUCHARSKI et Anna FABIJAŃSKA. "CNN-watershed: A watershed transform with predicted markers for corneal endothelium image segmentation". In : *Biomedical Signal Processing and Control* 68 (juil. 2021), p. 102805. DOI : 10.1016/j.bspc.2021.102805.

[19] Kin-Man LAM et Heng YAN. "Fast greedy algorithm for active contours". In : *Electronics Letters* 30 (1994), p. 21-23.

[20] Fahad LATEEF et Yassine RUICHEK. "Survey on semantic segmentation using deep learning techniques". In : *Neurocomputing* 338 (2019), p. 321-348. ISSN : 0925-2312. DOI : https://doi.org/10.1016/j.neucom.2019.02.003. URL : https://www.sciencedirect.com/science/article/pii/S092523121930181X.

[21] Honglak LEE et al. "Unsupervised Learning of Hierarchical Representations with Convolutional Deep Belief Networks". In : *Commun. ACM* 54 (oct. 2011), p. 95-103. DOI : 10.1145/2001269.2001295.

[22] Chenglong LI et al. "Segmenting Objects in Day and Night: Edge-Conditioned CNN for Thermal Image Semantic Segmentation". In : *IEEE Transactions on Neural Networks and Learning Systems* PP (juil. 2020), p. 1-14. DOI : 10.1109/TNNLS.2020.3009373.

[23] Lei LI et al. "Atrial scar quantification via multi-scale CNN in the graph-cuts framework". In : *Medical Image Analysis* 60 (2020), p. 101595. ISSN : 1361-8415. DOI : https://doi.org/10.1016/j.media.2019.101595. URL : https://www.sciencedirect.com/science/article/pii/S1361841519301355.

[24] Di LIN et Hui HUANG. "Zig-Zag Network for Semantic Segmentation of RGB-D Images". In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42.10 (2020), p. 2642-2655. DOI : 10.1109/TPAMI.2019.2923513.

[25] Hongya LU et al. “A 3D Convolutional Neural Network for Volumetric Image Semantic Segmentation”. In : *Procedia Manufacturing* 39 (2019). 25th International Conference on Production Research Manufacturing Innovation: Cyber Physical Manufacturing August 9-14, 2019 | Chicago, Illinois (USA), p. 422-428. ISSN : 2351-9789. DOI : https://doi.org/10.1016/j.promfg.2020.01.386. URL : https://www.sciencedirect.com/science/article/pii/S2351978920304571.

[26] Ishani MISHRA et al. “Enhanced Framework for Semantic Segmentation of Agricultural Products”. In : *2022 International Conference on Advanced Computing Technologies and Applications (ICACTA)*. 2022, p. 1-5. DOI : 10.1109/ICACTA54488.2022.9753597.

[27] Pushmeet Kohli NATHAN SILBERMAN Derek Hoiem et Rob FERGUS. “Indoor Segmentation and Support Inference from RGBD Images”. In : (2012).

[28] Stanley OSHER et James A SETHIAN. “Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations”. In : *Journal of Computational Physics* 79.1 (1988), p. 12-49. ISSN : 0021-9991. DOI : https://doi.org/10.1016/0021-9991(88)90002-2. URL : https://www.sciencedirect.com/science/article/pii/0021999188900022.

[29] Kerem SAHIN et Ilkay ULUSOY. “Automatic multi-scale segmentation of high spatial resolution satellite images using watersheds”. In : *2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS*. 2013, p. 2505-2508. DOI : 10.1109/IGARSS.2013.6723330.

[30] Shuran SONG, Samuel P. LICHTENBERG et Jianxiong XIAO. “SUN RGB-D: A RGB-D scene understanding benchmark suite”. In : (2015), p. 567-576. DOI : 10.1109/CVPR.2015.7298655.

[31] Satoshi SUZUKI et Keiichi ABE. “Topological structural analysis of digitized binary images by border following”. In : *Comput. Vis. Graph. Image Process.* 30 (1985), p. 32-46.

[32] Quan TANG et al. “Attention-guided Chained Context Aggregation for Semantic Segmentation”. In : *CoRR* abs/2002.12041 (2020). arXiv : 2002.12041. URL : https://arxiv.org/abs/2002.12041.

[33] Yikai WANG et al. “Deep Multimodal Fusion by Channel Exchanging”. In : 33 (2020). Sous la dir. de H. LAROCHELLE et al., p. 4835-4845. URL : https://proceedings.neurips.cc/paper/2020/file/339a18def9898dd60a634b2ad8fbbd58-Paper.pdf.

[34] Chen XIAOKANG et al. “Bi-directional Cross-Modality Feature Propagation with Separation-and-Aggregation Gate for RGB-D Semantic Segmentation”. In : nov. 2020, p. 561-577. ISBN : 978-3-030-58620-1. DOI : 10.1007/978-3-030-58621-8_33.

[35] Chenyang XU et Jerry L. PRINCE. “Gradient vector flow: A new external force for snakes”. English (US). In : *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (jan. 1997). Proceedings of the 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition ; Conference date: 17-06-1997 Through 19-06-1997, p. 66-71. ISSN : 1063-6919.

[36] Jun XU et al. “Convolutional neural network initialized active contour model with adaptive ellipse fitting for nuclear segmentation on breast histopathological images”. In : *Journal of Medical Imaging* 6 (fév. 2019), p. 1. DOI : 10.1117/1.JMI.6.1.017501.

[37] Xiaojun YANG et Xiaoliang JIANG. “A Hybrid Active Contour Model based on New Edge-Stop Functions for Image Segmentation”. In : *International Journal of Ambient Computing and Intelligence* 11 (jan. 2020), p. 87-98. DOI : 10.4018/IJACI.2020010105.

[38] Yuchun YUE et al. “Two-Stage Cascaded Decoder for Semantic Segmentation of RGB-D Images”. In : *IEEE Signal Processing Letters* 28 (2021), p. 1115-1119. DOI : 10.1109/LSP.2021.3084855.

[39] Zhenyu ZHANG et al. “Pattern-Affinitive Propagation across Depth, Surface Normal and Semantic Segmentation”. In : *CoRR* abs/1906.03525 (2019). arXiv : 1906.03525. URL : http://arxiv.org/abs/1906.03525.

[40] Ling ZHOU et al. “Pattern-Structure Diffusion for Multi-Task Learning”. In : (2020).

CHAPITRE 4

EXPÉRIMENTATIONS ET ANALYSE

La problématique initiale est la segmentation de pièces de carrosserie d'autobus ou d'autocars sans erreur de mesure. Comme vu dans le chapitre 1.3, la solution proposée est composée de trois blocs, dont deux ont été mis en place à l'issue de cette thèse. Ce sont :

— le bloc permettant d'extraire la profondeur de la scène avec une faible erreur de mesure ;

— le bloc permettant de segmenter les pièces de carrosserie avec une faible erreur de segmentation.

Ces blocs, détaillés dans le chapitre 2 et le chapitre 3, sont développés pour des applications générales. Le but de ce chapitre est de tester les blocs et les méthodes associées sur des exemples de bus pour analyser les performances de la solution proposée sur un cas d'usage.

4.1 Génération d'images de synthèse

Comme vu dans le chapitre 1.3.3, aucune base de données ayant toutes les caractéristiques requises pour l'application n'est disponible. Cependant quatre problèmes ne permettent pas d'obtenir les données réelles nécessaires pour tester la solution :

1. le système d'acquisition n'est pas opérationnel ;

2. le temps pour obtenir tous les types de bus est important ;
3. le temps pour labéliser toutes les images est important ;
4. la segmentation réalisée par un humain contiendra de nombreuses erreurs.

Par conséquent, la solution retenue est de créer une base de données grâce à un logiciel de modélisation 3D. Ce logiciel sera utilisé pour la génération de la base de données nécessaire pour le bloc de segmentation des pièces, mais aussi pour le bloc permettant l'acquisition de la profondeur de l'arrière du bus. En effet, le système d'acquisition n'étant pas opérationnel, le logiciel permettra de simuler la projection des mires du projecteur sur le bus et ainsi de tester la méthode.

4.1.1 Outils

Blender est un logiciel libre de droits et open source qui permet de modéliser et d'animer des objets/modèles 3D dans un environnement que l'on peut programmer. Il dispose d'une API et d'une interface permettant d'exécuter des scripts Python afin d'automatiser des tâches.

La création d'une scène 3D est la toute première étape du processus. En effet, elle permet d'initialiser un contexte et des repères qui seront utiles pour tous les autres objets à intégrer dans cette scène. C'est à partir de cette étape que plusieurs éléments peuvent être ajoutés :

— Caméra ;
— Fond ;
— Lumière ;
— Modèles 3D ;
— Animation ;
— etc.

4.1.1.1 Scènes et modèles 3D

Dans l'application, les quatre premiers éléments sont utilisés pour générer les images nécessaires pour paramétrer puis tester les blocs fonctionnels développés.

La caméra est définie par un emplacement, une direction, une résolution, un angle de champ de vision et une méthode de projection. La projection perspective est choisie pour obtenir un rendu réaliste dans les images.

Le fond initial est un fond noir. Cependant, il est possible de mettre d'autres couleurs ou des images. Ainsi, quel que soit le point de vue de la scène, l'image de fond reste visible. De plus, elle permet que la scène semble réaliste sans qu'il y ait besoin de créer tout l'environnement autour.

Les lumières sont définies par un emplacement, une direction, un type de lumière, une intensité lumineuse et une couleur. Plusieurs types de lumière existent et permettent de modéliser plusieurs sources lumineuses rencontrées dans la réalité : des lampes, le soleil ou des lumières ambiantes par exemple. Un projecteur est défini comme une lumière avec une projection de motifs dans la scène.

Un modèle 3D est une représentation d'un objet, réel ou non, dans un espace 3D numérique. Les modèles 3D de bus, utilisés par la suite, ont été fournis par Segula, sur la base de fichiers CAO. Ces modèles 3D sont les modèles d'entrées qui seront utilisés pour générer les images de bus dans une scène (RGB-D et labelisée), que ce soit pour la validation de la méthode d'estimation de la profondeur et la méthode d'amélioration de la segmentation des pièces du bus. Tous les modèles 3D sont composés de points qui sont reliés entre eux par des triangles. Plus un objet contient de triangles, mieux il est défini et/ou complexe et sera ressemblant à la réalité (Figure 4.1). Ils sont stockés dans différents types de format tel que .3ds, .dae, .obj, .stl ou encore .fbx. Ces fichiers ont les coordonnées des points du modèle, l'association des points permettant de créer les triangles et pour certains, des informations complémentaires comme la texture, la normale ou le matériau associé à chaque triangle.

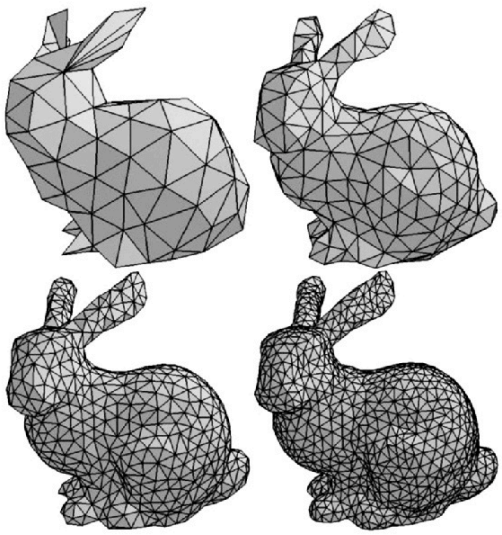
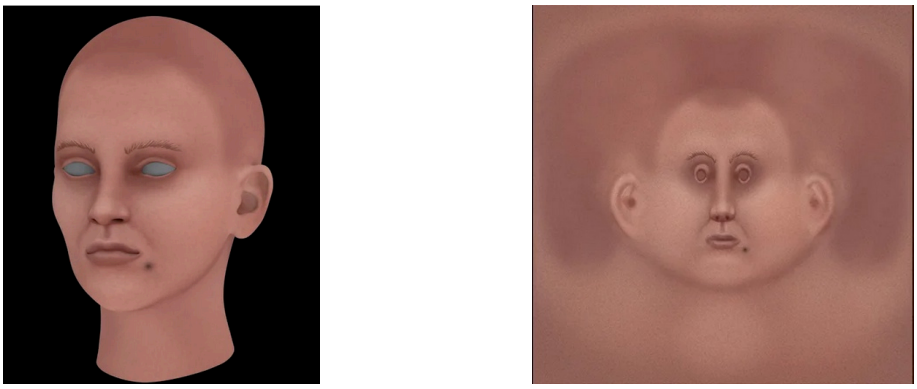


FIGURE 4.1 – Modèles 3D d'un lapin constitué d'un nombre plus ou moins important de triangles [2]

La géométrie de la forme est une partie importante du modèle 3D. Cependant, pour un rendu réaliste, la texture et le matériau associés sont également très importants.

La texture peut être une couleur et une image qui sera par la suite affectée sur les triangles de l'objet. Dans le cas d'une image, celle-ci est déformée du fait qu'elle est la représentation 2D d'une texture 3D (Figure 4.2). Pour affecter la texture sur les triangles, chaque point du modèle 3D a une coordonnée 2D dans l'image de texture. La texture d'un triangle est obtenue en projetant sur ce triangle la portion d'image définie par ces points 3D qui sont associés à des coordonnées 2D de l'image.



(a) Modèle 3D (b) Texture du modèle 3D

FIGURE 4.2 – Images du modèle 3D ainsi que de la texture projetée sur celui-ci

Le matériau d'un modèle 3D permet de déterminer les propriétés de surface du modèle, c'est-à-dire la réflexion de la lumière et la transparence de l'objet. La réflexion de la lumière permet de définir comment l'objet réagit à la scène, c'est-à-dire s'il reflète la scène intégralement, en partie ou pas du tout. La transparence de l'objet permet de définir si l'objet permet de voir un autre objet ou la scène à travers lui.

Ces deux caractéristiques se cumulent. A titre d'exemple, une vitre et une route sont considérées comme opposées en termes de matériaux. La vitre est transparente et reflète la scène, alors que la route est opaque et ne reflète pas la scène qui l'entoure.

L'ensemble de ces éléments permettent de créer des scènes proches de la réalité (Figure 4.4).

Cependant la partie réflexion des pièces et enregistrement des images sont gérés par le rendu graphique de Blender.

FIGURE 4.3 – Exemple de différents matériaux comme une route bitumée, des vitres ou des éléments de carrosserie

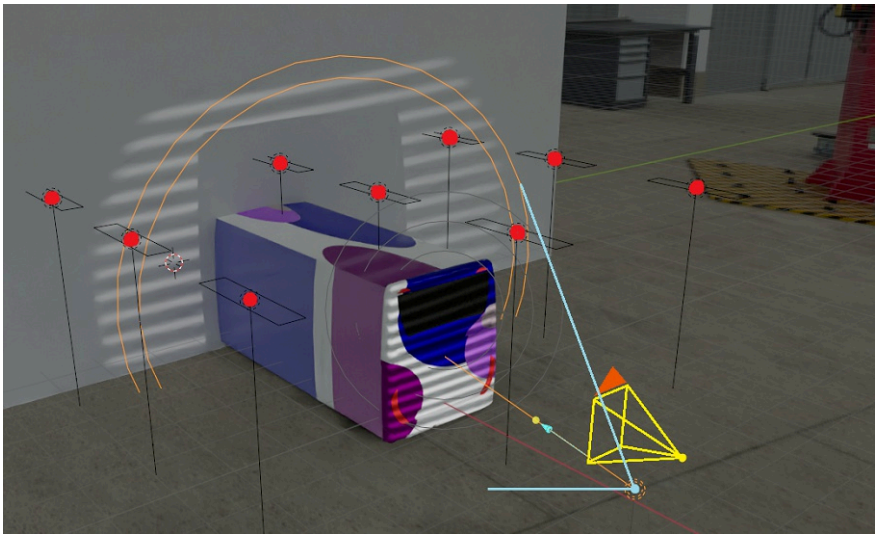


FIGURE 4.4 – Éléments de la scène 3D contenant le bus composé de lumière (rouge), d'un fond, d'une caméra (jaune) et du projecteur (bleu) affichant une mire sur le véhicule

4.1.1.2 Moteur de rendu graphique

Le rendu graphique permet de projeter la scène et toutes les réflexions des objets sur la caméra d'acquisition de la scène. Deux moteurs de rendu graphique sont disponibles sur Blender et utilisent deux méthodes différentes de rendu :

— le moteur "Eevee" utilise la méthode de rasterisation ;

— le moteur "Cycles" utilise la méthode de lancer de rayons.

La rasterisation est une méthode qui crée le rendu en projetant les modèles 3D sur le capteur. Les valeurs des pixels du rendu sont donc égales aux valeurs de la texture de l'objet le plus proche de la caméra.

Le lancer de rayons, comme vu dans le chapitre 2.1.3.1, part de la caméra, lance un

rayon pour chaque pixel et rebondit autant de fois que nécessaire. Cela permet d'avoir des reflets beaucoup plus qualitatifs et donc un rendu très réaliste. Cependant le temps de calcul est beaucoup plus long que la rasterisation dont les rendus sont de moins bonne qualité (Figure 4.5).

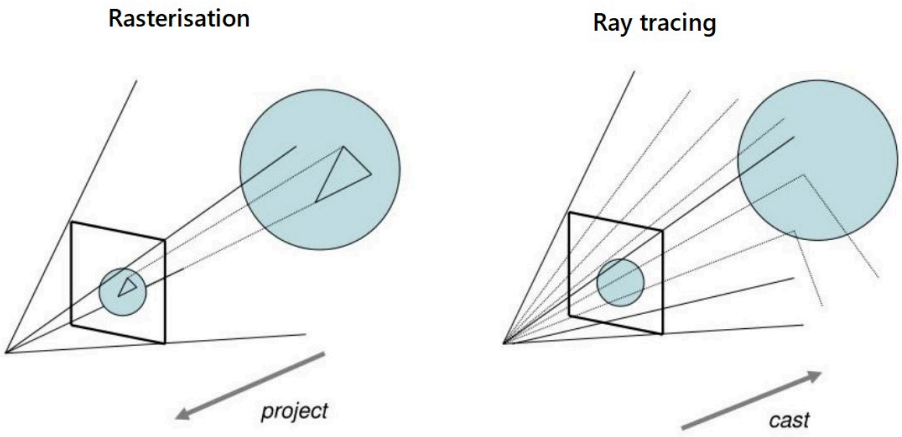


FIGURE 4.5 – Schéma synthétisant les deux méthodes de rendu graphique : rasterisation et lancer de rayon (raytracing) [1]

Le moteur de rendu graphique peut aussi être utilisé pour le projecteur. Cependant c'est le processus inverse car le but est de déterminer la texture à appliquer aux triangles d'un objet à partir des motifs projetés par le projecteur.

Comme précédemment, la rasterisation permet d'avoir un rendu rapide mais l'ombre n'est pas prise en compte alors que le lancer de rayons permet d'avoir un rendu réaliste de la projection du projecteur sur le modèle 3D, avec cependant un temps de calcul long. Par comparaison, la génération d'une image prenant 5 secondes avec la méthode Eevee, prend 35 minutes avec la méthode Cycles. La méthode Eevee est choisie comme moteur de rendu graphique pour générer les images pour la base de données pour la segmentation et pour les images permettant l'estimation de la profondeur.

4.1.2 Bloc acquisition de la profondeur

Comme son nom l'indique, le bloc d'acquisition de la profondeur permet d'estimer la profondeur de la scène pour chaque pixel grâce à la méthode décrite dans le chapitre 2.1. Pour cela, il est nécessaire de générer les images de synthèse des mires projetées sur le bus et les plans de référence, ainsi qu'une image de profondeur.

4.1.2.1 Paramètres de test

En entrée, le logiciel récupère les informations de l'utilisateur qui sont :

— l'angle de champs de vision du projecteur (51,3°) et de la caméra (43,6°) ;

— la résolution du projecteur (768 X 1024 pixels) ;

— l'erreur de mesure maximale ;

— la liste des profondeurs des plans de référence nécessaires pour obtenir une erreur de mesure maximale définie auparavant.

L'erreur de mesure déterminée par le cahier des charges est initialement à 0,33 mm. Cela revient à générer des images de résolution égale à 9000 X 12000 pixels. Pour cette résolution, il faudrait donc enregistrer 4437 images, sachant que chaque image nécessite 190 Mo de stockage de mémoire. Au total, 820 Go de mémoire seraient nécessaires pour enregistrer toutes les images des plans de référence et du bus. Il est donc convenu de se limiter à une résolution égale à 3600 x 4800. L'erreur de mesure testée au niveau de la profondeur est définie à 1,5 mm.

L'image de profondeur nécessaire pour pouvoir tester l'erreur de mesure de la méthode sur le bus sera enregistrée dans une image 16 bits. En effet, étant donné que l'erreur maximale souhaitée est égale à 1,5 mm et que la zone d'analyse est comprise entre 4250 et 5750 mm, il est nécessaire d'avoir au minimum 6000 valeurs, cela exclut le format 8 bits (256 valeurs). Le format 16 bits est lui composé de 65535 valeurs ce qui est suffisant pour enregistrer les profondeurs de la scène.

4.1.2.2 Projection des mires

La méthode a besoin des images des mires projetées sur l'objet. Pour cela, une méthode est mise en place sous Blender pour la projection des mires (Figure 4.12).

La première étape de la méthode de projection des mires consiste à récupérer les points et les triangles des modèles 3D sur lesquels les mires doivent être projetées. Ces points, ayant des coordonnées dans le repère monde, sont modifiés pour avoir des coordonnées 3D selon le repère projecteur. Pour cela, la position du projecteur est récupérée. Grâce aux points 3D des modèles 3D, des informations du projecteur, c'est-à-dire sa matrice intrinsèque composée de sa focale et son centre optique, et de l'image que le projecteur affiche sur l'objet, il est possible de réaliser un raycasting inverse, i.e. une projection de l'image sur le modèle 3D. Le motif projeté recouvre les motifs déjà présents sur l'objet. Pour ne pas les recouvrir, une fonction prenant en entrée la texture initiale

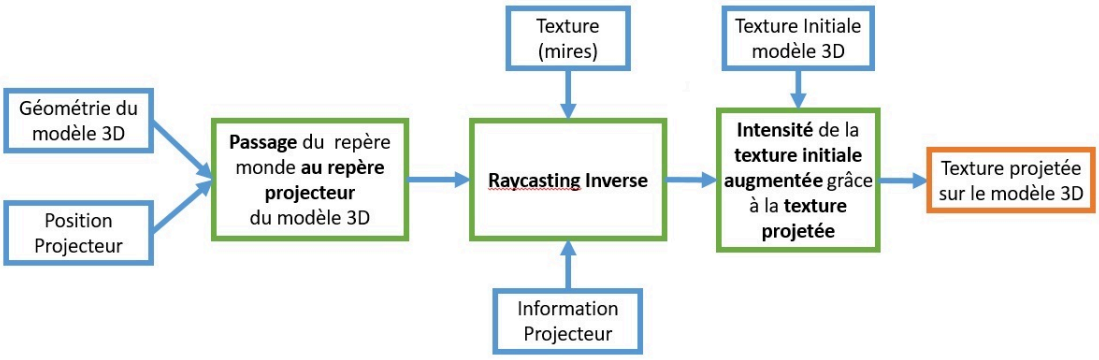


FIGURE 4.6 – Processus permettant de projeter une texture sur un modèle 3D. Les carrés en bleu sont les entrées des fonctions (carrés en vert). Les sorties sont représentées par des carrés orange.

de la pièce est ajoutée. Elle permet d'augmenter l'intensité de la texture initiale en fonction de l'intensité de la texture projetée. De la sorte, le projecteur est modélisé de façon réaliste.

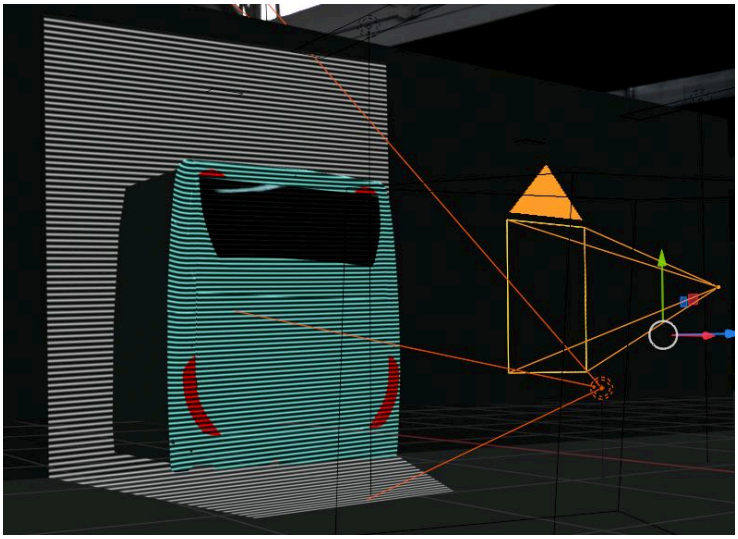
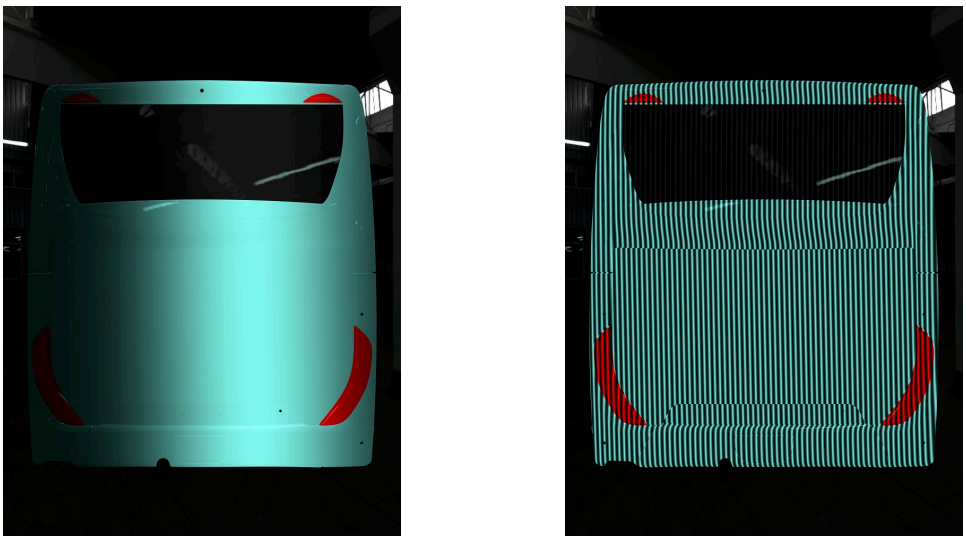


FIGURE 4.7 – Scène montrant la projection d'une mire sur un bus

Les images de la scène avec les motifs sont ensuite enregistrées. Il suffit pour cela d'acquérir l'image de la scène avec la caméra et de changer la texture utilisée lors du raytracing inverse avant chaque acquisition.

4.1.2.3 Acquisition des coordonnées de la scène

La méthode d'estimation de la profondeur a aussi besoin des coordonnées de la scène des plans de référence pour estimer la profondeur du bus.



(a) Image du bus avec la projection d'une mire de fréquence 1 sur celui-ci

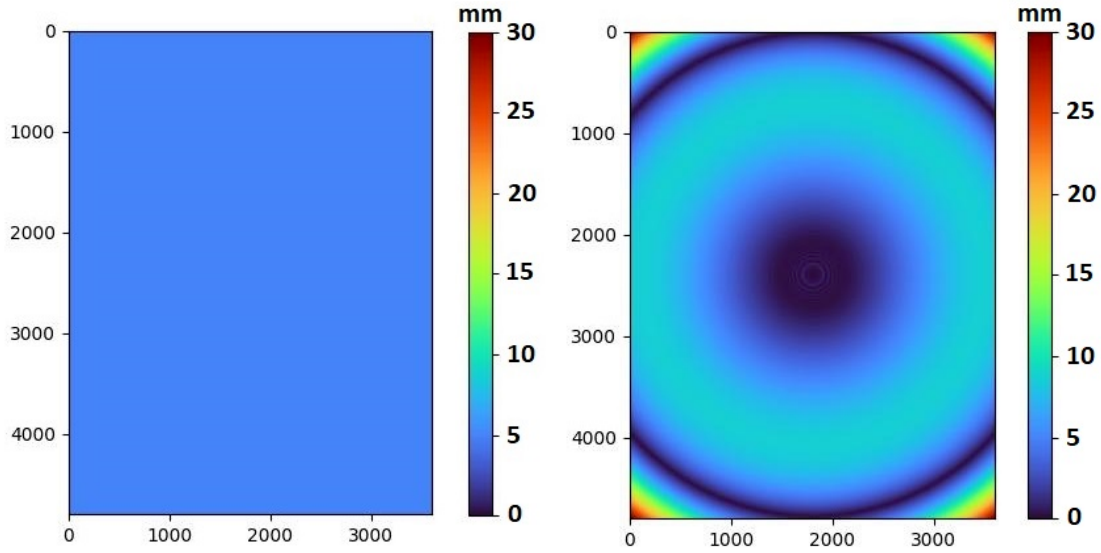
(b) Image du bus avec la projection d'une mire de fréquence 96 sur celui-ci

FIGURE 4.8 – Images de projection de mires sur un bus

Une méthode simple est déjà implémentée par Blender. Elle utilise les valeurs du Z-buffer, c'est-à-dire la matrice stockant la profondeur en chaque point du pixel de la caméra, pour créer l'image finale. Celle-ci doit être sauvegardée en 16 bits, où 1 pixel est égal à 0,1 mm.
Or, Blender ne gère pas le passage de la profondeur issue du Z-Buffer à une image 16 bits. L'erreur est de 5 mm, ce qui n'est pas compatible avec le cahier des charges (Figure 4.9a). Une autre solution prenant la norme des coordonnées a été testée, mais une déformation non connue et importante impactait le résultat final. L'erreur est supérieure à 30 mm dans certaines parties de l'image, avec une moyenne de 10 mm d'erreur pour cette solution ce qui n'est pas du tout suffisant (Figure 4.9b).

Une autre méthode n'utilisant pas directement le Z-Buffer permet de contourner ce problème. La Figure 4.10 illustre le processus associé.

Pour l'acquisition de profondeur, la première étape de la méthode est de récupérer les points et les triangles des modèles 3D de la scène. Ces points, ayant des coordonnées dans le repère monde, sont modifiés pour avoir des coordonnées 3D selon le repère caméra, puis selon le repère de type image. Cela signifie que les axes X et Z ne sont pas modifiés, mais que l'axe Y est tourné vers le bas. Une conversion est ensuite effectuée. Celle-ci permet de passer d'une profondeur à une image en couleur, où chaque valeur représente une profondeur définie. L'acquisition de l'image de profondeur de la scène peut être réalisée à l'aide des informations de la caméra. Les coordonnées X et Y sont acquises de la même manière. L'erreur de cette méthode est proche de 0,1 mm. Étant



(a) Erreur absolue de l'estimation de la profondeur avec l'acquisition direct du Z-buffer pour un plan situé à 5 mètres

(b) Erreur absolue de l'estimation de la profondeur avec la méthode MIST pour un plan situé à 5 mètres

FIGURE 4.9 – Erreur absolue des différentes méthodes d'acquisition des coordonnées du plan

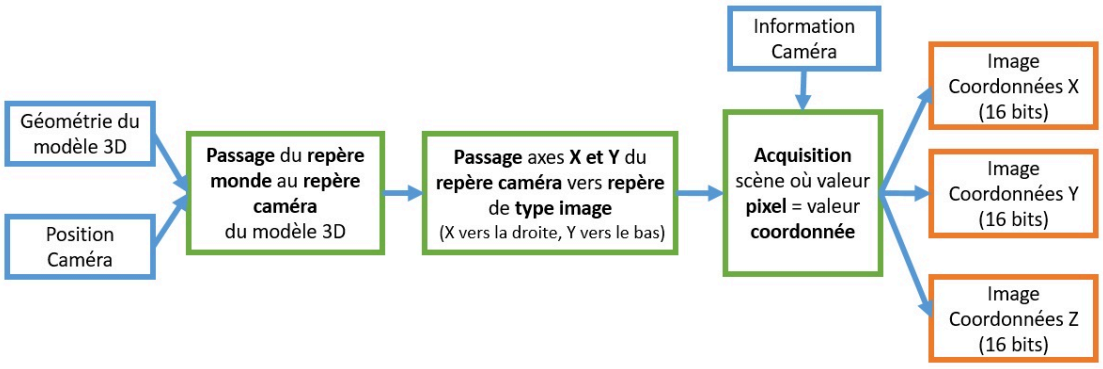


FIGURE 4.10 – Processus permettant d'acquérir l'image de profondeur de la scène. Les carrés en bleu sont les entrées des fonctions. Celles-ci sont représentées par les carrés en vert. Les sorties sont représentées par des carrés orange.

donné que l'erreur maximale souhaitée de la profondeur est de 1,5 mm, cette méthode est validée (Figure 4.11).

4.1.3 Bloc de segmentation multi-échelle haute résolution

Le bloc de segmentation permet de segmenter, sur une image haute résolution, une scène contenant aussi bien des objets de petites dimensions que des objets de grandes

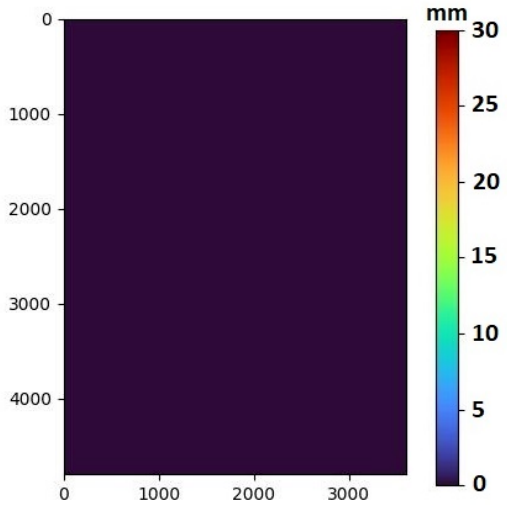


FIGURE 4.11 – Erreur absolue de l'estimation de la profondeur avec la méthode utilisant la texture pour un plan situé à 5 mètres

dimensions. Ce bloc est décrit dans le chapitre 3.1. L'outil de génération d'images de synthèse permet de réaliser une base de données utile pour deux étapes de la segmentation :

— la base de données d'apprentissage pour le CNN de segmentation ;

— des images de test pour analyser l'ensemble de la méthode proposée.

4.1.3.1 Paramètres de test

La base de données d'apprentissage pour le CNN de segmentation a besoin d'une image RGB, d'une image de profondeur et d'une image labélisée pour chaque exemple, de façon à configurer le CNN de segmentation. La base de données doit contenir un nombre important d'exemples avec un nombre important de configurations d'environnements et de textures différents. L'environnement est modifié lorsque le fond change, lorsque le modèle 3D ou la caméra bouge et lorsque les lumières sont plus ou moins intenses. La texture ne doit pas être uniquement composée de couleurs mais aussi de motifs différents pour que le réseau puisse segmenter des objets ayant des textures complexes. Parmi ces textures, on retrouve plusieurs catégories :

— un nombre de carrés aléatoire, de taille aléatoire, avec des couleurs aléatoires, sur un fond de couleur aléatoire ;

— un nombre de carrés aléatoire, de taille aléatoire, avec des couleurs aléatoires, sur un fond de couleur blanc (le fond de couleur blanc a été choisi car c'est une couleur plus courante que les autres) ;

— un nombre de cercles aléatoire, de rayon aléatoire, avec des couleurs aléatoires, sur un fond de couleur aléatoire ;

— un nombre de cercles aléatoire, de rayon aléatoire, avec des couleurs aléatoires, sur un fond de couleur blanc (tout comme pour les cercles, le fond de couleur blanc a été choisi en particulier car c'est une couleur plus courante que les autres) ;

— différents niveaux de gris ;

— des motifs similaires à ceux utilisés par les bus de la STAS à Saint-Etienne ;

— des couleurs unies.

La Figure 4.12 permet de visualiser quelques unes de ces textures.

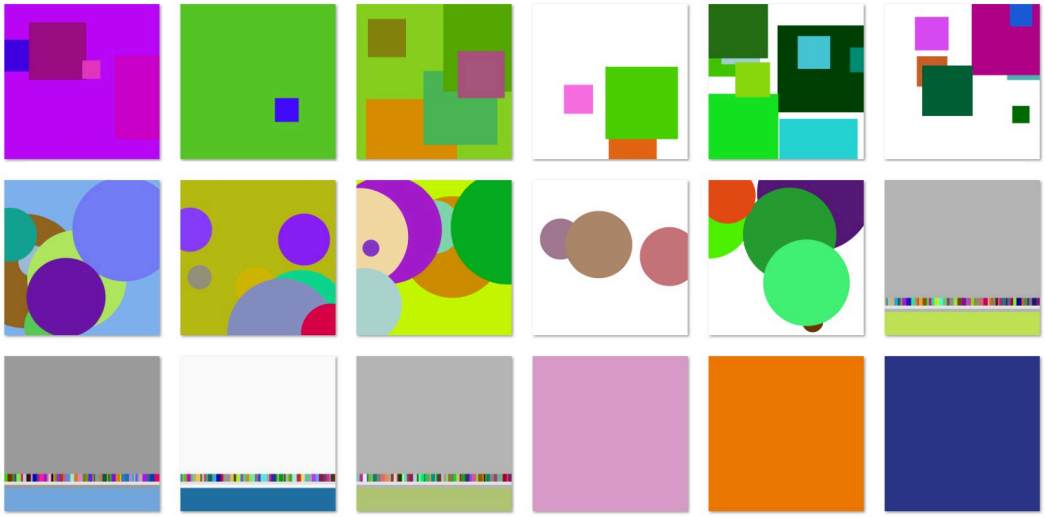


FIGURE 4.12 – Textures créées pour la base de données

Ces paramètres sont modifiés durant la génération de la base de données pour obtenir un nombre important (700) d'exemples pour cette base .

Étant donné que les images préalablement acquises par le bloc ont une résolution de 3600 X 4800 pixels, les images acquises pour ce bloc seront de cette même taille.

4.1.3.2 Blocs fonctionnels

La méthode a besoin d'acquérir un nombre important de scènes composées d'une image RGB, d'une image de profondeur et d'une image labélisée pour pouvoir configurer le CNN. Pour cela, une méthode de génération d'images est mise en place sous Blender (Figure 4.13).

La méthode de génération de la base de données de bus prend plusieurs données en entrée :

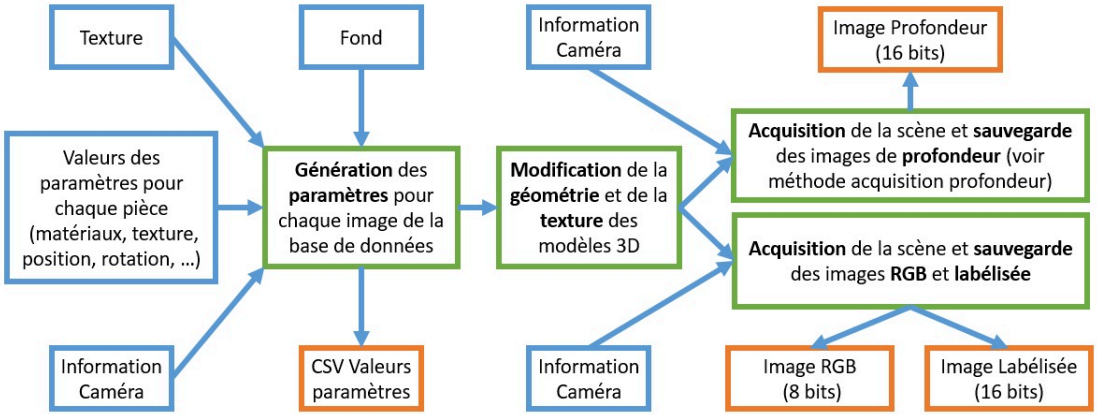


FIGURE 4.13 – Processus permettant d'acquérir une base de données complète contenant des images RGB, de profondeur et labélisées d'une scène. Les carrés en bleu sont les entrées des fonctions. Celles-ci sont représentées par les carrés en vert. Les sorties sont représentées par des carrés orange. Les carrés en pointillés représentent les données déterminées aléatoirement (dans des intervalles préalablement définis).

— l'ensemble des textures générées auparavant et à appliquer sur le bus ;

— les valeurs des paramètres pour chaque pièce, c'est-à-dire le nom de la pièce, son matériau et sa couleur lors de la labélisation, ainsi que la position et la rotation initiales et leurs bornes minimale et maximale.

— le fond ;

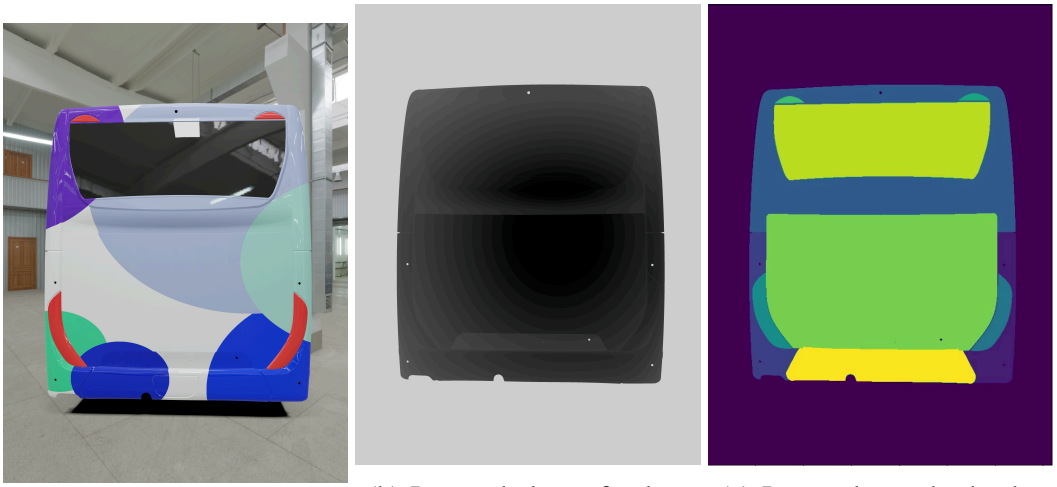
— la position de la caméra.

Ces données sont utilisées pour générer un ensemble de combinaisons de paramètres. Chaque combinaison, unique, permettra de générer un exemple de la base de données. L'ensemble des combinaisons est stocké dans un fichier CSV. En fonction des paramètres présents dans chaque combinaison, la géométrie des modèles 3D, la texture et le fond sont modifiés. Grâce aux informations sur la caméra, il est possible d'acquérir l'image RGB (8 bits) et labélisée (16 bits). L'image de profondeur est acquise grâce à la méthode développée précédemment dans le chapitre 4.1.2.3.

La Figure 4.14 illustre les images générées en sortie de cette méthode.

4.2 Bloc acquisition de la profondeur

Grâce aux images générées par Blender, il est possible d'appliquer la méthode proposée sur une scène comportant un autobus et ainsi, de tester son erreur de mesure sur cette scène. Pour que la méthode puisse être validée, deux véhicules ayant une texture différente sont choisis. Une texture unie et une comportant des disques. Cela permet de



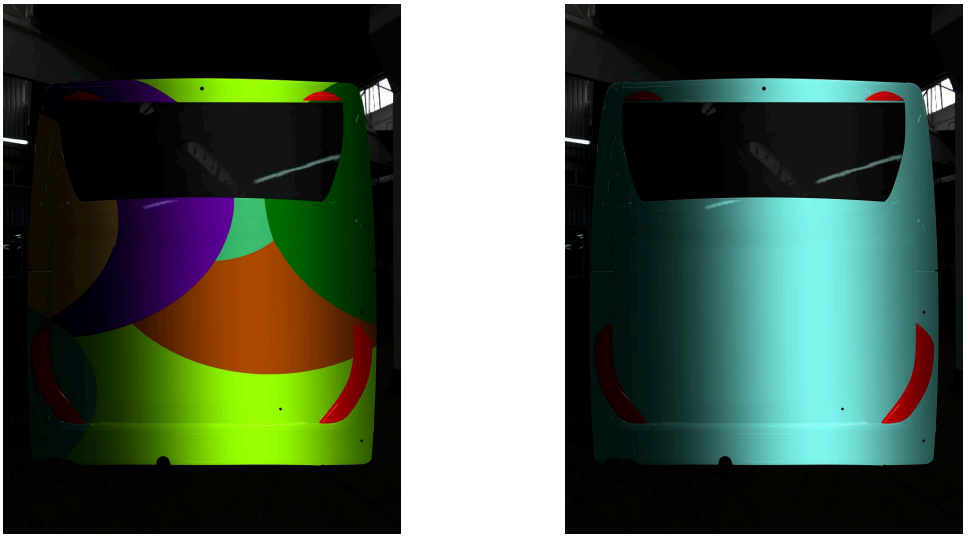
(a) Image du rendu de la scène contenant un bus

(b) Image de la profondeur de la scène contenant un bus

(c) Image du rendu des labels de la scène contenant un bus

FIGURE 4.14 – Images couleur, profondeur et labélisée en sortie de processus

tester la résilience de la méthode au changement de texture.



(a) Bus de test avec un motif uni

(b) Bus de test avec un motif de type STAS

FIGURE 4.15 – Bus de test pour l'estimation de la profondeur

4.2.1 Méthode appliquée sur le bus

Calibrage

La première étape est normalement le calibrage du système. Cependant, il est possible d'accéder aux coordonnées et aux rotations de la caméra et du projecteur directement dans Blender. Les matrices intrinsèques et extrinsèques sont donc connues. Par

conséquent, le calibrage n'est pas nécessaire dans ce cas, et n'est donc pas réalisé.

Estimation et acquisition des plans

La deuxième étape est l'estimation de la profondeur des plans ainsi que leur acquisition. Cinq plans sont acquis pour estimer l'évolution de l'erreur en fonction de la distance. La distance maximale entre l'objet et la caméra est égale à 5m750 et la distance minimale est égale à 4m250. Les plans permettant d'étudier l'évolution de l'erreur sont donc les plans situés à la profondeur 5m750, 5m375, 5m, 4m675 et 4m250. Comme dans la méthode développée dans le chapitre 3, la courbe d'évolution est calculée et permet de connaître l'emplacement des plans à acquérir (Figure 4.16).

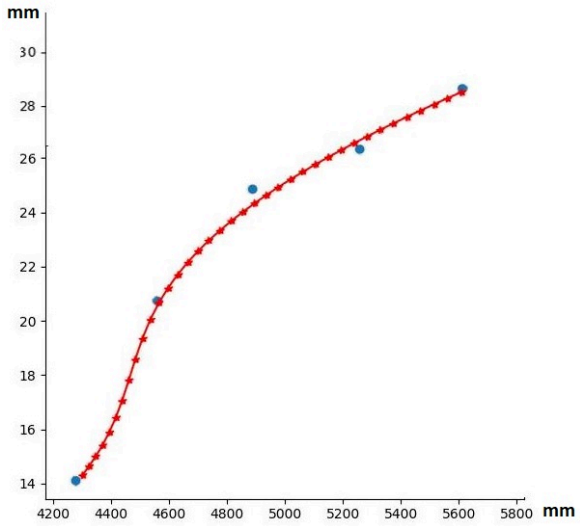


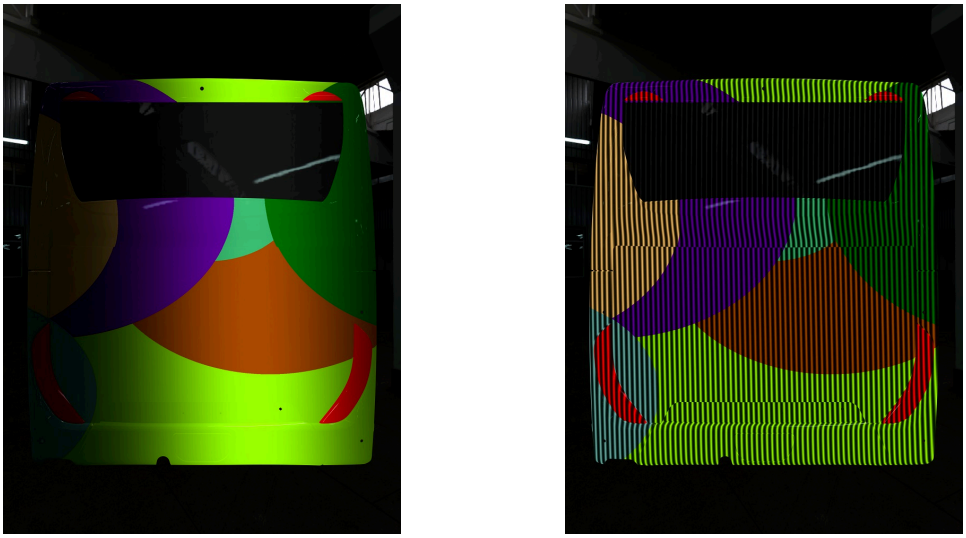
FIGURE 4.16 – Graphique représentant en bleu les distances maximales pour les 5 plans calculés précédemment et en rouge tous les plans nécessaires pour que la méthode puisse estimer la profondeur avec une erreur inférieure à l'erreur maximale souhaitée (ici 1,5 mm). L'axe vertical représente le décalage maximal calculé pour les points bleus pour obtenir au maximum l'erreur souhaitée.

172 plans sont nécessaires pour avoir, en profondeur, une erreur maximale de 1,5 mm par pixel. Grâce à Blender, la méthode Gray Code + Phase shifting n'est pas utilisée pour corriger le positionnement de la caméra car celle-ci est positionnée de façon parfaitement perpendiculaire au plan.

Les plans sont acquis pour des mires de fréquence 1 et 96 avec un déphasage égal à 9 (Figure 4.17).

Calcul de la phase déroulée et de la longueur d'onde des plans

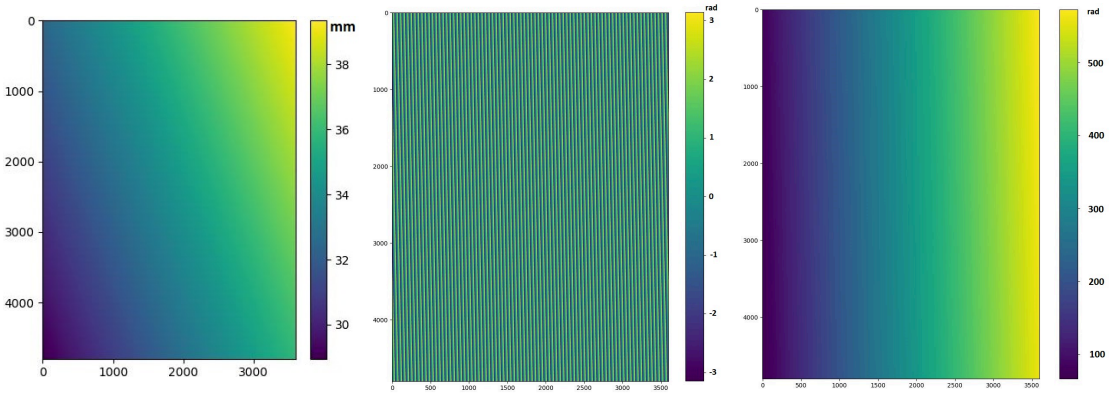
Pour calculer la distance, il est nécessaire de déterminer la phase déroulée ainsi que



(a) Motif de fréquence 1 projetée sur le bus (b) Motif de fréquence 96 projetée sur le bus

FIGURE 4.17 – Motifs de fréquence 1 et 96 projetés sur le bus

la longueur d'onde en chaque point de la caméra, pour tous les plans. Pour accélérer le processus, ces phases et les images de longueur d'onde associées sont sauvegardées pour ne pas recommencer le processus lors des acquisitions suivantes. Tout d'abord, la phase enroulée est calculée grâce aux images de mire de fréquence 96 projetées sur les plans. Ensuite, la phase déroulée est calculée avec les images de mire de fréquence 1 et la phase enroulée calculée précédemment.

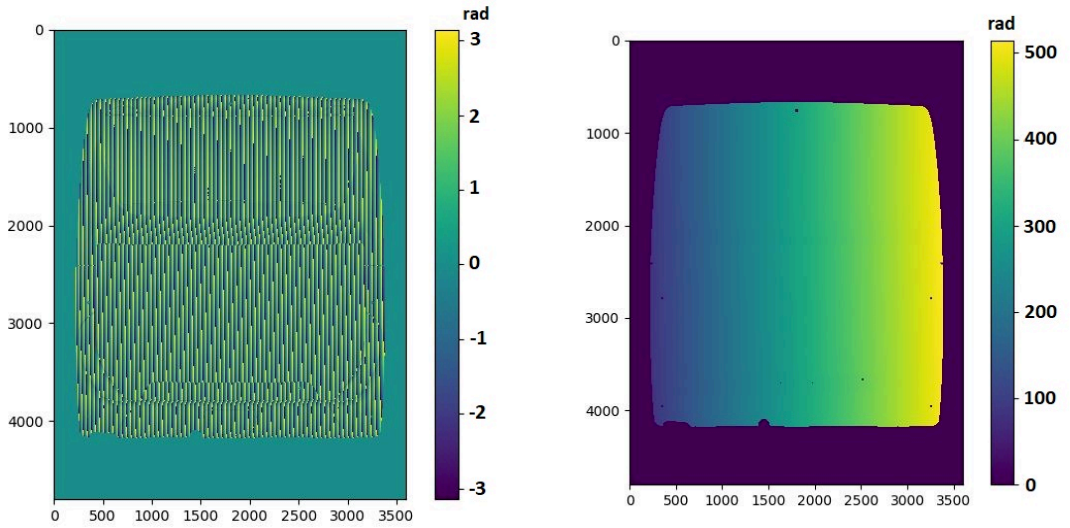


(a) Longueur d'onde du plan (b) Phase enroulée d'un plan (c) Phase déroulée d'un plan

FIGURE 4.18 – Longueur d'onde, phase enroulée et déroulée du plan étant à 5 mètres de la caméra

Calcul de la phase déroulée et de la longueur d'onde du bus

Comme pour les plans, la phase enroulée puis la phase déroulée du bus sont calculées grâce aux images de mire de fréquence 96 projetées sur le bus.



(a) Phase enroulée du bus

(b) Phase déroulée du bus

FIGURE 4.19 – Phase enroulée et déroulée du plan ayant son centre situé à 5 mètres de la caméra

Calcul de la profondeur

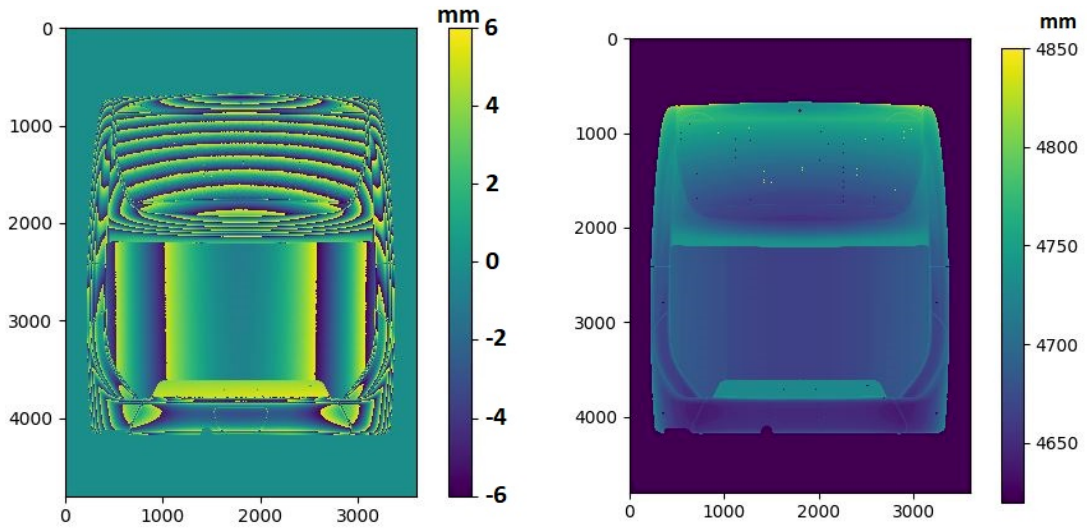
La dernière étape nécessaire pour connaître la distance finale est de réaliser la différence de phase entre les plans et le bus. Pour chaque plan, la profondeur par rapport au plan est calculée. Cette profondeur relative est comparée en chaque pixel par rapport aux autres plans. La profondeur relative retenue pour chaque pixel de l'image finale est la profondeur relative minimale entre tous les plans.

Pour connaître la profondeur finale, il suffit d'ajouter, pour chaque pixel, la distance caméra-plan de référence choisi à la profondeur relative calculée précédemment .

4.2.2 Analyse

Les deux bus de test permettent d'estimer la fiabilité de la méthode. La Figure 4.21a permet de visualiser les erreurs de la méthode, notamment où celles-ci sont situées, et leurs amplitudes. Les résultats présentés sont identiques pour les deux types de bus. Il est possible de remarquer, dans un premier temps, que l'erreur trouvée dans l'image est comprise entre -1,5 mm et 1,5 mm se qui confirme que la méthode permet d'obtenir une erreur de mesure fixe. La moyenne des erreurs est de 0,028 mm et de variance égale à 0,0054 mm.

Cependant, quelques points (représentés dans la Figure 4.21b et principalement

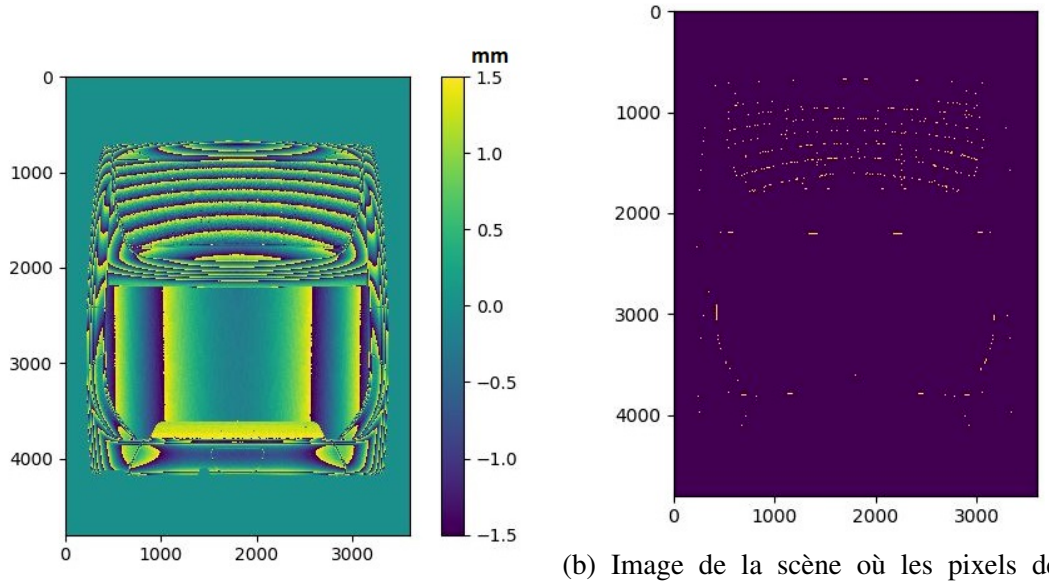


(a) Profondeur relative aux plans de référence (b) Profondeur estimée du bus dans la scène

FIGURE 4.20 – Profondeur relative aux plans et profondeur absolue (référence monde positionnée sur la caméra)

situés dans la zone de la vitre) ont une erreur de mesure supérieur à l'erreur de mesure maximale souhaitée.

Cela peut venir du matériau qui réfléchit moins bien les motifs projetés et de la méthode de projection utilisée (raycasting).



(a) Erreur de profondeur du bus dans la scène

(b) Image de la scène où les pixels de la scène au dessus de l'erreur de mesure souhaitée (1,5mm) sont représentés en jaune

FIGURE 4.21 – Profondeur relative aux plans et profondeur absolue (référence monde positionnée sur la caméra)

Cette image de profondeur peut donc être utilisée pour déterminer l'écart entre les pièces de carrosserie du bus car elle donne la distance objet-caméra avec une erreur inférieure à celle demandée. Étant donné que l'image contient une erreur faible, elle peut être utilisée pour le bloc permettant d'identifier et de segmenter les pièces de carrosserie.

4.3 Bloc segmentation haute résolution et multi- échelle

La méthode globale, présentée dans le chapitre 1, et plus en détails pour la méthode CNN-CA, dans le chapitre 3, est testée grâce aux images de profondeur acquises dans le bloc précédent et à l'outil de simulation Blender.

4.3.1 Méthode appliquée sur le bus

4.3.1.1 Soustraction du fond de la scène

La première étape est un post-traitement, en l'occurrence, le fait d'enlever le fond de la scène. Cette étape est facilement réalisable car le fond est très éloigné des pièces du bus à traiter. Cela permet par la suite que les segmentation des pièces ne puissent pas déborder dans le fond de la scène et ainsi générer des erreurs.

(a) Image initiale de la scène

(b) Image de la scène sans fond

(c) Image de la profondeur permettant d'enlever le fond en post-traitement

FIGURE 4.22 – Étape de pré-traitement : enlever le fond de la scène grâce à l'image de profondeur

L'image est par la suite traitée par deux branches distinctes qui reprennent le même processus, c'est-à-dire segmenter l'image grâce à un CNN et améliorer la segmentation

grâce aux contours actifs.

4.3.1.2 CNN

Avant de pouvoir utiliser un CNN pour segmenter l'image du bus et ses pièces, il est nécessaire de le choisir et surtout de le paramétrer. Pour cela, une base de données d'entraînement contenant des images RGB-D et labélisées représentant des exemples de bus dans la scène est nécessaire.

Le réseau U-Net[3] (Figure 4.23) est choisi pour réaliser la segmentation de la scène. En entrée, l'image RGB est concaténée à l'image de profondeur. Ainsi, le réseau prend en entrée une image quatre canaux.

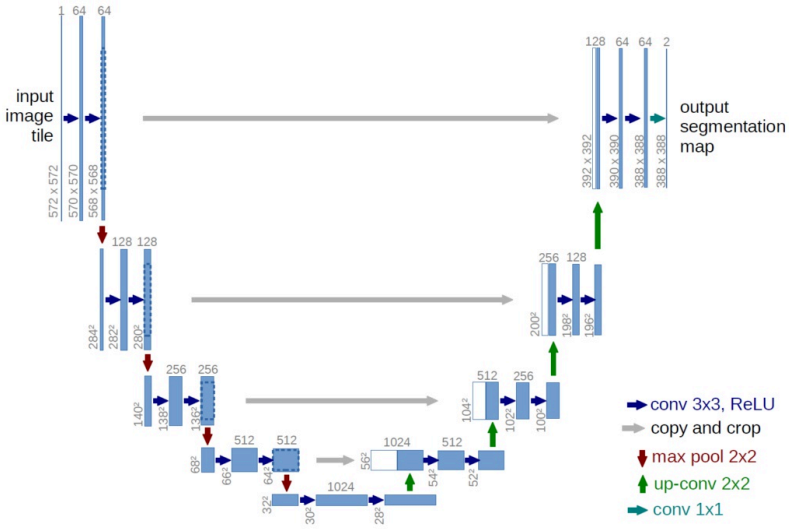


FIGURE 4.23 – Architecture U-Net [3]

La base de données utilisée pour entraîner le réseau est générée grâce à Blender. Elle contient 500 images RGB-D et labélisées de dimension 1920x1440 pixels. Toutefois, en raison de la limitation de mémoire sur le GPU (16Go), la résolution des images en entrée du réseau a dû être divisée par deux. Les images sont donc de 960x720 pixels. Avec cette résolution, la taille de batch maximal, c'est-à-dire le nombre maximal d'images pour chaque itération du paramétrage du CNN, est limitée à trois.

Le réseau U-net a été choisi par rapport au réseau GAD détaillé dans le chapitre 3.4.1 pour de multiples raisons. La première raison est que U-net est simple d'implémentation et peu gourmand en terme de mémoire, par rapport à GAD. En effet, étant donné que le cas d'application nécessitait de prendre en entrée de CNN une image de taille "moyenne voir grande", c'est à dire au minimum 500x500 pixels, voir au dessus de

700x700 pixels, GAD ne permettait pas de remplir cette condition. De plus, après avoir effectué l'entraînement et le test du réseau U-Net sur la base de données de bus, l'IoU final était de 97 %, cela suffit amplement pour avoir un résultat correct, c'est-à-dire proche du contour des objets.

Objets de grande taille

Pour les objets de grande taille, tous les objets sauf les trous sont recherchés. On utilise l'image complète en entrée du CNN.

Objets de petite taille

Les objets de petite taille, c'est-à-dire les trous, sont recherchés dans la deuxième branche. Deux façons de segmenter sont possibles. La première consiste à segmenter l'image en trois labels qui sont le fond, les grandes pièces et les trous. La deuxième façon est de segmenter l'image en deux labels, c'est à dire les trous d'une part, et les autres pièces et le fond d'autre part.

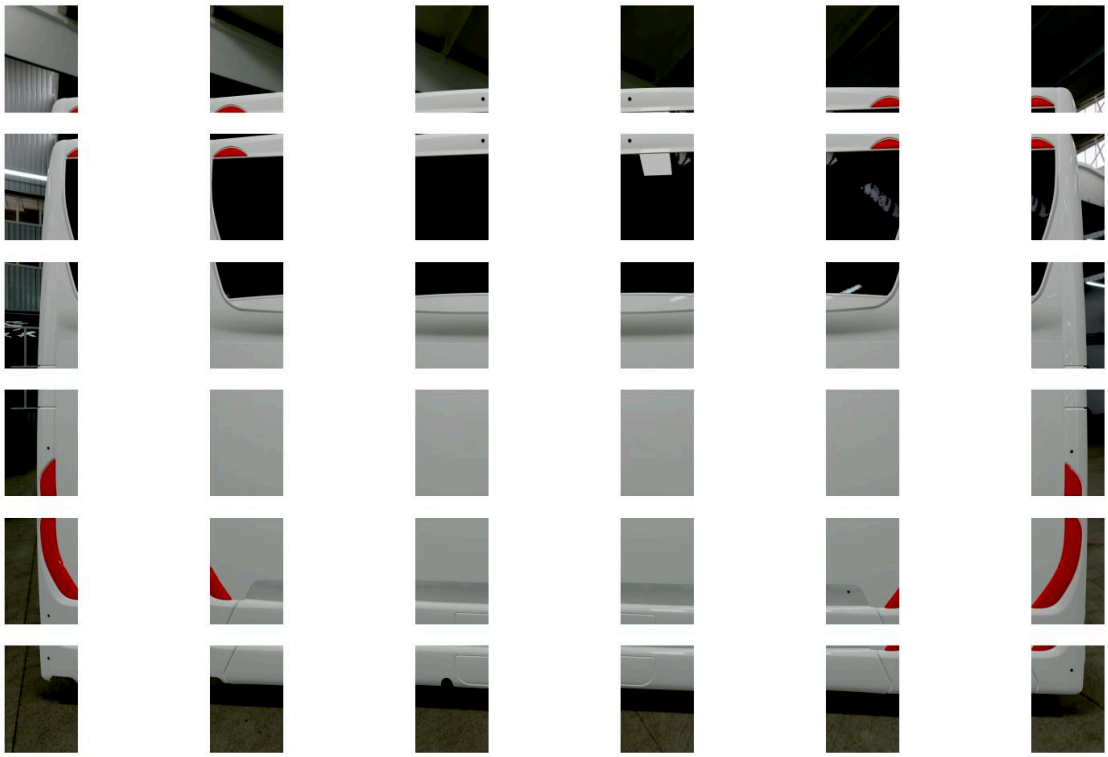
Après plusieurs tests d'apprentissage avec le CNN, il s'avère que la méthode permettant d'obtenir une segmentation correcte est de rechercher trois types d'objets dans l'image : le fond, les grandes pièces et les trous.

Pour percevoir les petits objets dans l'image en entrée de CNN, l'image est découpée en tuiles avec un recouvrement de 80 pixels. Cela permet qu'un objet de 5 cm maximum soit perçu en entier dans au moins une image. Cela permet de ne pas segmenter des bouts de trou mais bien des trous complets (Figure 4.24).

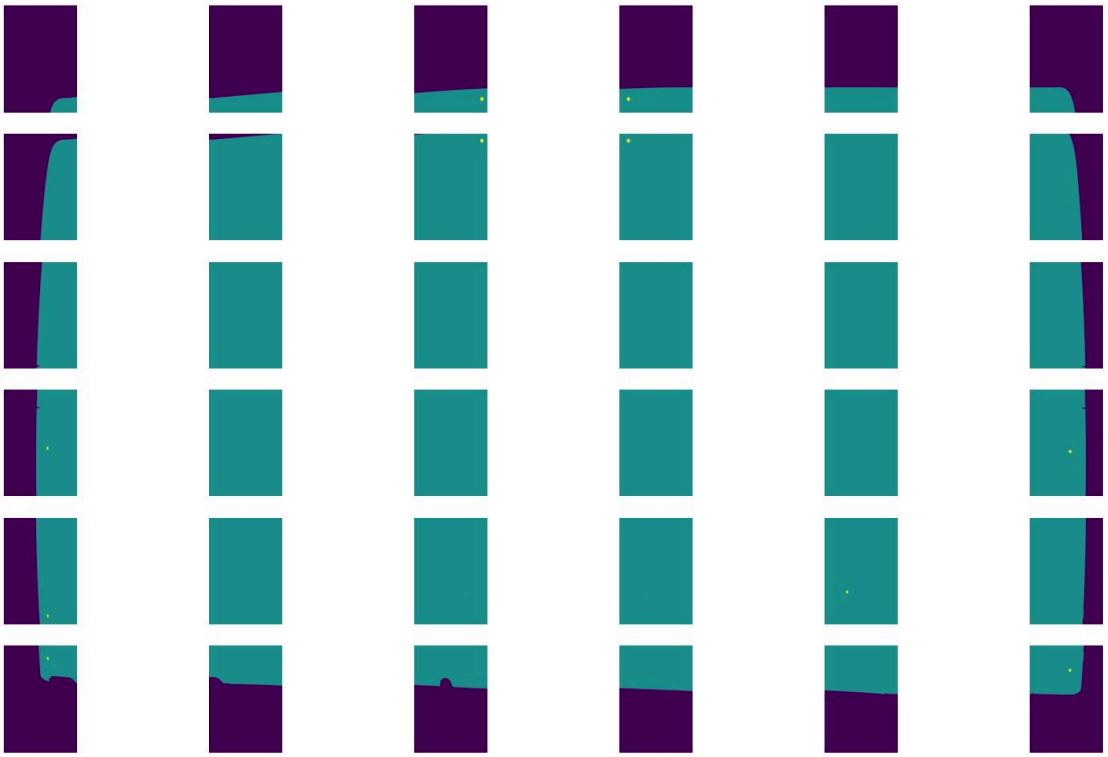
Après 5000 itérations, l'IoU moyen des différents objets obtenu à la fin de l'entraînement se situe autour des 97%. Un exemple de résultat de segmentation est visible en Figure 4.26 .

L'IoU souhaité doit être égal à 100% (Figure 4.26-d). Cependant, le CNN ne délivre pas une segmentation sans erreur (97%) (Figure 4.26-b). Cela implique que des erreurs apparaissent sur le contour des objets, qu'ils soient gros ou petits (Figure 4.26-c). Cela est dû au paramétrage du CNN lors de son entraînement.

Avant que cette image ne soit traitée par le contour actif, elle est remise à la résolution maximale. Pour cela, une fonction de redimensionnement avec une interpolation au plus proche est utilisée. Ce type d'interpolation est utilisé pour éviter le flou aux bords des objets et ainsi garder une image labélisée avec des valeurs de labels réelles. Ce redimensionnement affecte la qualité de segmentation finale. En effet, l'erreur est plus

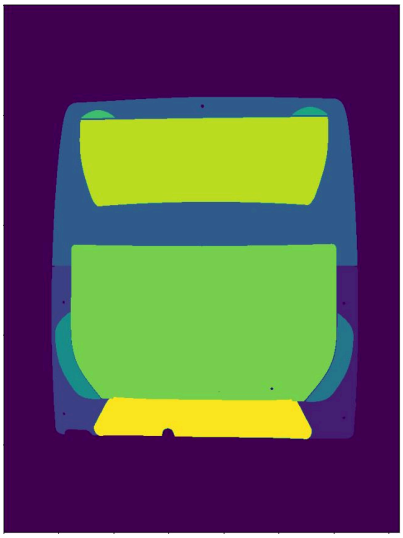


(a) Tuiles de l'image de couleur

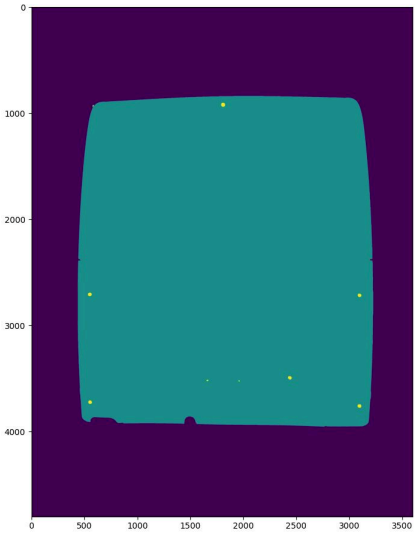


(b) Tuiles de l'image labélisée

FIGURE 4.24 – Découpage de l'image RGB et labélisée en tuiles



(a) Image de sortie du CNN pour la branche des grandes pièces



(b) Image de sortie du CNN pour la branche des petites pièces

FIGURE 4.25 – Sorties de CNN en fonction des types d'objets traités. A gauche, traitement des objets de grande taille et à droite, les trous.

importante autour des objets comme vu dans la Figure 4.27.

4.3.1.3 Amélioration de la segmentation

Initialisation

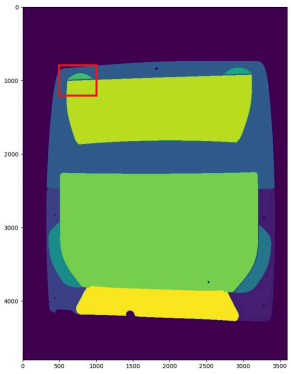
La première étape consiste à faire une ouverture, c'est-à-dire une érosion suivie d'une dilatation de la segmentation initiale. Cela permet d'éliminer les pixels mal prédits. Étant donné que le CNN donne des résultats avec un IoU proche de 100 %, très peu de pixels sont modifiés.

Ce n'est qu'après cette étape que les objets sont traités un par un. Pour cela, les images sont centrées sur ces objets (Figure 4.28).

Calcul des forces et exploration de leurs coefficients

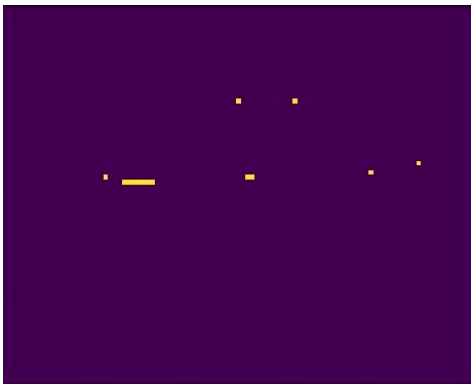
C'est à partir de ces images que les forces de gradient et de GVF sont calculées. La Figure 4.29 représente les forces GVF et les gradients de l'image centrée sur le feu arrière droit.

Grâce à ces forces, les coefficients sont calculés pour chaque objet , où la segmentation doit être améliorée. C'est grâce au contour actif et notamment à l'initialisation de la courbe issue du label de sortie du CNN, des forces et de leurs paramètres qu'il est

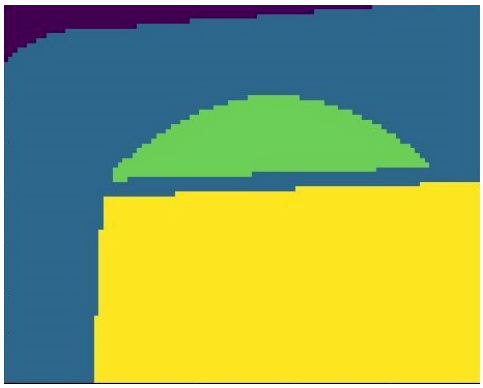


(a) Image de la scène où une zone de segmentation est prise en exemple pour illustrer l'erreur de segmentation

(b) Image segmentée en sortie du CNN



(c) Erreur (pixel en jaune) entre la sortie du CNN et la vérité terrain



(d) Image labélisée (vérité terrain) en basse résolution

FIGURE 4.26 – Sortie du CNN avec erreur par rapport à la vérité terrain

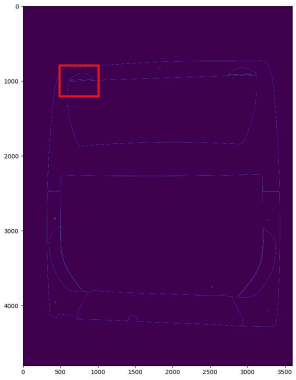
possible d'améliorer la segmentation.

Contours actifs

En calculant la carte de déplacement, la courbe initiale se déplace vers le bord de l'image. La segmentation finale semble plus proche des contours des pièces de carrosserie que la segmentation en sortie de CNN, que ce soit pour le bus uni (Figure 4.30), ou pour le bus de couleur et non uni (Figure 4.31).

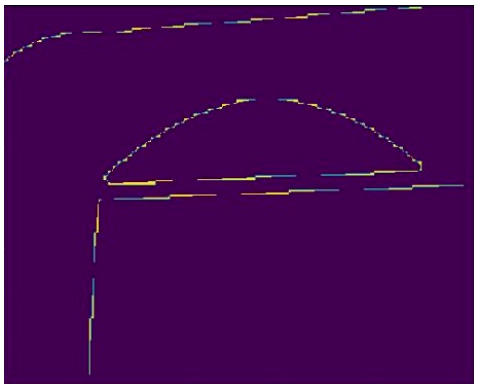
4.3.2 Analyse

Pour que l'analyse soit complète, la méthode a été réalisée sur deux bus. Le premier bus a une texture unie alors que le deuxième bus à une texture avec des formes. Les deux bus ne sont pas situés sur le même plan (Figure 4.32).

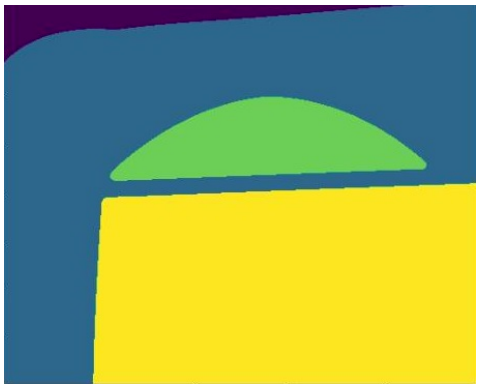


(a) Image des erreurs de la segmentation dues au redimensionnement de l'image pour une haute résolution

(b) Image segmentée en sortie du CNN et redimensionnée



(c) Erreur (pixels en jaune) entre la sortie du CNN et la vérité terrain



(d) Image labélisée (vérité terrain) en haute résolution

FIGURE 4.27 – Erreur due au redimensionnement de l'image segmentée

Le Tableau 4.1 permet de visualiser le gain de la segmentation grâce à la méthode CNN-CA.

La méthode améliore la segmentation de la majorité des pièces. Le gain en IoU est faible car sa valeur initiale était déjà très proche de 100%. L'IoU n'est pas amélioré pour le feu arrière droit, le hayon et le montant inférieur. Pour ces trois pièces, un objet ou un bout de la texture du bus attire le contour actif vers un emplacement qui n'est pas le bord de l'objet à segmenter. Cela est visible sur la Figure 4.33 où une ombre (Figure 4.33-a) ou un objet connexe (Figure 4.33-b) attire le contour.

Des résultats similaires sont obtenus sur le bus ayant une texture non unie.

La méthode permet donc d'améliorer la segmentation en sortie d'un CNN sur la majorité des pièces. Cependant, le gain est maximal lorsqu'un des contours de l'objet est très présent, que ce soit en termes de profondeur, de couleur (RGB) ou les deux.

(a) Image de couleur centrée sur le feu arrière droit

(b) Image de couleur centrée sur le hayon

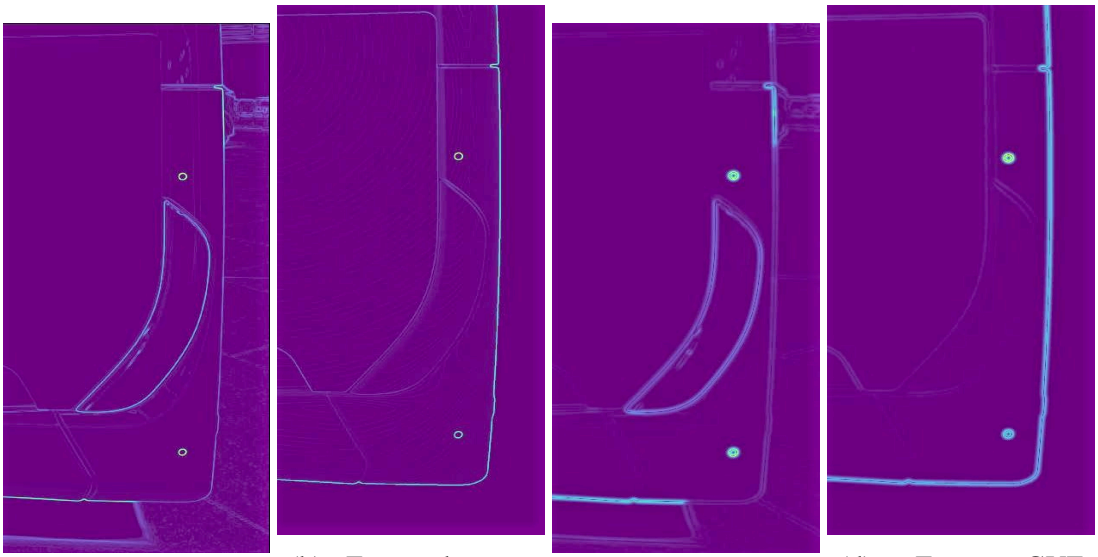
(c) Image de profondeur centrée sur le feu arrière droit

(d) Image de profondeur centrée sur le hayon

(e) Image labélisée centrée sur le feu arrière droit

(f) Image labélisée centrée sur le hayon

FIGURE 4.28 – Images centrées sur deux pièces du bus



(a) Force de gradient (image RGB)

(b) Force de gradient (image de profondeur)

(c) Force GVF (image RGB)

(d) Force GVF (image de profondeur)

FIGURE 4.29 – Forces GVF et gradients de l'image centrée sur le feu arrière droit.

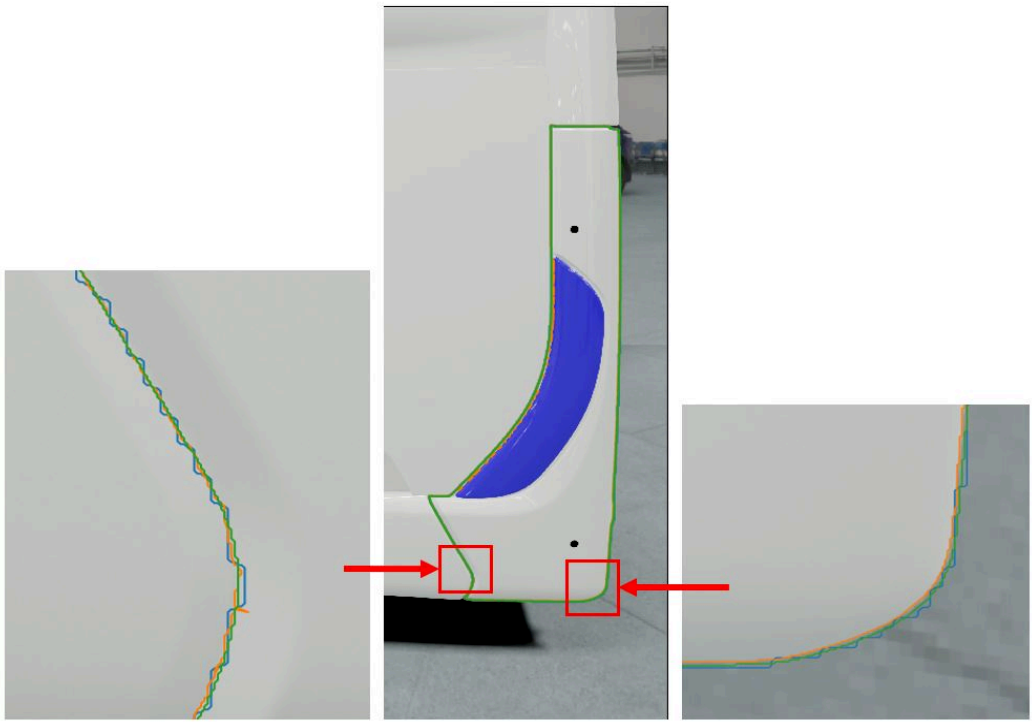


FIGURE 4.30 – Résultat de la méthode CNN-CA sur le bus de texture unie. La courbe orange est le résultat de la méthode proposée, la courbe bleue l'initialisation et la courbe verte, la vérité terrain

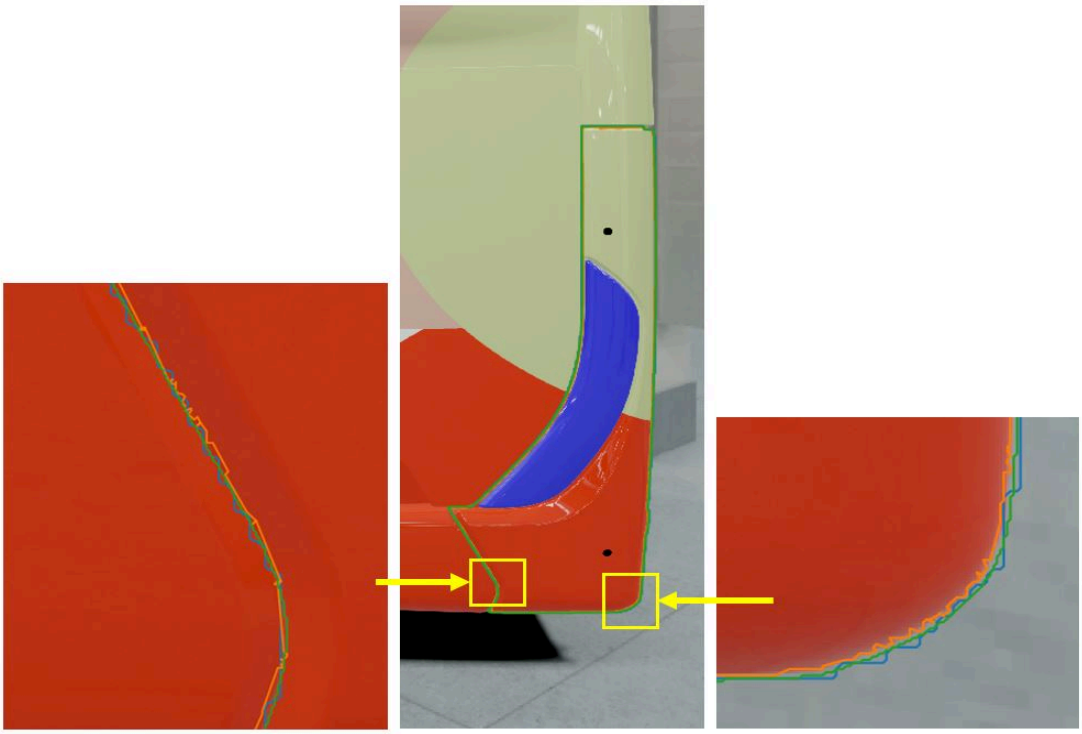


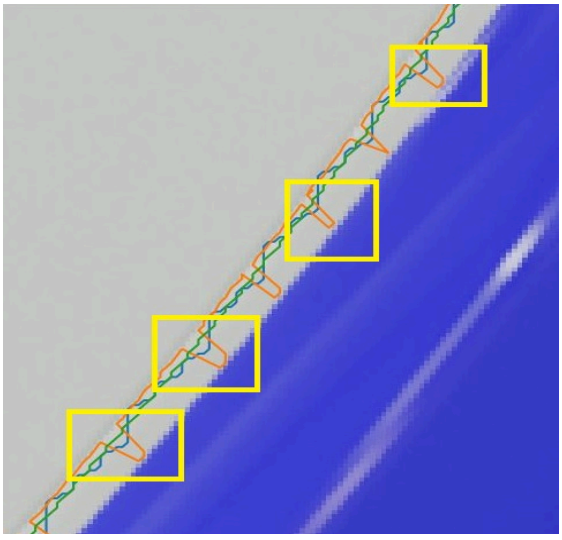
FIGURE 4.31 – Résultat de la méthode CNN-CA sur le bus de texture non unie. La courbe orange est le résultat de la méthode, la courbe bleue l'initialisation et la courbe verte, la vérité terrain.

(a) Premier bus de test (b) Deuxième bus de test

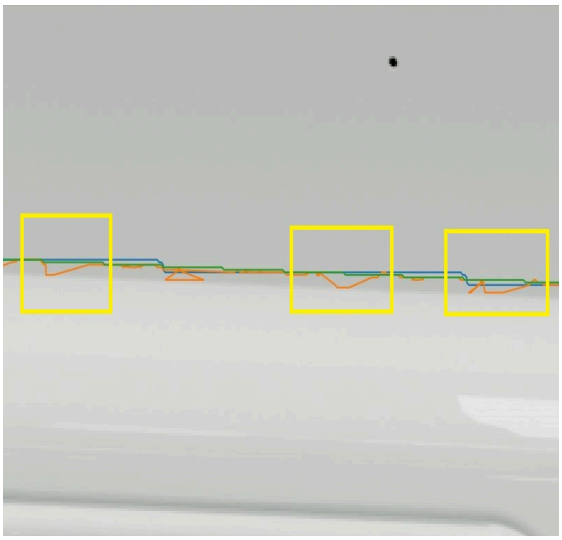
FIGURE 4.32 – Deux exemples utilisés pour l'analyse de la méthode

TABLE 4.1 – Image de couleur centrée sur le feu arrière droit

Classe	IoU CNN	IoU CNN ouverture	IoU CNN + CA	différence entre IoU CNN - CNN + CA
Montant arrière droit	9,82E-01	9,83E-01	9,79E-01	-3,52E-03
Montant arrière gauche	9,74E-01	9,75E-01	9,85E-01	1,08E-02
Montant supérieur	9,91E-01	9,91E-01	9,94E-01	3,42E-03
Feu latéral droit	9,83E-01	9,84E-01	9,89E-01	5,89E-03
Feu latéral gauche	9,82E-01	9,78E-01	9,85E-01	3,31E-03
Feu supérieur droit	9,45E-01	9,47E-01	9,58E-01	1,38E-02
Feu supérieur gauche	9,51E-01	9,29E-01	9,61E-01	1,04E-02
Hayon	9,98E-01	9,98E-01	9,97E-01	-2,31E-04
Vitre	9,96E-01	9,96E-01	9,98E-01	1,41E-03
Montant inférieur	9,92E-01	9,92E-01	9,88E-01	-4,01E-03
Trous	9,45E-01	9,45E-01	9,78E-01	3,30E-02



(a) Erreurs dues à un objet ayant une texture contrastée



(b) Erreurs dues à l'ombre d'un objet. La courbe orange est le résultat de la méthode, la courbe bleue l'initialisation et la courbe verte, la vérité terrain.

FIGURE 4.33 – Cas rencontrés où la méthode CNN-CA commet des erreurs de segmentation. La courbe orange est le résultat de la méthode, la courbe bleue l'initialisation et la courbe verte, la vérité terrain.

BIBLIOGRAPHIE

[1] Matthieu LEGOUGE. *Ray tracing : qu'est-ce que c'est et comment ça améliore vos jeux vidéo*. fr-FR. Jan. 2020. URL : https://www.frandroid.com/dossiers/660693_dossier-ray-tracing-explication (visité le 09/05/2022).

[2] Predrag NOVAKOVIĆ et al. *3D Digital Recording of Archaeological, Architectural and Artistic Heritage*. Déc. 2017. ISBN : ISBN 978-961-237-898-1 (pdf). DOI : 10.4312/9789612378981.

[3] Olaf RONNEBERGER, Philipp FISCHER et Thomas BROX. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In : *CoRR* abs/1505.04597 (2015). arXiv : 1505.04597. URL : http://arxiv.org/abs/1505.04597.

CHAPITRE 5

CONCLUSION GÉNÉRALE ET PERSPECTIVES

Ce chapitre présente dans un premier temps les résultats des travaux et les contributions associées, et dans un deuxième temps les perspectives et pistes d'améliorations des méthodes présentées, les poursuites de recherches possibles ainsi que les domaines dans lesquels ces travaux pourraient s'appliquer.

5.1 Conclusion

Dans l'industrie, le contrôle qualité est omniprésent pour vérifier un nombre de caractéristiques toujours plus important. Cela a notamment pour impact la complexification des systèmes de vision utilisés pour les contrôles de pièces. En effet, ces dispositifs sont composés de matériels et d'algorithmes sophistiqués pour répondre aux exigences toujours accrues.

Le contrôle du montage de véhicules industriels et plus particulièrement de type autobus et autocars, notre cas d'étude, n'en est pas exempt. Au contraire, il nécessite de contrôler finement le positionnement des pièces de carrosserie de l'arrière d'un bus, notamment pour répondre aux exigences de la "qualité perçue" et de l'esthétisme du véhicule. Les autobus et autocars sont composés aussi bien de grandes pièces (par exemple le hayon) que de petites pièces (par exemple les feux arrières). La surface d'analyse est très grande (5m x 4m x 1m50). Pour s'assurer d'atteindre l'erreur de 1 mm spécifiée dans le cahier des charges (chapitre 1.2), une erreur encore inférieure (i.e. en dessous du millimètre) est recherchée. Le contrôle du montage doit être réalisé sur des images de très haute résolution pour visualiser la scène en entier tout en ayant une erreur de mesure faible souhaitée.

La solution proposée est séparée en trois blocs. Le premier permet d'acquérir avec une faible erreur la profondeur de la scène, le deuxième permet de segmenter en multi-échelle les pièces de carrosserie grâce à des images de couleur et de profondeur. Enfin, le troisième bloc, non exploré durant cette thèse, permet d'estimer l'écart entre les pièces de carrosserie. L'acquisition et l'utilisation des images RGB-D dans la solution proposée, c'est-à-dire associant une image couleur et une image de profondeur, permettent dans un premier temps d'améliorer la segmentation finale et, dans un deuxième temps, de mesurer les jeux entre les pièces de carrosserie.

Concernant le bloc acquisition de la profondeur, aucune méthode de la littérature ne permet d'obtenir l'erreur souhaitée pour un volume et une distance d'acquisition aussi importants. La méthode rassemblant le plus de qualités pour acquérir la profondeur est la lumière structurée. La méthode proposée associe ainsi plusieurs algorithmes de lumière structurée, en l'occurrence, une méthode de décalage de phase (Bi-frequency) et une de triangulation (Gray Code + Phase Shifting), pour obtenir une image de profondeur où les valeurs maximales ne dépassent pas l'erreur de mesure souhaitée. Cette méthode est donc très précise car l'erreur issue des algorithmes est limitée à un seuil défini au préalable. Lorsque ce seuil est atteint, un nouveau plan de référence est acquis, permettant de réinitialiser l'erreur sur ce nouveau plan de référence et ainsi de diminuer l'erreur de la mesure. Pour une erreur recherchée de 2 mm en profondeur en chaque pixel, tous les pixels estimés sont en dessous de cette valeur et l'erreur moyenne est en dessous à 0,2 mm. Par rapport à l'approche du décalage de phase utilisée seule, la moyenne des erreurs obtenue avec la méthode proposée est divisée par 12000, passant de 5mm à 0,005 mm d'erreur pour un plan proche du plan de référence, ce qui représente un gain significatif.

Dans le bloc de segmentation multi-échelle haute résolution des pièces de carrosserie, une solution permettant de segmenter les différents types de pièces est proposée. Elle sépare le traitement des objets en fonction de leur taille (grand, moyen et petit). Les objets d'une même taille sont traités dans une branche spécifique. Dans chaque branche, le processus combine des algorithmes de Deep Learning (CNN de segmentation) et de traitements d'image (contour actif) pour segmenter finement une image haute résolution. Les CNN permettent de segmenter des images en prenant en compte la complexité de la scène (environnement non maitrisé), la profondeur de celle-ci, le placement et la texture des objets. Les CNN utilisent énormément de ressources matérielles, ce qui constitue une limite. Ainsi, la segmentation des images ne peut être réalisée que sur des images de résolution plus faible que celle nécessaire pour obtenir l'erreur voulue pour l'application (1 mm). Les contours actifs permettent donc de prendre la segmentation basse

résolution issue du CNN pour pouvoir améliorer celle-ci en haute résolution. Le gain d'IoU de cette méthode est significatif car elle permet à 70% des objets testés d'avoir une segmentation finale améliorée par rapport à la sortie du CNN. Le gain d'IoU sur la base NYUv2 pour les objets améliorés est en moyenne de 9.5% d'IoU.

Une expérimentation visant à valider la solution proposée est réalisée sur des bus. Pour de multiples raisons, notamment le fait que le système d'acquisition permettant d'obtenir la profondeur n'est pas construit, la base de données nécessaire à la validation de la solution est générée en utilisant un logiciel de visualisation 3D. La solution est ensuite appliquée aux images générées. La solution d'estimation de la profondeur de la scène permet de respecter l'erreur voulue (dans ce cas-là, l'erreur a été définie à 1,5mm). Cette image de profondeur est ensuite utilisée dans la méthode de segmentation fine. La segmentation finale des pièces est proche de 98% d'IoU pour 82% des pièces du bus, et permet d'améliorer la segmentation issue du CNN dans 82% des cas (type de pièces).

Les travaux réalisés durant cette thèse proposent donc une solution globale permettant de segmenter un bus complet en haute résolution avec une faible erreur.

5.2 Perspectives

5.2.1 Bloc acquisition de la profondeur

La méthode d'acquisition de la profondeur a été testée et validée grâce à de la simulation. Cependant, plusieurs paramètres peuvent altérer les résultats obtenus.

Le premier paramètre est la déformation de la lentille du projecteur et de la caméra qui peut impacter l'estimation de la rotation et de la profondeur du plan.

Le deuxième paramètre est la puissance du projecteur. La projection des mires a été considérée comme parfaite au niveau éclairage, c'est-à-dire que la puissance du projecteur a été considérée comme suffisante. Si cela n'est pas le cas, une perte d'informations et une erreur au niveau de la phase peuvent apparaître. Des algorithmes permettant de corriger cette erreur existent et restent à implémenter pour rendre la solution proposée plus robuste.

Le troisième paramètre est la prise en compte de la réflexion spéculaire (i.e. les reflets) du projecteur. Lors de la projection des mires, l'image résultante ne prend en compte que les réflexions spéculaires dues à la scène mais non celles liées au projecteur. Cette réflexion spéculaire peut saturer la luminosité de certains pixels de l'image acquise

par la caméra, cela va altérer les résultats à l'endroit de la réflexion. En effet, si les valeurs des pixels ne changent pas, il est impossible de calculer la phase des motifs projetés. Une solution envisagée est de mettre en place une deuxième caméra ou un deuxième projecteur pour avoir un deuxième point de vue de l'émetteur ou du projecteur et ainsi acquérir les zones rendues non exploitables par la réflexion.

Le quatrième et dernier paramètre est le fait que cette méthode a été testée pour une distance d'acquisition proche de 5 mètres. Des tests sur des distances différentes (de 1 à 20 mètres) permettraient de valider la qualité et la faisabilité de la méthode.

Tout cela implique de réaliser des tests en réel, c'est-à-dire sur des bancs de tests avec des caméras, lentilles, projecteurs et objets physiques et connus, pour valider la méthode élaborée et présentée dans ces travaux.

Cette méthode peut être utilisée dans tous les cas d'applications de contrôle de produits et nécessitant une erreur très faible. Les dimensions des objets peuvent être petites ou grandes. Les paramètres limitants étant respectivement la puissance du projecteur et la mesure pouvant être acquise avec les outils de métrologie.

5.2.2 Bloc segmentation fine et multi-échelle

La méthode combinant un CNN avec un contour actif permet d'obtenir une meilleure segmentation que celle délivrée par le CNN seul.

La première amélioration possible est liée à la façon dont l'image de segmentation finale est obtenue. La méthode de segmentation traite les objets un à un, ce qui peut engendrer des conflits au niveau de la segmentation finale. En effet, un pixel peut obtenir deux labels de deux objets différents, ce qui est problématique. Une méthode de fusion est à approfondir pour que ce type d'erreur soit géré.

Le deuxième point d'amélioration concerne la segmentation. La méthode hybride est utilisée en grande partie car les CNN utilisent énormément de ressources, ce qui limite la taille des images. L'utilisation et optimisation de l'apprentissage des CNN sur un ou plusieurs GPU permettraient d'augmenter considérablement la taille des images en entrée du CNN et ainsi segmenter des objets ayant des différences de taille plus importantes. Cela réduirait le temps de traitement.

Enfin, cette méthode est particulièrement adaptée pour les objets ayant leurs contours bien définis, comme pour les pièces du bus. Cependant, elle présente des limites en termes de robustesse et de rapidité de calcul, et n'est pas performante dans toutes

les applications. D'autres méthodes hybrides seraient peut être plus adaptées dans des contextes moins simples comme la segmentation d'objets dans une scène. Une approche région peut ainsi être envisagée.

5.2.3 Bloc estimation des jeux entre les objets

Bien que le bloc d'estimation des jeux des objets ne soit pas encore disponible, le processus global a quand à lui déjà été élaboré. Ainsi, ce bloc doit prendre en entrée la segmentation améliorée, l'image de profondeur ainsi que les règles fixées par le constructeur ou apprises sur les modèles 3D, pour pouvoir ensuite calculer la distance entre deux pièces de carrosserie, de bord à bord. La recherche du point du bord de l'objet connexe est une étape importante dans ce bloc. Les règles utilisées pour contrôler le placement des pièces peuvent être une distance maximale à ne pas dépasser ou une symétrie axiale à vérifier et respecter. De même, des tolérances au niveau de la déformation des pièces peuvent être ajoutées pour que le système soit robuste.

Ainsi, bien que ce dernier bloc ne soit pas opérationnel, l'ensemble du processus a pu être décrit. De plus, les deux premiers blocs sont complets et fonctionnels. Ils peuvent être mis en oeuvre conjointement ou séparément, sur des applications telles que les véhicules industriels comme proposé dans ce manuscrit, ou pour d'autres usages.

Ainsi, le premier domaine concerné est celui du contrôle qualité de produits avec une erreur très faible et où un temps de traitement relativement long, de l'ordre de 5 secondes, est acceptable. Ces méthodes ouvrent des portes pour la segmentation ou l'acquisition de scènes de grandes dimensions, ce qui sera essentiel dans un futur proche, car la demande tend vers un contrôle avec toujours moins d'erreur et une unité de mesure toujours plus petite. De nombreuses autres perspectives sont aussi envisageables, comme le secteur médical pour les CNN et contours actifs ou l'ensemble des applications industrielles pour les systèmes de vision. Cela ouvre à des applications encore non explorées.