

Dossier Technique

Année 2020



A.	Présentation de la société et de sa stratégie de R&D	3
I.	<i>Présentation générale de la société</i>	3

1. Le positionnement marché	3
2. L'organisation interne	4
II. Présentation de la stratégie de R&D	3
1. Objectifs de la R&D.....	3
2. Domaines d'applications	3
B. Méthodologie de valorisation.....	6
C. Éligibilité des opérations de R&D déclarées au CIR	7
I. Contexte et objectif de l'opération	5
II. Nature R&D de l'opération.....	6
1. Verrous technologiques	6
2. Etude de l'état de l'art	24
III. Contribution scientifique, technique ou technologique.....	30
1. Connaissances acquises sur la partie acquisition des données	31
2. Connaissances acquises sur la partie reconstruction 3D temps réel	33
IV. Démarche scientifique et travaux réalisés.....	35
1. Rappel des travaux antérieurs.....	36
2. Travaux 2020 : projet YOLO.....	42
Ressources humaines	53

I. *Présentation de la stratégie de R&D*

1. Objectifs de la R&D

L'idée du projet présenté dans ce document est le fruit de deux expériences, celle du porteur de projet en tant que spécialiste de la réalité virtuelle, réalité augmentée et de la capture de mouvement, et celle d'Audensiel Technologies Sud-Ouest implanté dans un bassin d'activités où le multimédia, l'IT ainsi que le secteur militaire sont dynamiques.

L'objectif de nos travaux est de pallier aux solutions existantes de réalité augmentée actuellement limitées en termes de détection des objets, et notamment en conditions de luminosité faible ou d'environnement enfumé. Pour ce faire, nous voulons proposer une solution de perception d'un environnement en 3D quelles que soient les conditions météorologiques (pluie, brouillard, fumée, etc.), quelles que soient les conditions de luminosité (y compris dans l'obscurité totale), ou encore en zones accidentées.

2. Domaines d'applications

Le fruit de nos recherches peut s'avérer particulièrement pratique, voire vital, dans de nombreux domaines dont les conditions d'exercices ne sont pas aisées (luminosité inadéquate pour l'œil humain, danger, discrétion, urgence de situation, etc.). Par ailleurs, le contexte actuel en termes de catastrophes climatiques et d'attentats en a renforcé l'intérêt.

En accord avec les domaines de compétence d'Audensiel sud-ouest les domaines d'applications sont les suivants :

2.1. Organismes de recherche

Parmi les besoins des organismes de recherche, on retrouve l'exploration des zones inaccessibles ou dangereuses pour l'Homme. Le système que nous proposons peut les aider dans la réalisation des études et l'archivage de modèles 3D afin d'étudier son évolution dans le temps. En effet, on envoie le drone pour numériser ces zones et les visualiser en temps réel.



Figure 1 : Explorer les zones inaccessibles par les Humains

2.2. Défense

Ce système pourra être utile pour le secteur de la défense. Il permet aux soldats de visualiser en réalité augmentée l'environnement reconstruit en temps réel du champ de bataille et de partager des informations entre eux. Il sera possible d'envoyer un drone afin d'inspecter, espionner, détecter

et localiser des cibles même dans les mauvaises conditions de vision (nuit, fumée, brouillard, etc.).

De plus, ce système permet de numériser un environnement même par un avion. Dans ce cas, le scanner sera aéroporté, et le modèle 3D sera affiché dans le cockpit.



Figure 2 : Visualisation en réalité augmentée dans le cockpit d'un avion (à gauche) et dans le champ de bataille (à droite)

2.3. Équipe de sauvetage

Afin de planifier un plan de secours, et détecter les chemins accessibles, détecter les positions des survivants, le drone scanner 3D sera envoyé dans les zones de catastrophes. Il numérise en temps réel l'environnement. L'équipe de sauvetage visualise ce qui se passe en même temps et peut intervenir immédiatement.



Figure 3 : Etudier l'environnement pour définir le plan de secours

2.4. Serious game

Le serious game permet à l'utilisateur de se former ou de s'entraîner dans les mêmes conditions que la réalité. Il est possible de simuler un entraînement militaire virtuel dans un environnement réel en utilisant notre système et en ajoutant par exemple des personnages ou des objets 3D intrus dans la scène. Ces modèles 3D intrus seront dotés d'une intelligence artificielle et seront capables d'interagir avec les personnes réelles.



Figure 4 : Entraînement en réalité augmentée (à gauche) et en réalité virtuelle (à droite)

I. Contexte et objectif de l'opération

Le porteur de projet et les ingénieurs d'Audensiel Technologies Sud-Ouest ont pu observer que le processus de numérisation 3D est actuellement un processus long. La technologie actuelle n'offre pas la possibilité de scanner en 3D et de visualiser un même objet concomitamment. Il est donc apparu nécessaire aux équipes de rechercher et développer une nouvelle technologie, un système qui devra permettre un rendu 3D en temps réel ou en temps différé. De plus, cette visualisation de l'environnement considéré devra être possible quelles que soient les conditions de visibilité. Ce système doit ainsi fonctionner dans le brouillard, la fumée, ou encore dans l'obscurité totale.

Grâce à la numérisation de l'environnement par un scanner 3D laser, on peut avoir un nuage de points représentant les dimensions et les aspects de l'environnement réel. Ce scanner 3D est porté par un drone aérien ou terrestre. Ce dernier sera semi-autonome et piloté par un utilisateur, ce qui se justifie car nous devons anticiper de potentielles pertes de connexion entre le pilote et le drone en raison des environnements à explorer. Après l'acquisition du nuage de points, notre système devrait être capable de traiter ces informations et reconstruire progressivement les surfaces 3D dans le but d'avoir un modèle 3D virtuel identique à l'environnement réel.

Le modèle 3D serait visualisé en temps réel, et affiché sur un dispositif de visualisation 3D comme les casques de réalité augmentée ou de réalité virtuelle. La Figure 5 présente un cas d'utilisation du système que nous souhaitons mettre en place. L'environnement est numérisé par un drone aérien. Et l'utilisateur visualise le modèle 3D recalé et affiché en maillage.

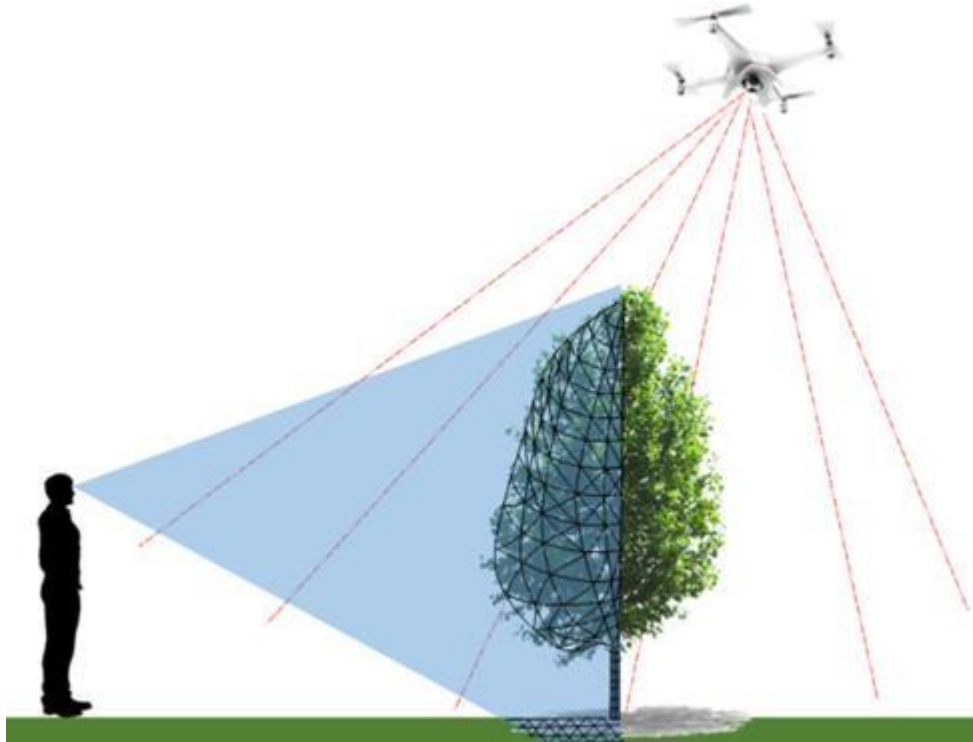


Figure 5 : Visualisation 3D en utilisant un casque de réalité augmentée d'un environnement scanné par un drone aérien

En 2019, nous nous sommes focalisés sur la réalisation d'une POC de reconstruction 3D d'un environnement réel. Ce projet s'est concentré sur l'acquisition de données en 3D et la reconstruction de l'environnement en 3D, avec notamment le positionnement de la caméra stéréoscopique directement sur le robot. A travers ces travaux, nous cherchons à développer un robot mobile terrestre autonome évoluant dans un environnement réel et capable de recréer cet environnement numériquement afin de le visualiser. Les objectifs de ce projet sont multiples.

- Le robot doit premièrement pouvoir se déplacer en autonomie sur terre. Cela veut dire que le robot doit prendre en considération son environnement et décider seul du parcours à réaliser.
- Le robot doit aussi scanner son milieu d'application, pour créer une représentation 3D et 2D de l'environnement qu'il découvre. Le scan de l'espace qui l'entoure va lui permettre de définir une projection de son environnement, à partir de laquelle...
- ... il va décider de l'endroit où il doit se déplacer pour compléter sa carte.

II. Nature R&D de l'opération

Les objectifs que nous visons nous mènent à des difficultés techniques et technologiques certaines, que nous avons tout d'abord identifiées puis confrontées à l'état de l'art pour chacune. C'est ce que nous vous présentons ci-dessous.

1. Verrous technologiques

Grands axes

Lors de nos travaux de recherche nous avons dû faire face à des verrous technologiques sur chacun des trois grands axes sur lequel repose notre projet.

1. L'acquisition des données :

- Dans certains cas de visibilité difficile, par exemple en cas de brouillard ou de fumée, l'environnement peut être masqué par des particules ou gouttes en suspension dans l'air qui limitent l'acquisition de celui-ci. Cette acquisition limitée ne permet pas de tenir compte réellement de l'environnement durant l'étape de reconstruction 3D.
- Le risque de saturation du réseau peut provenir d'un grand volume de données acquis par les différents capteurs et devant transiter jusqu'à l'outil de reconstruction 3D. Cette saturation peut ensuite provoquer la perte des données et ainsi affecter négativement le processus de reconstruction 3D, du fait qu'une partie des données peut s'avérer manquante dans le modèle 3D. De plus, les objets en mouvement perturbent les données collectées lors de la numérisation de ces dernières.
- La limite d'autonomie du drone peut négativement impacter le processus du scan tout en limitant à la fois les fonctionnalités mais aussi le nombre d'informations collectées.

2. La reconstruction 3D temps réel :

- Les problèmes liés à la numérisation et la perte de données peuvent en conséquence provoquer d'autres erreurs dans la reconstruction des surfaces en 3D qui de fait ne correspondent plus à la réalité du terrain. Cela peut s'avérer problématique en usage réel, selon le niveau de dangerosité de la situation.
- Tout objet en mouvement (animaux, robots, êtres humains, véhicules...) peut causer des perturbations dans le processus de numérisation allant jusqu'à provoquer des erreurs dans la construction 3D de l'environnement, qui présentera alors les mêmes risques que cités précédemment.

3. La visualisation 3D temps réel :

- L'exactitude relevée des coordonnées GPS se situe entre 0,5 et 5 mètres, ce qui pose des problèmes de justesse de l'information, car dans ce cas, nous ne pouvons pas avoir les coordonnées exactes de l'utilisateur (ou de l'objet) et ne pouvons par conséquent pas déterminer convenablement son point de vision. Par ailleurs, le recalage de l'image 3D géolocalisée sur l'environnement réel peut par effet de bord se retrouver erroné.

Enfin, ce problème peut être partagé par la partie « Reconstruction 3D temps réel », puisqu'il convient aussi de déterminer la position du robot vis-à-vis de l'environnement pour obtenir une bonne reconstruction de cet environnement.

- Il existe un temps de latence entre le traitement des données obtenues, la reconstruction 3D et la visualisation 3D. Cette latence ayant pour conséquence de fournir finalement une visualisation 3D en « temps réel différé », ce qui n'est donc logiquement plus du temps réel.

En complément à ces trois grands ensembles de verrous, nous devons aussi faire face à l'incapacité du robot de se retrouver dans un couloir, que celui-ci fasse 1 ou 10 mètres. Pour être plus clair, en cas de boucle du chemin emprunté par le robot, ce dernier se repère difficilement. C'est-à-dire qu'il y aura un delta, il ne sera pas capable de savoir où, et incidemment quand, précisément il a fait un tour complet. Nous observons donc un double décalage, qui est à la fois spatial et temporel.

Positionnement Marché :

Le marché de la réalité virtuelle s'élève à 895 millions de dollars en 2016 selon Strategy Analytics. La vente des casques de réalité virtuelle devrait exploser aussi pour cette année [CITATION FRE16 \ 1036]. Par ailleurs les casques HTC Vive, Oculus Rift et Sony Playstation VR représenteraient 77% du marché. Par contre, les casques de réalité virtuelle nécessitant un smartphone comme le [Samsung Gear VR](#) présentent la majorité des ventes grâce au prix bas en comparant avec les autres casques.

Selon le cabinet Digi Capital [CITATION RID15 \ 1036], le marché mondial de produit de réalité virtuelle et augmentée devrait atteindre 150 milliards de dollars en 2020 vu l'explosion des usages dans le grand public. La réalité augmentée représenterait la majorité avec 120 milliards de dollars tandis que la réalité virtuelle ne pèserait que 30 milliards de dollars.

Facebook a investi 2 milliards de dollars pour acheter le casque de réalité virtuelle Oculus Rift, et Google a investi 1,4 milliards dans la technologie de réalité mixte Magic Leap[CITATION Nic162 \ 1036].

D'autre part, le marché des drones est en train d'exploser, le chiffre d'affaire des drones professionnels était de 50 millions d'euros en 2014, 100 millions d'euros en 2015 et devrait atteindre plusieurs centaines de millions dans les 5 ans à venir [CITATION POU15 \ 1036].

Contexte scientifique et technique du projet :

L'objectif de cette partie est de réaliser une première analyse du contexte dans lequel se place le projet de visualisation 3D temps réel. Nous allons définir la réalité virtuelle et la réalité augmentée, et les dispositifs de visualisation 3D. De plus, nous allons exposer les méthodes d'acquisition des données 3D, de reconstruction 3D et les techniques de visualisation 3D temps réel.

1.1. Réalité virtuelle

Définition

La définition technique de la réalité virtuelle [CITATION BAr03 \ 1036] est : « la réalité virtuelle est un domaine scientifique et technique exploitant l'**informatique** (1) et des **interfaces comportementales** (2) en vue de simuler dans un **monde virtuel** (3) le comportement d'entités 3D, qui sont en **interaction en temps réel** (4) entre elles et avec un ou des utilisateurs en **immersion pseudo-naturelle** (5) par l'intermédiaire de canaux sensori-moteurs »

La partie (1) exprime la nécessité d'exploiter les potentialités de l'**informatique**, matérielles et logicielles afin de réaliser un environnement virtuel interactif.

Les **interfaces comportementales** (2) représentent les interfaces matérielles. Elles sont composées d'interfaces :

- Sensorielles qui informent l'utilisateur par ses sens de l'évolution du monde virtuel ;
- Motrices qui informent l'ordinateur de l'impact des actions motrices de l'homme sur le monde virtuel ;
- Sensori-motrices informent dans les deux sens.

(3) Le **monde virtuel** crée doit être interactif et en temps réel.

(4) **Interaction en temps réel** est obtenue s'il n'y a pas de décalage temporel entre l'action effectuée par l'utilisateur et la réponse du système.

(5) L'utilisateur doit être en **immersion pseudo-naturelle** dans l'environnement virtuel. L'immersion ne peut être naturelle car l'Homme a appris à agir d'une façon naturelle seulement dans un monde réel et non virtuel.

De plus, la réalité virtuelle exploite et développe quatre catégories de dispositifs d'interaction [CITATION Bur03 \l 1036] :

- Les dispositifs de capture de position et mouvement comme par exemple des capteurs, un gant de données, une combinaison...
- Les dispositifs de visualisation 3D comme par exemple un grand écran, un visiocasque...
- Les dispositifs de retour proprioceptifs et cutanés (touché, chaleur...).
- Les dispositifs d'entrée et de présentation sonore comme le son spatialisé, la reconnaissance vocale...



Figure 6 : Utilisateur immergé dans un monde virtuel en utilisant un casque, un gant et des capteurs

Domaines d'application

La réalité virtuelle peut être utilisée dans de multiples domaines. Plusieurs applications et produits sont disponibles sur le marché comme par exemple des applications de formation pour le secteur industriel, médical, ou même l'éducation. De plus, il y a aussi des systèmes de simulation pour l'entraînement militaire, d'entraînement de pilotage d'avion de ligne ou de chasse, ou d'entraînement de conduite de véhicule...

La réalité virtuelle est utilisée aussi dans le secteur des jeux vidéo. Par ailleurs, elle cible le secteur médical comme par exemple le traitement des phobies en mettant la personne dans une situation virtuelle ressemblante à une situation réelle qui lui fait peur.

De plus, en médecine il est difficile de passer de la théorie à la pratique. La réalité virtuelle offre un entraînement chirurgical pour les médecins et les étudiants afin de réaliser des opérations sensibles en virtuelle dans les mêmes conditions réelles.

Et de plus, dans le but de vendre un produit, des applications de réalité virtuelle offrent au consommateur la possibilité de tester et voir le prototype modélisé en 3D.

Il est possible aussi, de rapprocher des personnes dans différents endroits grâce aux réunions virtuelles. Chaque utilisateur est représenté par un avatar 3D contrôlé par lui-même. Il a la possibilité d'interagir avec les avatars présents dans la réunion.



Figure 7 : Entraînement de l'armée en utilisant des casques de réalité virtuelle

Casque de réalité virtuelle

Il existe plusieurs casques sur le marché [CITATION Gui16 \l 1036], prenons des exemples de l'Oculus Rift, le HTC Vive et PlayStation VR qui sera disponible prochainement.

Dans la première version de l'**Oculus Rift** DK1 [CITATION Ocu16 \l 1036], le casque possède un écran 7 pouces de résolution 1280 x 800 (640 x 800 pixels par œil). Les mouvements de la tête sont reconnus sur 3 axes grâce à un gyroscope, accéléromètre et un magnétomètre intégrés dans celui-ci.

Dans le DK2, la vibration et le flou cinétique causé par le mouvement ont été réduits. L'écran est amélioré en full HD¹. La version commerciale est sortie au premier trimestre 2016, possède un écran à diodes électroluminescentes organique (OLED²) en résolution 1200 pixels. De plus, les mouvements de la tête sont détectés sur 6 degrés de liberté.

Le casque **HTC Vive** [CITATION Jea16 \l 1036] permet à l'utilisateur d'être immergé dans une nouvelle dimension grâce à la définition de ses écrans OLED 1080 x 1200 pixels avec un champ de vision de 110° et une vitesse de rafraîchissement de 90 Hz. Il intègre plusieurs capteurs pour la détection de mouvement de la tête de l'utilisateur. Grâce à la présence de deux contrôleurs, l'utilisateur a la possibilité de se déplacer réellement sur plusieurs mètres.

¹ High definition

² Organic Light-Emitting Diode

PlayStation VR [CITATION Pla16 \l 1036] sera disponible sur le marché en octobre 2016, il possède un écran OLED d'une résolution de 1980 x1080 pixels, et une vitesse de rafraichissement de 120 Hz.

En revanche, ce dispositif ne sera compatible qu'avec la console de jeu PlayStation 4 développée par Sony, contrairement aux casques précédents qui sont destinés pour un usage sur PC.



Figure 8 : Casque de réalité virtuelle de gauche à droite : HTC vive, PlayStation VR, Oculus Rift

Le casque HTC Vive correspondra mieux aux besoins de notre projet de visualisation 3D. En effet, il offre plus de liberté liée aux mouvements de la tête et aux déplacements de l'utilisateur.

1.2. Réalité augmentée

Définition

La réalité augmentée consiste à ajouter des objets virtuels dans un environnement réel filmé en direct. L'utilisateur visualise les objets virtuels et réels ajoutés dans son champ de vision et qui coexistent dans le même espace [CITATION AZU01 \l 1036].

Un système de réalité augmentée doit respecter trois règles fondamentales, selon Ronald Azuma [CITATION AZU97 \l 1036] :

- Combiner le réel et virtuel
- Interagir de manière interactive en temps réel
- Être enregistré en 3 dimensions

Cette définition exclu le domaine du cinéma et de la post-production comme par exemple le collage de deux dimensions dans les effets spéciaux qui n'est pas en temps réel.

De nombreuses applications sont liées à la vision, tandis que la réalité augmentée peut augmenter n'importe lequel des cinq sens.

Dans le cadre de notre projet, cette définition vient renforcer la nécessité d'une approche en temps réel.

Paul Milgram et Fumio Kishino [CITATION Pau94 \ 1036] définissent la réalité augmentée comme un continuum³ qui relie le réel au virtuel en les considérant comme tous les deux complémentaires. Cela introduit le terme de réalité mixte qui se compose de la réalité augmentée et de la virtualité augmentée.

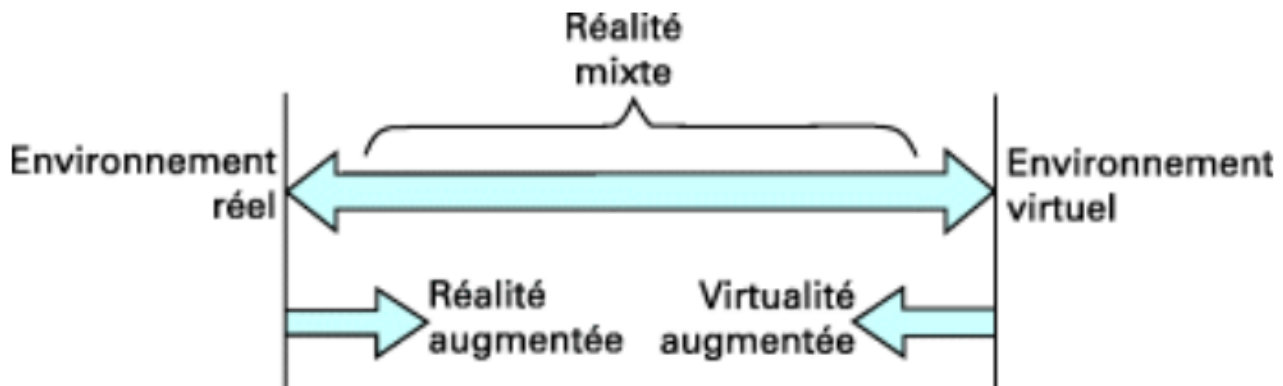


Figure 9 : Le continuum de Milgram

Domaines d'application

Les applications de réalité augmentée offrent la possibilité à l'utilisateur d'interagir avec les objets virtuels. Ces derniers sont visualisés à travers un dispositif possédant une caméra qui filme le monde réel en directe et recalc⁴ les objets virtuels dessus. Ces dispositifs peuvent être un smartphone, une tablette tactile, un casque de réalité augmentée, ou un ordinateur avec une webcam.

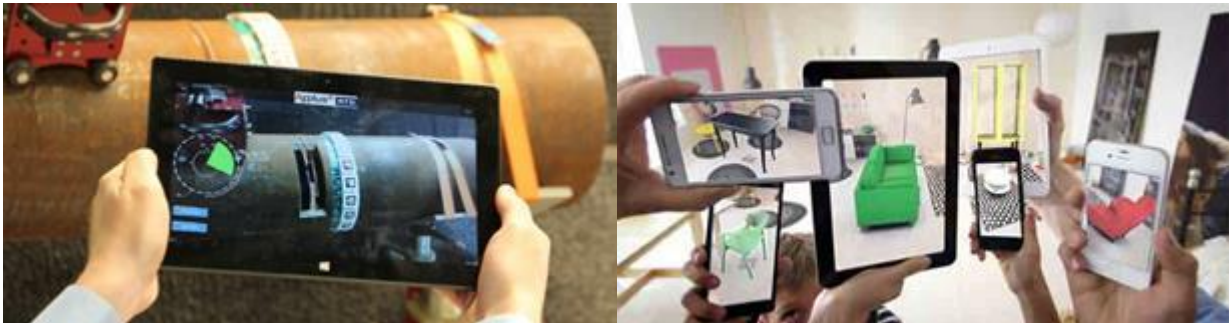


Figure 10 : Applications de réalité augmentée

Les applications disponibles touchent plusieurs domaines tels que les jeux vidéo, l'éducation, l'industrie, le militaire, le marketing, le médical...

Parmi les diverses applications on cite par exemple les applications de formation en réalité augmentée, qui facilitent l'apprentissage en utilisant les modèles 3D.

Il existe aussi des applications de maintenance industrielle qui apportent de l'aide à l'utilisateur. Quand ce dernier pointe son dispositif de réalité augmentée sur la machine à maintenir, plusieurs

³ Un continuum est un ensemble d'éléments tels que l'on peut passer de l'un à l'autre de façon continue

⁴ Le recalage visuel est la détermination des informations géométriques sur la scène permettant un placement cohérent des objets virtuels dans la scène réelle.

informations sont affichées. Ces informations concernent les pièces à entretenir, sont utiles et compréhensibles facilement par l'utilisateur.

Dans le secteur du marketing [CITATION Tar16 \ 1036], certaines entreprises choisissent de gamifier leur marque ou un produit, c'est-à-dire créer un jeu en utilisant un concept, un produit ou un slogan. Le consommateur manipule un objet en 3D qui représente le produit lié à une marque. Ce genre de jeux génère une liaison de plaisir et une bonne relation entre le consommateur et la marque.

De plus, le domaine du e-commerce utilise aussi le concept de la réalité augmentée pour offrir au consommateur une visualisation d'un produit en 3D et il peut également consulter les caractéristiques ainsi que les instructions de l'objet.

Dernièrement la NASA a utilisé un casque de réalité augmentée dans son projet Sidekick, afin d'offrir la possibilité à l'astronaute de communiquer en temps réel avec une personne sur terre [CITATION JUL16 \ 1036]. Grâce à la caméra embarquée sur le casque, l'astronaute peut voir en temps réel des informations concernant l'action effectuée par l'utilisateur distant.

Lunette de réalité augmentée

Nous allons utiliser dans notre projet des lunettes de réalité augmentée afin d'offrir une visualisation 3D et une liberté pour les mains de l'utilisateur.

Plusieurs lunettes sont disponibles sur le marché pour le grand public, l'industrie ainsi que pour le secteur militaire.

Les plus populaires sont les **Google Glass**. Cette paire de lunettes est équipée d'une caméra, d'un micro, d'un pavé tactile sur l'une des branches et d'un mini écran qui permet l'affichage des images 3D [CITATION Goo16 \ 1036].



Figure 11 : Casque de réalité augmentée Google Glass

Les plus récents sont les **HoloLens** de Microsoft. Ces lunettes de réalité augmentée sont disponibles uniquement pour les développeurs depuis le 1^{er} Avril 2016. Elles permettent d'intégrer des hologrammes simulés dans le champ de vision de l'utilisateur. De plus, elles possèdent trois processeurs : un CPU⁵ principale, un processeur graphique GPU⁶, ainsi qu'un processeur pour gérer

5 Central Processing Unit

6 Graphics Processing Unit

les hologrammes HPU⁷. Des capteurs de mouvements, une caméra, et un microphone sont embarqués dans le casque [CITATION Mic16 \l 1036].



Figure 12 : Casque de réalité augmentée HoloLens

Q-Warrior [CITATION Mat14 \l 1036] est un dispositif de réalité augmentée proposé aux troupes militaires. Il possède une meilleure autonomie comparée aux Google Glass. Il s'agit d'un monoculaire transparent qui affiche des données reçus en temps réel. Il utilise la transmission par satellite pour communiquer entre la base et les autres soldats. Ce système traite les données provenant d'une caméra de vision nocturne, et peut aussi recevoir les images capturées par des drones d'observation sur le champ de bataille.



Figure 13 : Monoculaire pour les soldats Q-Warrior

Le système **ClearVision** d'Elbit [CITATION Léo15 \l 1036] est un outil d'aide de vision pour les pilotes de chasse. Il possède un système de vision amélioré couplé avec un système de vision synthétique. Grâce à la combinaison de six différents capteurs (thermique, nocturne et caméra TV), le pilote peut voir grâce au casque **Skylens** l'environnement aéroportuaire, même en cas de mauvais temps ou encore de nuit. Les lunettes connectées **Skylens** offrent au pilote un affichage à tête haute de 26°.

⁷ Holographic Processing Unit



Figure 14 : Skylens pour les pilotes de chasse

Vu que les lunettes de réalité augmentée pour le grand public sont peu nombreuses, nous allons utiliser les lunettes **HoloLens** de Microsoft. Elles offrent un angle de vision de 120° et possèdent des capteurs de détection de mouvement de tête.

1.3. Réalité mixte

Définition

Contrairement à la réalité augmentée et virtuelle, la réalité mixte n'est pas une technologie ancienne. Le terme en lui-même fût évoqué par Paul Milgram et Fumio Kishino [CITATION Pau94 \ 1036] lors de leurs travaux sur le concept de continuum⁸ de la virtualité. Il s'agit à l'époque de qualifier tout ce qui se situe entre la réalité augmentée et virtuelle mais qui n'entre pas précisément dans ces deux catégories, comme l'illustre la Figure 9.

Aujourd'hui les casques de réalité mixte disponibles sur le marché sont plus proches de la réalité augmentée que de la réalité virtuelle, et sont par conséquent considérés comme une extension de la réalité augmentée. D'ailleurs, par abus de langage, la grande majorité des personnes et des journalistes englobent la réalité mixte dans la réalité augmentée.

Ces deux technologies ont en commun de rajouter de l'information numérique au monde réel. La principale différence entre la réalité augmentée et la réalité mixte est la capacité du casque de réalité mixte à prendre en compte l'espace réel, et ce, grâce aux capteurs intégrés au casque. De cette façon, là où la réalité augmentée ajoute de l'information numérique en superposition de l'environnement, la réalité mixte fusionne des objets de synthèse avec l'environnement réel sous la forme d'un hologramme qui peut interagir de façon cohérente avec l'environnement ainsi que l'utilisateur. Par exemple, si un objet réel s'intercale entre la personne et l'objet virtuel, il sera pris en compte et apparaîtra devant l'hologramme, avec la profondeur réelle.

Domaines d'application

Bien que le concept de la réalité mixte ait été introduit à la fin des années 2000, les premières démonstrations concrètes sont apparues à partir de 2015 avec l'arrivée du casque HoloLens de la société Microsoft. La réalité mixte est par conséquent une technologie naissante et encore difficile d'accès pour le grand public à cause du prix élevé des casques. Cette technologie est donc principalement tournée pour des applications industrielles et éducatives.

⁸ Un continuum est un ensemble d'éléments tels que l'on peut passer de l'un à l'autre de façon continue

Les similitudes avec la réalité augmentée se retrouvent aussi dans les domaines d'application de la réalité mixte avec une capacité de projection de l'information numérique dans l'environnement réel renforcé. On peut souligner les exemples notables d'applications suivants :

- Le 5 décembre 2017, la première intervention chirurgicale utilisant la réalité mixte a été réalisée. Les caméras qui se situent sur le casque analysent les gestes du chirurgien et l'interface propose plusieurs outils permettant de les interpréter ludiquement (par exemple : accès aux antécédents du patient durant l'opération). Cette technologie présente un intérêt tout particulier pour le chirurgien car il est habillé de façon stérile et ne peut donc pas toucher un clavier ou utiliser une souris.
- Plusieurs sociétés du bâtiment ont commencé des partenariats avec la société Microsoft pour permettre d'exploiter la réalité mixte afin de faciliter la communication entre les différents acteurs durant la phase de conception. Cette communication se traduit par la visualisation directement sur le chantier des éléments qui ont été modélisés et qui vont être ajoutés à la construction (par exemple : passage des éléments de plomberie). De plus, la réalité mixte vient rajouter la possibilité de déplacer les éléments modélisés pour avoir un rendu en temps réel.
- Pour les mêmes raisons, la réalité mixte est de plus en plus présente dans la conception de prototype car celle-ci permet de projeter directement le prototype dans son environnement, ou encore de projeter les éléments en conception du prototype sur la maquette initiale.

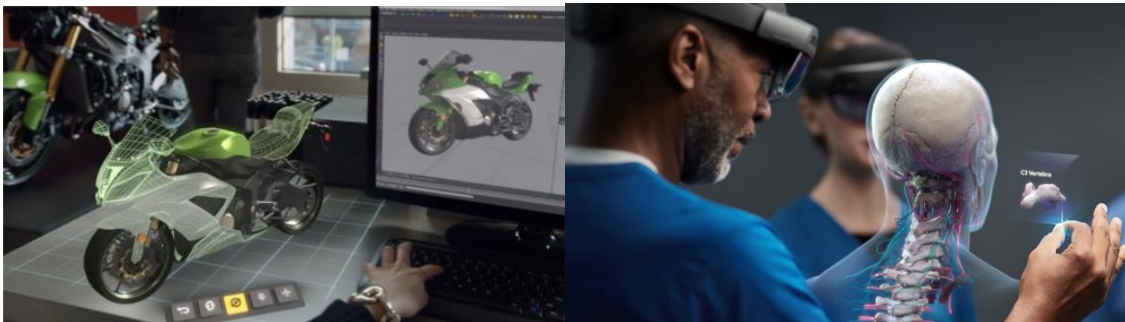


Figure 15 : Exemples d'application de la réalité mixte

Casques de réalité mixte

Tout comme les casques de réalité virtuelle, deux types de casques de réalité mixte existent : les casques de réalité mixte autonomes et les casques de réalité mixte qui nécessitent un appareil externe (smartphone). Bien que ces derniers soient moins chers que les casques autonomes (sans intégrer le prix du smartphone, qui est assez élevé car doit intégrer des performances élevées), les dispositifs utilisant un smartphone ne présentent pas le même rendu que les casques autonomes. En effet, c'est l'ensemble caméra + écran du smartphone qui est utilisé pour réaliser l'affichage de l'environnement réel avec l'ajout de l'information numérique. De cette façon, le rendu perçu par l'utilisateur est limité par ces deux périphériques, là où les casques autonomes permettent de percevoir directement l'environnement extérieur grâce à une visière transparente. Ainsi, nous limitons notre présentation des casques de réalité mixte à ceux équipés de visière transparente c'est-à-dire les casques de réalité mixte autonomes et un casque de réalité mixte qui nécessite d'être connecté à un pc car l'utilisateur peut percevoir directement l'environnement extérieur.

Le marché des casques de réalité mixte autonomes est dominé par les casques HoloLens de Microsoft et Magic Leap One de la société MagicLeap, cependant d'autres sociétés ont aussi développé des casques de réalité mixte. Ci-dessous sont listés les principaux casques de réalité mixte.

Le casque de réalité mixte **Lightfield** est développé par la société Avegant. Celui-ci est encore à l'état de prototype et n'est par conséquent pas disponible à la vente (commerciale). Ce casque est pour l'instant utilisable uniquement en étant branché sur un ordinateur qui va réaliser les opérations de création des hologrammes. Il intègre la technologie Light Field destinée à augmenter l'immersion et le réalisme perçus par les utilisateurs dans la réalité mixte. Ce visiocasque filaire peut afficher, par exemple, des prototypes virtuels ou des schémas en 3D.



Figure 16 : Casque de réalité mixte Lightfield

Le casque de réalité mixte **AjnaLens** est développé par la société Dimension NGX. Il est le moins cher des casques de réalité mixte autonome, cependant il n'intègre pas les fonctionnalités d'eye tracking et de hand tracking. Intégrant des fonctions d'intelligence artificielle (AI), de cartographie environnementale et de hologrammes, cette startup basée en Inde vise les entreprises avec un casque MR rivalisant ceux des marques les plus chères. Le champ de vision de 90 degrés du AjnaLens (encore en phase de développement) est également plus large que celui de HoloLens ou de Magic Leap.



Figure 17 : Casque de réalité mixte AjnaLens

Le casque de réalité mixte **Magic Leap One** est développé par la société Magic Leap. Cette startup a été créée en 2010 pour sortir la première version du casque le 8 Août 2018. On notera que cette startup est connue principalement pour avoir réalisée plusieurs levées de fonds records auprès d'investisseurs de premier rang (Google, Andreessen Horowitz, Alibaba, ...) pour un total de dépassant les 2 milliards de dollars depuis sa création. Le Magic Leap One est doté de technologie Light Field et est livré avec un Lightpack (mini PC portable) pour alimenter le visiocasque. Le potentiel de ce casque MR crée beaucoup d'attentes de la part du public, et les développeurs travaillent désormais sur la production de contenu (applications, jeux, etc.).



Figure 18 : Casque de réalité mixte Magic Leap One

Le casque de réalité mixte **HoloLens 2** est développé par la société Microsoft et succède à la première version en y apportant plusieurs améliorations. Parmi celles-ci, on trouve le dernier capteur Kinect, une puce AI personnalisée pour améliorer la performance MR, un champ de vision plus large, et un design plus confortable. Deux nouvelles fonctionnalités clés sont également présentes : le tracking “inside-out” (ne nécessitant pas de capteurs externes) ainsi que l’intégration de lentilles holographiques.



Figure 19 : Casque de réalité mixte HoloLens 2

Le tableau ci-dessous synthétise les informations des casques de réalité mixtes venant d’être présentés :

Casques et lunettes de réalité mixte	Catégorie	Année de sortie	Prix (PDSF ⁹) en €	Champ de vision (FOV ¹⁰)	Eye Tracking	Hand Tracking
Avegant Lightfield	Pour PC	-	-	-	Non	Oui
Dimension	Autonome	2018	1350	95°	Non	Non

⁹ Prix de Détail Suggéré par le Fabricant

¹⁰ Field Of View

NGX AjnaLens						
Magic Leap One	Autonome	2018	2000	40°	Oui	Oui
Microsoft HoloLens 2	Autonome	2019	3100	52°	Oui	Oui

1.4. Acquisition des données 3D

L'acquisition des données 3D consiste à utiliser des instruments pour relever les formes, les dimensions et les aspects de la surface d'un objet. Il existe deux méthodes d'acquisition :

1. Une méthode directe qui restitue en temps réel les points acquis sous forme de coordonnées.
2. Une méthode d'acquisition indirecte comme par exemple la photogrammétrie et la photo modélisation qui se base sur l'exploitation du support photographique afin d'extraire les coordonnées spatiales. Et ceci en utilisant plusieurs algorithmes de restitution.

Nous nous intéressons dans notre étude aux méthodes d'acquisitions en temps réel tel que celles associées à un scanner laser 3D. D'ailleurs, il existe trois catégories principales qui seront détaillées par la suite :

1. Le laser par triangulation
2. Le scanner par temps de vol
3. Le scanner par décalage de phase

Le laser par triangulation

Un scanner 3D laser est un système de balayage laser produisant des données caractérisées et distribuées en nuage de points [CITATION AXE99 \l 1036]. Il permet de gagner du temps sur les échantillonnages¹¹ complexes jusqu'à 1 000 000 de points par seconde. La densité des points dépend de plusieurs facteurs tels que la position du scanner, la vitesse de déplacement, son champ de vision et la fréquence. Le scanner laser par triangulation utilise un faisceau laser et le pointe sur la surface de l'objet à scanner. Enfin il utilise un appareil photo pour situer le point. Le point laser, l'appareil photo et l'émetteur laser forment un triangle, c'est pour cela que cette technique est appelée triangulation [CITATION Sca16 \l 1036]. Ce type de scanner est généralement utilisé pour scanner les objets qui ne dépassent pas la taille d'une voiture.

¹¹ L'échantillonnage consiste à passer d'un signal à temps valeurs (valeurs mesurées à intervalles réguliers).

continu en une suite discrète de

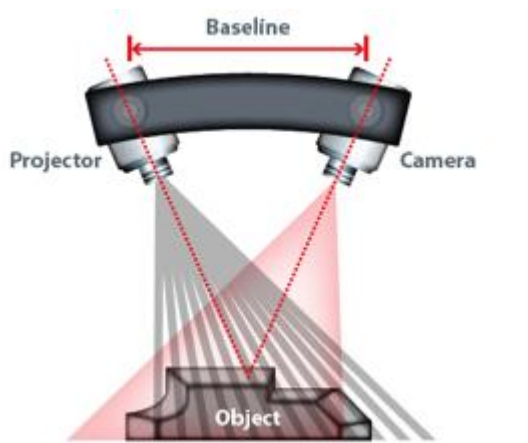


Figure 20 : Principe de triangulation

Le scanner par temps de vol

Le scanner par temps de vol LiDAR (Light Detection And Ranging, détection et télémétrie par ondes lumineuses) est un appareil actif de télédétection optique qui utilise un faisceau laser pour sonder le sujet. Le principe est d'envoyer des impulsions laser sur une cible et d'enregistrer le temps d'émission et de réception. Puisque la vitesse de la lumière est connue, le temps de retour permet de déterminer la distance parcourue par la lumière [CITATION Sca16 \l 1036]. Le télémètre inclus dans l'appareil détecte un point à la fois dans la direction pointée.

Afin de scanner tout le champ de vision, il faut scanner point par point et changer la direction de vue à chaque mesure, en utilisant une rotation soit de l'appareil soit en utilisant un système de miroirs rotatifs. D'ailleurs le scanner par temps de vol peut mesurer entre 10 000 à 100 000 points par seconde. La densité des points peut être ajustée en fonction de la demande de l'utilisateur. Cette technique est utile pour la mesure de longues distances.

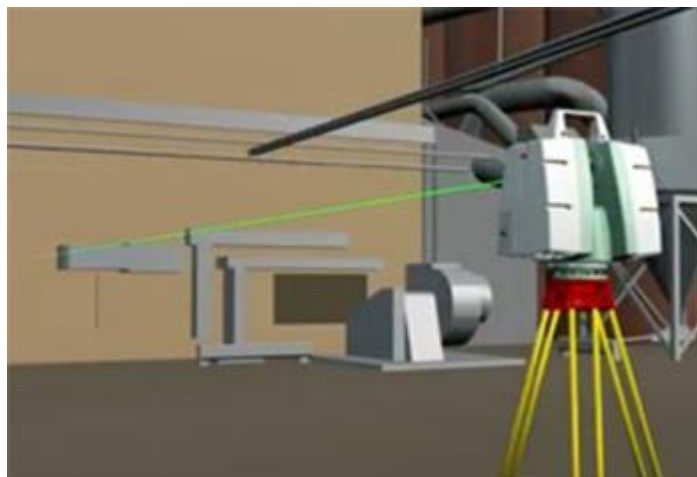


Figure 21 : Scanner laser 3D "Leica"

Le scanner par décalage de phase

Pour la technique du scan par décalage de phase [CITATION Mat13 \l 1036], le scanner émet dans un premier temps un rayon laser en continu qui sera réfléchi au contact de l'objet. Dans un second temps, la distance sera calculée grâce au décalage de phase entre l'émission du rayon et de sa réception. Cette technique permet de collecter jusqu'à 1 million de points par seconde, une quantité

de données plus précise que les scanners 3D à temps de vol. Mais les scanners par décalage de phase ont une portée plus réduite qui ne devra pas dépasser une distance de 120 mètres.

Conclusion

Le tableau suivant présente un récapitulatif des techniques de scan laser 3D :

Technique de scan laser 3D	Nombre de données collectées par seconde	Distance portée
Scan par triangulation	1 000 000	Quelques mètres
Scan par temps de vol	10 000 à 100 000	Jusqu'à 300 mètres
Scan par décalage de phase	1 000 000	Maximum 120 mètres

L'intérêt d'utiliser un scanner laser 3D pour notre projet est que ce dernier permet la mesure de millions de points par minute rapidement quelles que soient les conditions de visibilité (comme dans la nuit, ou par mauvais temps). Dans notre cas, nous avons besoin de scanner un environnement vaste et ouvert, à l'intérieur des bâtiments ou souterrains, le plus rapidement possible et porté par un drone à longue distance. C'est pour cela que nous allons nous référer à la technique du scan par temps de vol ou LIDAR, mais les conditions dynamiques d'utilisation imposent une expérimentation pour vérifier la faisabilité de cette hypothèse.

Aujourd'hui, les logiciels fournis avec les scanners laser 3D ne sont pas capables de réaliser un modèle 3D automatique et en temps réel. Le traitement des nuages de points et la modélisation de l'environnement se font en post-traitement, après que l'opération du scan se soit terminée. C'est pour cela que nous souhaitons proposer la possibilité d'augmenter la performance de traitement et de l'exploitation du nuage de points, et ceci afin d'**automatiser le processus de modélisation progressive en même temps que l'acquisition**.

Les scanners LIDAR

Nous présentons ci-dessous quelques scanners 3D Lidar disponibles sur le marché :

a. VUX-1HA scanner laser 3D compact

Le scanner VUX-1HA est un scanner laser léger, compact et robuste, qui possède une haute précision de scan. Ce scanner est facile à monter dans des plateformes mobiles, parfaitement adapté pour les missions d'études et d'inspections basées sur des voitures, des trains ou des robots...



Figure 22 : Scanner laser 3D VUX-1HA

b. Leica Pegasus : Backpack

Leica Pegasus : Backpack est une plateforme mobile pour la capture réelle d'environnements intérieurs et extérieurs. Cette plateforme présentée dans la Figure 23 possède plusieurs capteurs et caméras et un scanner 3D Lidar. Le Pegasus : Backpack est conçu pour scanner l'environnement rapidement avec une précision de 600 000 points par seconde. Le nuage de points est combiné avec les images prises par la caméra afin d'avoir plus de détails sur l'environnement numérisé.



Figure 23 : Leica Pegasus : Backpack

c. Leica Pegasus : Two

Leica Pegasus : Two est une solution complète mobile pour la numérisation d'un environnement extérieur. Elle peut être portée facilement sur les voitures, le train, le bateau...

Elle se compose de plusieurs capteurs et un scanner 3D Lidar. De plus, cette plateforme peut capturer une vue complète en 360° grâce aux 7 caméras intégrées.



Figure 24 : Leica Pegasus : Two

Leica geosystems propose des solutions logicielles pour accompagner les utilisateurs de ses scanners laser 3D. Ces logiciels offrent un transfert direct des données collectées par le scanner vers le PC de l'utilisateur. Après l'import du nuage de points, l'utilisateur peut visualiser en 3D à partir de plusieurs perspectives ainsi que manipuler le modèle du scan. De plus, il est possible de reproduire une représentation en 3D du nuage de points.

Les drones scanner 3D

Des nouvelles technologies et innovations cherchent à améliorer les méthodes de cartographie, d'inspection et d'étude des environnements urbains ou souterrains. Des nouvelles technologies de drone scanner 3D vont remplacer la main d'œuvre afin d'éviter tout risque pour l'Homme quand il exploite des environnements dangereux.

a. MINEFLY

MINEFLY, développé par Clickmox, est un drone pour scanner et cartographier les mines souterraines et l'intérieur des bâtiments. Ce drone aérien est équipé d'un scanner laser 3D, d'une ampoule LED¹², des senseurs sonar, d'une caméra HD¹³, et d'une caméra faible résolution. Il est compact et léger. Il est aussi équipé d'un système d'évitement de collision composé de plusieurs capteurs ultrasoniques. Ce système utilise l'algorithme SLAM (Simultaneous localization and mapping) pour combiner les données issues du laser et les positions des senseurs afin de générer un nuage de points 3D. À partir des nuages de points, il génère la carte en temps réel ou non, cela dépend de la préférence de l'utilisateur. De plus, la caméra HD et l'ampoule permettent à l'utilisateur d'avoir un retour visuel durant l'opération du vol du drone.

12 Light Emitting Diode

13 High Definition



Figure 25 : MINEFLY drone aérien scanner

b. RIEGL RiCOPTER

RIEGL RiCOPTER est un drone aérien scanner laser 3D qui utilise la technique LiDAR pour les missions d'inspection. Ce drone est robuste, fiable et peut porter des différents senseurs jusqu'à 16 kg. Cet octocopter possède des bras pliables et un châssis absorbant de choc, permettant un vol stable et un atterrissage sûr. Mais son temps de vol ne dépasse pas les 30 minutes.



Figure 26 : RIEGL RiCOPTER drone aérien scanner

2. Etude de l'état de l'art

2.1. Nuage de points

La représentation commune des mesures du scan 3D est le nuage de points [CITATION Wu103 \l 1036]. Chaque point est représenté par des coordonnées x, y et z et une couleur (r, g, b, a).

La figure ci-dessous présente un nuage de points d'un environnement scanné par un système Lidar aéroporté.

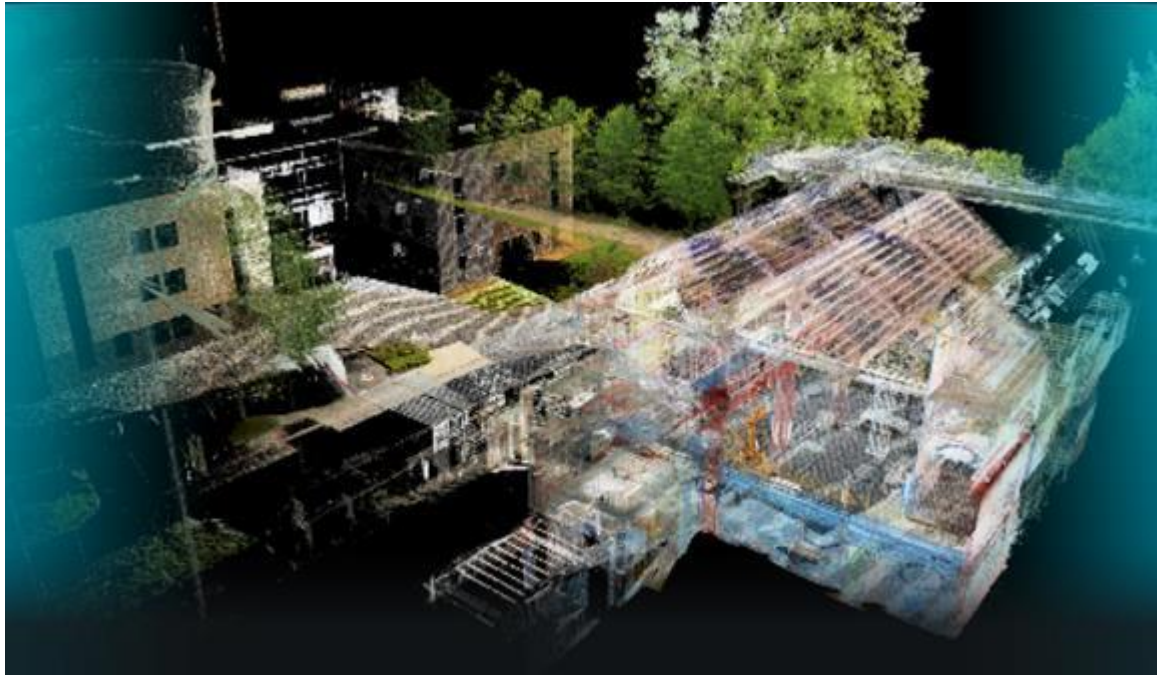


Figure 27 : Nuage de points

La technologie du Lidar collecte les mesures de distance à un objet en utilisant un télémètre laser. Ce dernier envoie des faisceaux lumineux par un miroir oscillant afin de balayer la surface de la terre. La distance des surfaces est calculée à partir de la différence de temps entre le moment de l'émission du faisceau laser et le moment du retour du même signal réfléchi par l'objet touché [CITATION Lau10 \l 1036]. La répétition des mesures à une fréquence élevée permet d'obtenir un nuage de points décrivant les objets. Les points sont enregistrés sous forme de coordonnées x , y et z .

Le système Lidar est associé à un système de géolocalisation par satellite (GPS) et une unité de mesure inertielle (IMU). Ces unités de mesures permettent d'enregistrer les coordonnées géolocalisées, l'accélération et les déplacements afin de définir la position et l'orientation du système au moment de l'émission des faisceaux laser. Ces données sont indispensables pour le calcul du nuage de points géolocalisé.

L'avantage de la technologie Lidar est que ses faisceaux laser sont capables de pénétrer la couverture végétale pour enregistrer des points au sol. D'ailleurs, lorsque le faisceau laser est arrêté par la végétation, il continue sa trajectoire avec une intensité plus faible.

La compréhension et l'interprétation du nuage de points collecté grâce au scanner laser 3D est difficile. En effet, il faut passer à la reconstruction 3D et la présentation géométrique des formes. D'où le besoin de transformer le nuage de points en un ensemble de triangles qui formeront le maillage 3D.

2.2. Reconstruction 3D

Une fois le nuage de points obtenu après la numérisation 3D, il est possible d'effectuer des nettoyages, des assemblages, et d'autres actions à partir des logiciels existants comme par exemple le 3D Reshaper, ou PointCap. Ces logiciels sont dédiés au traitement des nuages de points, et de reconstruction des surfaces. Ils ont plusieurs fonctions telles que la réduction

intelligente et suppression des points où la densité est élevée dans certaines zones, l'homogénéisation de la densité, la réduction de bruit de mesure, texturer et colorer les nuages, fusionner plusieurs nuages, ...

4.2.1. Post-traitement

Les données acquises du scan 3D ne peuvent être ni comprises et ni utilisées directement, elles doivent être traitées afin d'être utilisées. Les traitements à effectuer sont [CITATION Lor09 \l 1036] :

- *Filtrage du bruit* : le bruit correspond à des points artefacts et qui ne correspondent pas à des points réels. Ils doivent être éliminés par un algorithme de filtrage du bruit. Cette méthode [CITATION MFo06 \l 1036] comprend trois phases.
 1. À partir d'un maillage bruité, la première étape est de calculer la transformée en distance TD du maillage,
 2. la deuxième étape est la réalisation du filtrage de la TD¹⁴,
 3. et la troisième est la triangulation de la TD.
- *Triangulation* [CITATION Jea92 \l 1036] : une fois les données filtrées, elles peuvent être triangulées. En effet, les emplacements des points sont calculés dans l'espace, puis les points de proximités sont calculés pour enfin construire le triangle et générer le maillage.
- *Segmentation et nettoyage* : les données acquises doivent donc être segmentées pour enlever la partie non désirée et ne garder que l'objet d'intérêt.
- *Recalage manuel / automatique et fusion des vues* : afin de fusionner plusieurs surfaces 3D présentant différents points de vue du même objet [CITATION Joa07 \l 1036], il est nécessaire de trouver au moins 3 points de repères. Dans le cas d'utilisation d'un scanner à temps de vol, des mires réfléchissantes sont posées dans l'environnement afin de les utiliser comme des points de repères pour le recalage automatique des vues.
- *Bouchage des trous* : l'absence des données dans le nuage de points peut causer des trous dans le modèle 3D. Dans ce cas, ces trous doivent être bouchés par la création des facettes permettant de relier ses arêtes.
- *Gestion de la texture* : la plupart des scanners 3D ne font pas l'acquisition de la couleur. Dans ce cas, ce genre de scanner est couplé à un appareil photographique pour prendre des photos et les utiliser comme texture pour le modèle 3D.

La phase du post-traitement est la plus coûteuse en temps, elle représente 80% du temps durant la procédure de la numérisation 3D.

Nous présentons par la suite quelques algorithmes de reconstruction des surfaces 3D à partir des points.

4.2.2. Triangulation Delaunay

La triangulation d'un ensemble de points dans un plan, est la création d'un ensemble de triangles qui ne s'intersectent pas, dont les sommets sont ces points. Les algorithmes de triangulation performants maximisent le nombre d'arêtes.

L'algorithme de Delaunay [CITATION Jon02 \l 1036] a pour objectif de construire un maillage triangulaire à partir d'un ensemble de points. Le principe est de trouver une triangulation afin

14 TD : la transformée en distance

qu'aucun des sommets ne soit à l'intérieur du cercle circonscrit d'un des triangles, et que les angles ne soient ni trop petits ni trop grands.

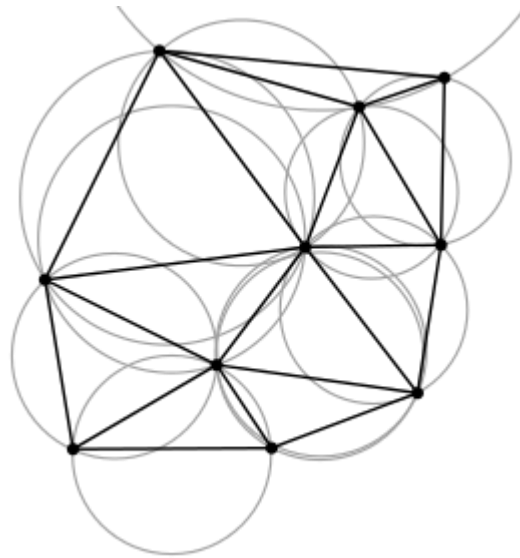


Figure 28 : Une triangulation de Delaunay avec les cercles circonscrits en gris [CITATION Tri V 1036]

4.2.3. Marching Cubes

L'algorithme du Marching Cubes [CITATION Har87 \I 1036] permet l'obtention de surfaces 3D à partir d'un ensemble de points. Ces surfaces 3D correspondent à un ensemble de triangles approximant la surface réelle.

Afin de représenter un volume, on divise l'espace 3D en cubes élémentaires. Après on fixe un seuil de densité et on parcourt l'espace cube par cube. Pour chaque cube, on compte le nombre de sommets au-dessus du seuil fixé au début. Ensuite, suivant ce nombre, on crée une table qui présente la combinaison des topologies possibles de triangles affichés à l'intérieur du cube. Chaque configuration correspond à un ensemble de facettes tracées à l'intérieur du volume. Ces triangles permettent donc la représentation de la surface du volume à déterminer. Ce volume est un cube qui possède 8 sommets, 12 arêtes et chaque sommet peut prendre 2 états. Dans ce cas il existe $2^8=256$ configurations possibles.

On peut se ramener à 15 configurations de base grâce aux symétries de rotation et aux inversions des points internes et externes.

On appelle donc un point interne, le sommet dont la densité est supérieure ou égale au seuil. Et le point externe, le sommet dont la densité est inférieure strictement au seuil.

La figure 29 présente la table des 15 configurations de base.

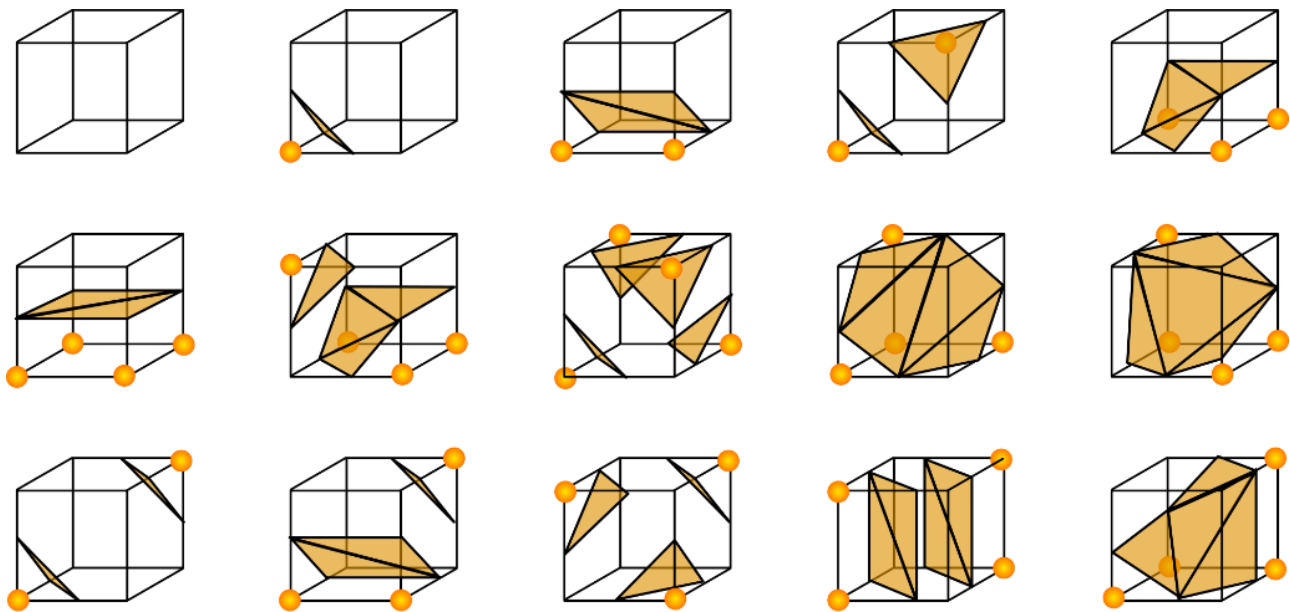


Figure 29 : Les 15 configurations de triangles [CITATION htt \ 1036]

L'algorithme du Marching Cubes nécessite un temps de calcul relativement faible, mais peut amener à la création des surfaces ayant un aspect en « marches d'escalier », dans ce cas il faut ajouter un algorithme de lissage.

4.2.4. Marching Tetrahedra

L'algorithme du Marching Tetrahedra [CITATION LuT12 \ 1036] dérive de l'algorithme du Marching Cubes. Son principe est de diviser le cube élémentaire en six tétraèdres. Chaque tétraèdre se comporte comme un volume élémentaire possédant 4 sommets et 6 arêtes. Chaque sommet peut prendre 2 états ce qui donne 16 configurations d'intersections possibles entre le volume et la facette. On peut se ramener à 8 configurations grâce aux cas symétriques.

Le fonctionnement de l'algorithme est le suivant : on parcourt chaque volume élémentaire du maillage. Ensuite, pour chacun de ces volumes et pour chaque tétraèdre, on compare les valeurs des sommets par rapport au seuil fixé au début. En fonction des résultats, un index binaire de 4 bits est créé. Le bit est égal à 1 si la valeur du sommet est inférieure au seuil, sinon il est à 0. Ensuite, pour chaque arête intersectée, on calcule par interpolation linéaire, les coordonnées du point d'intersection. Enfin, il suffit de dessiner et de stocker la facette.

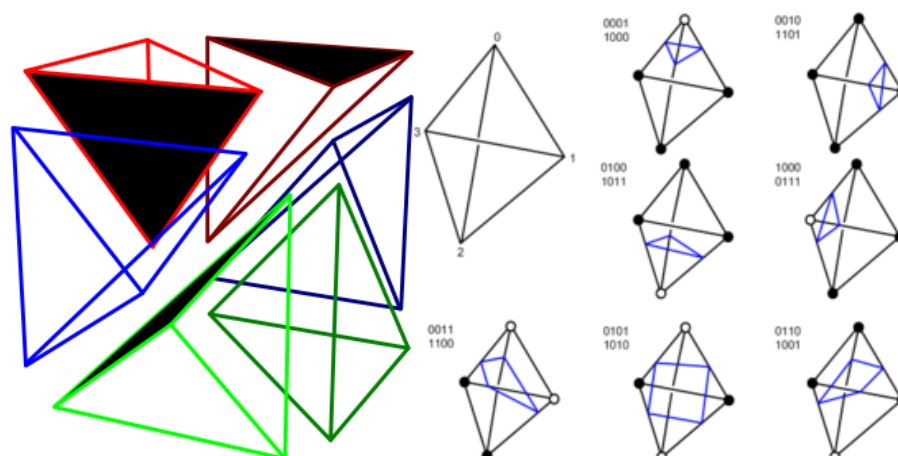


Figure 30 : Décomposition d'un cube en tétraèdres [CITATION Mar V 1036]

Contrairement au Marching Cubes, l'algorithme du Marching Tetrahedra peut causer des bosses artificielles sur la surface 3D reconstruite, car l'interpolation est faite sur la diagonale d'une face et non sur la face elle-même. De plus, il est plus simple que l'algorithme du Marching Cubes car il ne nécessite pas la création de la table ni sa consultation.

Les algorithmes présentés précédemment ne permettent pas la reconstruction 3D en temps réel. Il s'agit de la création des surfaces 3D à partir d'un nuage de points. Et ces algorithmes de post-traitement ne correspondent pas à notre démarche.

C'est pour cela que nous allons chercher à développer un algorithme innovant qui optimise le nombre de points afin de reconstruire progressivement les surfaces en 3D en temps réel.

2.3. Visualisation 3D temps réel

Les surfaces 3D obtenues par le traitement des données en temps réel seront visualisées sur différents supports. Nous nous intéressons aux technologies permettant la visualisation 3D et l'interaction en temps réel tels que les dispositifs de visualisation de réalité virtuelle ou réalité augmentée.

Par la suite nous exposons les différentes techniques pour la visualisation 3D.

4.3.1. Visualisation 3D sur écran PC

La Visualisation 3D sur un écran PC est la visualisation des images dynamiques séquentielles en 3 dimensions en utilisant un moteur 3D¹⁵ temps réel. Le moteur 3D offre à l'utilisateur la possibilité de changer le point de vision comme la position et l'angle et d'effectuer des modifications sur le modèle 3D.

4.3.2. Visualisation 3D en réalité virtuelle

En utilisant un périphérique de visualisation en réalité virtuelle, la perception de l'environnement 3D sera en relief. Grâce aux dispositifs de détection de mouvement, l'utilisateur a la possibilité d'interagir et se déplacer dans l'environnement. Il sera immergé totalement dans ce monde virtuel.

4.3.3. Visualisation 3D en réalité augmentée

La visualisation 3D en réalité augmentée est l'incrustation des images virtuelles issues du modèle 3D sur les images réelles filmées en temps réel par une caméra vidéo.

Le modèle virtuel est recalé sur le monde réel tout en respectant les mesures, les distances, la luminosité et les conditions réelles. L'utilisateur visualise en hologramme les objets virtuels.

Actuellement aucune application de numérisation disponible ne propose une visualisation temps réel du modèle 3D scanné par un système Lidar. Les applications fournies avec les scanners ou les drones scanner 3D offrent seulement la visualisation du nuage de points en temps réel sur un simple

¹⁵ Un moteur 3D est un composant logiciel qui crée des images matricielles à partir de coordonnées tridimensionnelles.

écran. Nous allons donc mener des recherches afin de développer **la visualisation du modèle 3D géolocalisé en temps réel**.

4.3.4. Méthode de recalage visuel temps réel

Le recalage visuel est la détermination des informations géométriques de la scène permettant un placement cohérent des objets virtuels dans la scène réelle. Cette méthode se décompose en deux étapes [CITATION Deh08 \l 1036] :

1. Une étape de traitement d'images pour l'obtention des mesures et la position de l'objet virtuel.
2. Une étape d'estimation afin de déterminer la pose et l'orientation de l'objet à ajouter.

Par ailleurs, afin d'aligner les objets virtuels avec le monde réel, il existe plusieurs méthodes :

- *Recalage avec des images 2D* : les images 2D ou les marqueurs sont localisés grâce au seuillage de l'image et la détection des régions spécifiques présentes dans l'image. Les informations géométriques détectées permettent d'identifier la position et la déformation de l'image et de se servir de ces informations pour l'affichage de l'objet 3D.
- *Détection des propriétés de l'image* : plusieurs informations de bas niveau sont utilisées tels que les points, les segments de droite, les contours, les textures... Ces informations sont calculées d'une image à une autre dans la séquence vidéo. Elles permettent de trouver les informations géométriques afin d'aligner les objets virtuels sur les objets réels [CITATION Mal08 \l 1036].
- *Système de capteur de position et de mouvement* : il s'agit de repérer la position des capteurs dans l'espace tout en calculant leurs positions de déplacement et l'orientation [CITATION Mal08 \l 1036]. Par exemple les capteurs magnétiques, les capteurs ultrasoniques, les capteurs inertiels...

Le projet vise à se placer dans des conditions de visibilité dégradées : brouillard, fumée, faible luminosité, etc. Dans ces conditions, les dispositifs de type optique actuels ne permettent pas d'atteindre nos objectifs concernant la qualité des images acquises dans le but de calculer les informations nécessaires pour l'alignement par exemple les contours, les points de repères, les textures... afin de recalculer le modèle virtuel dans le monde réel.

C'est pourquoi afin de répondre au besoin du projet, nous tenter de développer un algorithme de recalage et visualisation 3D en réalité augmentée, afin d'aligner le modèle 3D qui se reconstruit progressivement dans le monde réel.

III. Contribution scientifique, technique ou technologique

Au cours de l'année 2018, nos travaux précédents nous avaient notamment permis de solutionner une partie des problématiques liées à l'acquisition et à la reconstruction de l'environnement en 2D d'un point de vue matériel et logiciel. Ces solutions ont pu être mise en place au travers d'un prototype afin de les tester dans un environnement réel. Aussi, nous avons mené une première réflexion afin de rendre autonome l'exploration de l'environnement par le robot.

De plus, deux projets de réalité virtuelle ont été développés durant cette même année afin d'initier les travaux sur la partie visualisation 3D en temps réel. Ces projets nous ont permis de prendre en

main le matériel, et de vérifier l'adéquation de celui-ci pour répondre à la problématique de visualisation d'un environnement reconstruit en 3D avec un casque de réalité virtuelle.

L'année 2019 a elle aussi été riche d'enseignements, notamment sur les deux sous-domaines suivants :

- L'acquisition des données,
- La reconstruction 3D temps réel.

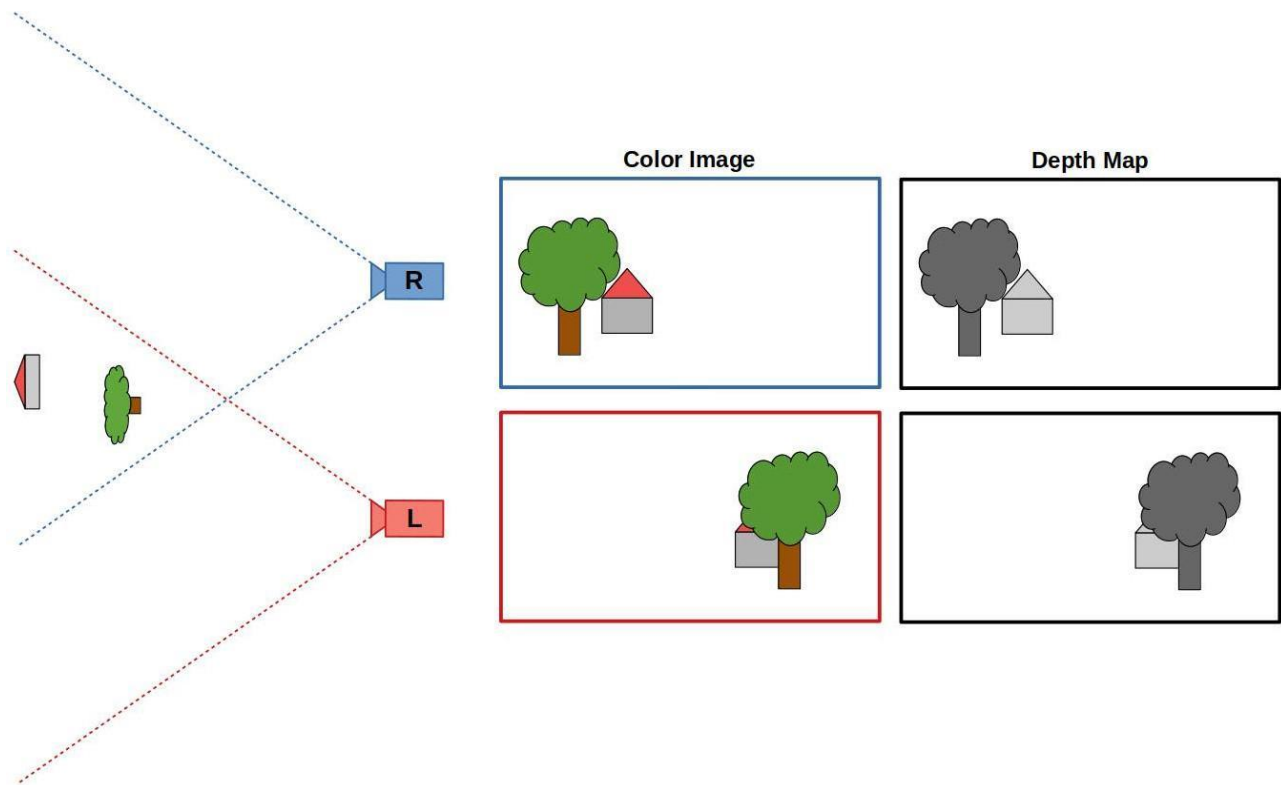
Nous détaillons ci-dessous l'ensemble de ce que nous avons appris.

1. Connaissances acquises sur la partie acquisition des données

Au cours de l'année 2019, nous avons exploré une nouvelle approche pour l'acquisition de données afin de réaliser une reconstruction 3D de l'environnement. Cette approche utilise le principe de la mesure stéréoscopique et ouvre deux nouvelles perspectives pour le projet, à savoir :

1. La possibilité de réaliser du SLAM 3D et donc de réaliser la reconstruction d'un point de vue géométrique de l'environnement 3D ;
2. L'ajout de la composante couleur à l'environnement. En effet le système d'acquisition se rapprochant du système de perception visuel de l'être humain (capteur passif), il est donc possible de récupérer l'information de la couleur de l'environnement. Et ainsi de reconstruire un environnement plus facilement interprétable par un utilisateur humain ;

La mesure stéréoscopique utilise le principe de la triangulation de l'environnement afin d'appréhender la notion de profondeur. La triangulation peut se résumer à la perception d'une même scène à partir de deux points de vue différents afin de déterminer la distance qui sépare la scène des points de prise de vue. Dans le cas de l'être humain les prises de vues sont réalisées par les yeux, dans le cas de notre système ce sera une caméra stéréoscopique (2 caméras solidaires d'un même axe) qui assurera ce rôle. La Figure 26 illustre le principe de la mesure stéréoscopique : une scène (représentée par un arbre et une maison) est perçue depuis deux caméras (en rouge et bleue), dont la scène est représentée par les images couleurs (« Color Image » en anglais). À partir de ces deux prises de vue (et connaissant la distance séparant les deux caméras et la distance focale du système), il est possible de déterminer des cartes de distances (« Depth Map » en anglais), où les objets les plus sombres sont les objets les plus proches des caméras.



La caméra que nous utilisons dans le cadre du projet est la caméra ZED fabriquée par la société Stéréolab (cf. Figure 31), et dont les principales caractéristiques sont indiquées dans la Figure 32. Celle-ci s'avère compatible avec la carte électronique Jetson TX2 qui est utilisée sur le robot terrestre mobile du projet, et qui a permis de réaliser un POC en 2018 en utilisant un LiDAR 2D.



Figure 31 :

Size and weight	Dimensions: $175 \times 30 \times 33$ mm Weight: 159 g
Individual image and depth resolution (pixels)	HD2K: 2208×1242 (15 FPS) HD1080: 1920×1080 (30, 15 FPS) HD720: 1280×720 (60, 30, 15 FPS) WVGA: 672×376 (100, 60, 30, 15 FPS)
Depth	Range: 1-20 m Format: 32 bits Baseline: 120 mm
Lens	Field of View: 110° $f/2.0$ aperture

Figure 32 :

Le LiDAR 2D est donc remplacé par la caméra stéréoscopique cette année dans notre POC afin d'explorer une nouvelle approche pour la reconstruction 3D. Cependant, à terme, ces deux moyens d'acquisitions sont complémentaires et permettront, lorsque fonctionnant de façon couplée, d'améliorer la fiabilité et la précision de l'environnement reconstruit en 3D. En effet, le LiDAR que nous avons utilisé permet uniquement l'acquisition de la distance de l'environnement sur un plan 2D mais de façon plus précise que la caméra stéréoscopique. En revanche, la caméra stéréoscopique permet d'apprécier la notion de couleur ainsi que d'étendre l'acquisition à la 3D. Il convient donc pour l'année 2020, de réaliser la fusion des données acquises par ces deux capteurs pour arriver à une reconstruction 3D précise et en même temps d'améliorer l'autonomie (du point de vue de la robotique mobile) du robot dans le cas d'exploration de l'environnement en cas de défaillance d'un des capteurs.

2. Connaissances acquises sur la partie reconstruction 3D temps réel

Pour réaliser la reconstruction 3D de l'environnement, nous nous appuyons sur les méthodes de SLAM. Nous rappelons que les méthodes de SLAM permettent de répondre à la problématique suivante : « *Un agent mobile a besoin d'une carte pour, dans un même temps, y placer ses observations en fonction de sa position (cartographie) et y déduire sa position en fonction de ses observations (localisation)* ». Plusieurs approches de méthode de SLAM sont proposées dans la littérature scientifique et peuvent être regroupées en trois grandes familles :

- Les méthodes de SLAM utilisant un filtre de Kalman ;
- Les méthodes de SLAM utilisant un filtre particulaire ;
- Les méthodes de SLAM utilisant une approche « graph-based ».

Ces trois grandes familles tirent leurs noms des méthodes d'estimation qui permettent de limiter les incertitudes sur la position à la fois de l'agent mobile, mais aussi de l'environnement.

Beaucoup d'implémentation de ces méthodes dans des outils de SLAM sont mis à disposition librement par des équipes de recherche, allant des petits laboratoires de recherche aux sociétés privées telles que Google, et en passant par des projets collaboratifs en ligne. Nous avons réalisé une veille des principaux outils existants (plus de 80 outils analysés), afin de sélectionner les plus

pertinents pour notre projet. L'un des principaux critères de sélection a été la compatibilité avec la caméra stéréoscopique qui fait l'objet d'une étude plus poussée cette année. Cependant, si on se projette plus loin dans la chronologie du projet Hypérion, il sera nécessaire d'utiliser un outil permettant de fusionner l'information de plusieurs capteurs d'acquisition de l'environnement (e.g. Caméra/LiDAR). Actuellement, peu d'outils de SLAM permettent de réaliser cette fusion, et souvent au prix d'une importante consommation de mémoire et de calcul. Ce qui nous amène sur deux autres critères dont nous tenons compte pour la sélection de l'outil de SLAM : la consommation de la ressource ainsi que la disponibilité des codes sources (sous licence libre). Ce dernier point nous permet à la fois de connaître précisément l'approche ainsi que les traitements réalisés par la méthode, mais aussi de pouvoir la faire évoluer sans contrainte de licence pour qu'elle soit plus en adéquation avec notre problématique. Enfin, un autre critère important de sélection de l'outil de SLAM est sa compatibilité avec le middleware ROS. Celui-ci, étant initialement développé par la communauté robotique pour des finalités de recherche et d'éducation, commence à trouver sa place dans l'industrie pour sa simplicité de mise en œuvre ainsi que la grande communauté qui développe et teste de nouvelles fonctionnalités.

De cette façon, plusieurs outils ont pu être sélectionnés, même si ne remplissant pas tous les critères à chaque fois, et sont listés ici :

- Google's Cartographer
- ORBSLAM 2
- RGBD-SLAM 2
- RTAB-Map
- S-PTAM
- Stereo DSO

Ces outils n'ont pas fini d'être comparés et devraient faire l'objet d'une comparaison précise durant l'année 2020 pour aboutir sur une publication scientifique, et surtout le choix de l'outil le mieux adapté au projet Hypérion. Les travaux réalisés en 2019 se focalisant uniquement sur la caméra stéréoscopique, nous avons utilisés deux outils de SLAM : ORBSLAM 2, et celui que propose le fabricant de la caméra, par simplicité d'utilisation mais aussi afin de créer une baseline avec laquelle réaliser la comparaison des outils de SLAM.

Enfin, nous avons aussi commencé à aborder la problématique de la communication entre le robot mobile et le serveur. Jusqu'à maintenant la solution que nous utilisons (utilisation de VNC) était temporaire car ajoutant une importante latence entre le moment d'acquisition de l'environnement et le moment où il est perçu sur un écran connecté au serveur. En effet, ce système permet de prendre le contrôle du robot et donc de visualiser les traitements qu'il réalise, soit la reconstruction de l'environnement en local. On comprend bien alors qu'énormément de données qui ne sont pas pertinentes pour la reconstruction de l'environnement sur le serveur transitent entre le robot et le serveur (consignes envoyées par le serveur vers le robot, données du système d'exploitation du robot, bureau du système d'exploitation, ...). De plus, cette solution ne permet pas de mettre en commun les environnements perçus par plusieurs agents mobiles (robots, drones, ...) et est limité à la capacité de traitement embarquée sur le robot. Cependant, cette solution avait été sélectionnée pour sa simplicité de mise en œuvre afin d'avancer sur des problématiques d'acquisition et de reconstruction locales sur le robot.

La solution que nous envisageons cette année est donc toujours d'utiliser un système de communication sans fil (e.g. Wifi), mais cette fois de s'appuyer sur les protocoles de communications inter-machines de ROS. En effet, ROS permet de faire communiquer des traitements déportés sur plusieurs machines, de plus nous utilisons déjà ROS pour réaliser les traitements depuis l'acquisition par les capteurs de l'environnement jusqu'à sa reconstruction sur le robot. Il convient alors de créer un réseau ROS sur plusieurs machines afin que celle-ci partagent leurs données, de cette façon nous limitons les données qui vont transiter entre serveur et agents mobiles car nous avons le contrôle dessus. Et aussi, l'ajout de nouveaux agents mobiles au sein de réseau se fera de façon plus naturelle. Toute la stratégie revient alors à créer ce réseau mais aussi de contrôler précisément

les données devant être envoyées au serveur qui réalisera la centralisation des données envoyées par les agents mobiles, ainsi que la reconstruction 3D.

Cependant, cette solution qui en théorie est prometteuse ajoute aussi son lot de difficultés dont la principale est la compatibilité matérielle et logicielle entre les différentes machines (serveurs et agents mobiles). Afin d'illustrer cet exemple nous prenons le POC actuelle du projet Hypérion où la carte électronique embarquée sur le robot (Jetson TX2) utilise une architecture ARM pour le CPU et une architecture Pascal pour le GPU. À cela s'ajoute les contraintes d'utilisation de la caméra ZED avec ROS qui font que la version du système d'exploitation utilisée date de 2016 et donc correspond à une version ancienne de ROS. Sur le serveur nous avons d'autres types de composants étant donné que celui-ci n'est pas conçu pour de l'embarqué (faible consommation) mais plutôt pour réaliser de la performance. Ainsi les architectures CPU et GPU sont différentes. De plus afin de bénéficier des développements les plus récents en termes de logiciel et de performance, celui-ci utilise une version plus récente de Linux ce qui entraîne une version différente de ROS. Cet exemple n'étant pas exhaustif, d'autres problèmes de compatibilité s'ajoutent à ceux déjà évoqués.

Pour répondre à ce problème de compatibilité, nous nous sommes tournés vers l'utilisation de conteneurs qui permettent d'accroître la flexibilité et la portabilité d'exécution des applications. En effet, les conteneurs connaissent un engouement récent important pour les gains de temps en termes de développement qu'apporte cette solution aux entreprises. À titre d'exemple d'utilisation, nous pouvons citer l'intégralité des services de Google ainsi que l'ensemble des applications sur smartphone. Le fonctionnement des conteneurs peut se résumer à « emballer » dans un conteneur l'application ainsi que les dépendances, et de standardiser les échanges avec la machine au travers du noyau d'un système d'exploitation déjà présent sur la machine (par exemples : noyaux Linux, Android, Windows, ...). En ce sens, l'utilisation de la solution au travers de ROS que nous envisagions ne dépend plus des architectures matérielles et sera donc compatibles sur tous types de machines (serveurs, agents mobiles, ...) si un noyau Linux est présent. Dans le cas où celui-ci n'est pas présent il est possible de le virtualiser à partir d'un système d'exploitation différent, tel que Windows. La solution retenue pour le projet Hypérion utilise la technologie Docker, solution de conteneurisation devenue très populaire depuis quelques années et étant la référence actuellement. Tout le travail consiste ensuite à développer les images de conteneurs qui intègrent ROS et pouvant travailler sur plusieurs machines dans le but d'y intégrer les POC réalisés sur le robot terrestre. Nous développons davantage la construction des images de conteneurs dans la partie **Error: Reference source not found** : Projet Docker ROS, qui est le point de départ pour l'intégration des POC du robot terrestre dans un environnement Docker, et par conséquent la réalisation d'une solution qui nous semble pérenne pour la communication entre les agents mobiles et le serveur.

IV. Démarche scientifique et travaux réalisés

Les travaux avancés en 2018 sur le projet nous avaient permis de mieux définir le sujet et les questions qu'il soulève. Nous avons pu en traiter certaines, telles que la sélection des outils de réalité virtuelle en adéquation avec le projet, et la mise en place d'une POC d'un robot terrestre autonome évoluant dans un environnement réel reconstruit en 2D.

Nous souhaitons en 2019 étendre le déplacement autonome du robot ainsi que la reconstruction à un environnement 3D. Cependant, de nombreux points restent également à étudier, tels que la mise en commun des données acquises par plusieurs agents mobiles et le portage vers des supports de réalité virtuelle et augmentée.

1. Rappel des travaux antérieurs

1.1 Projet SLAM 3D

Le projet SLAM 3D a permis de réaliser la reconstruction 3D de l'environnement à l'aide de la caméra stéréoscopique présentée dans la partie 5.1. Celle-ci équipe le robot mobile terrestre utilisé l'année dernière pour le projet de robot mobile terrestre autonome utilisant un LiDAR 2D. La première étape a été d'intégrer la caméra sur le robot. Pour cela le support de la carte électronique ainsi que celui de la caméra ont été conçus par CAO (cf. Figure 29), et ont été imprimés avec une imprimante 3D. Plusieurs contraintes de conceptions ont été prises en compte pour leurs conceptions :

- La possibilité de rotation de la caméra suivant 2 axes (Pan et Tilt) ;
- L'intégration des servomoteurs pour permettre les rotations ;
- La compatibilité avec le LiDAR 2D, c'est-à-dire ne pas gêner l'acquisition du LiDAR pour les projets futurs de fusion des données Lidar/caméra.

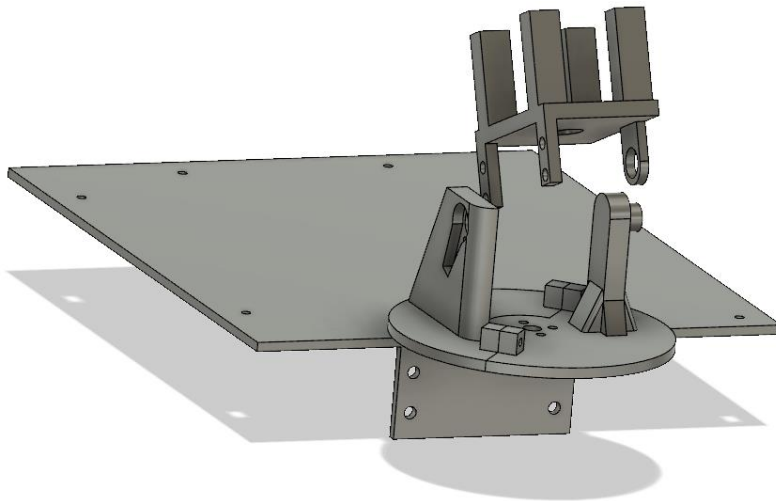


Figure 33 : CAO des supports caméra et carte électronique.

De cette façon la caméra a été intégrée sur le robot mobile, et son orientation (rotations Pan et Tilt) peut être contrôlée au travers d'actionneurs que l'utilisateur commande. La Figure 30 montre l'intégration du support ainsi que la caméra, la carte électronique et le LiDAR sur la base roulante.

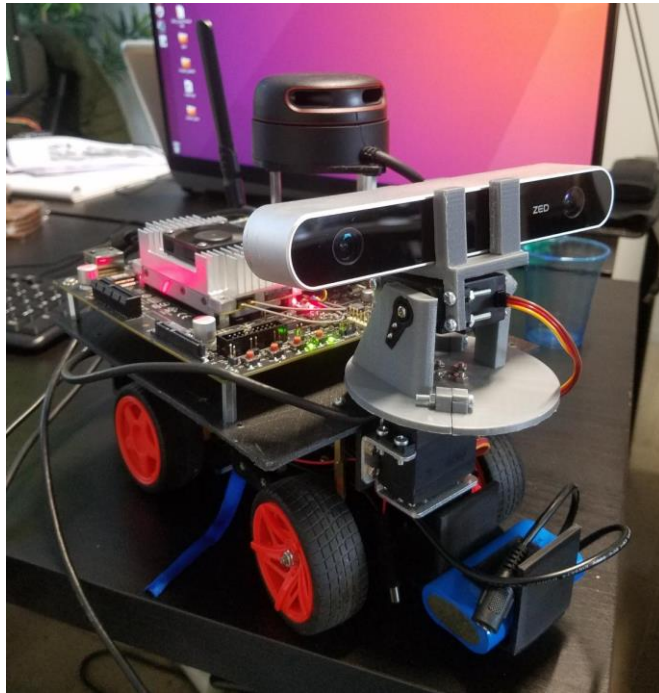


Figure 34 : Photode robot mobile terrestre équipé de la caméra et du LiDAR

La reconstruction 3D est réalisée avec l'outil de SLAM 3D du fabricant de la caméra que nous intégrons au middleware ROS. Celle-ci utilise les images acquises par la caméra stéréoscopique afin de déterminer la carte de profondeur de l'environnement, et donc la profondeur de chacun des pixels des images. Ce qui permet de placer les pixels des images en couleur dans l'espace 3D à partir du point de vu de la caméra. De cette façon, chacune des images acquises par la caméra est représentée par un nuage de points, où chacun des points est représenté par ses coordonnées dans l'espace 3D ainsi que par la couleur du pixel qui lui est associé. L'outil de SLAM permet de mettre en correspondance des ensembles de points à partir de plusieurs acquisitions d'images. Ainsi, les points nouvellement acquis par la caméra se rajoutent aux points déjà présents dans l'environnement 3D. On parle alors de représentation *dense* de l'environnement, illustrée par la Figure 35 qui représente le résultat de l'acquisition avec la caméra ZED après construction du nuage de points.

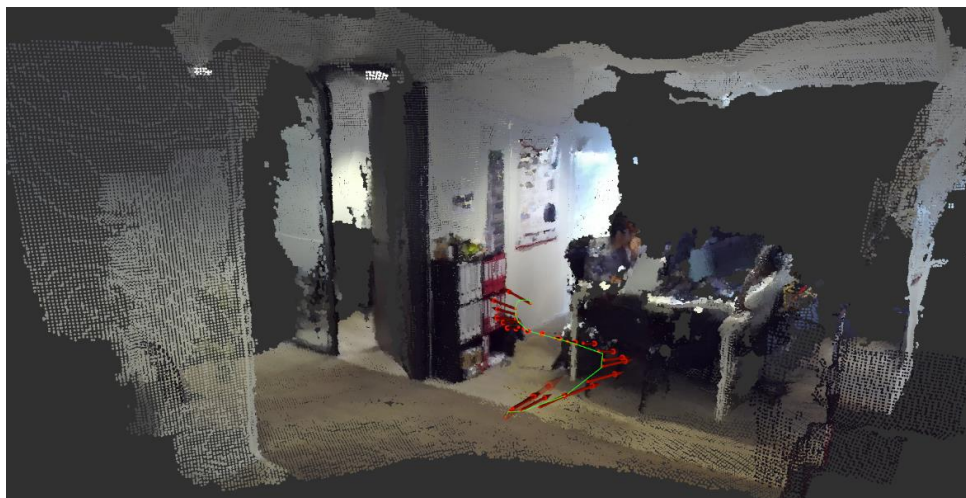


Figure 35

La représentation dense de l'environnement 3D permet à la fois d'obtenir une représentation visuelle facilement interprétable par un humain, mais surtout de créer un maillage des points et ainsi définir des surfaces de l'environnement 3D (cf. Figure 36)

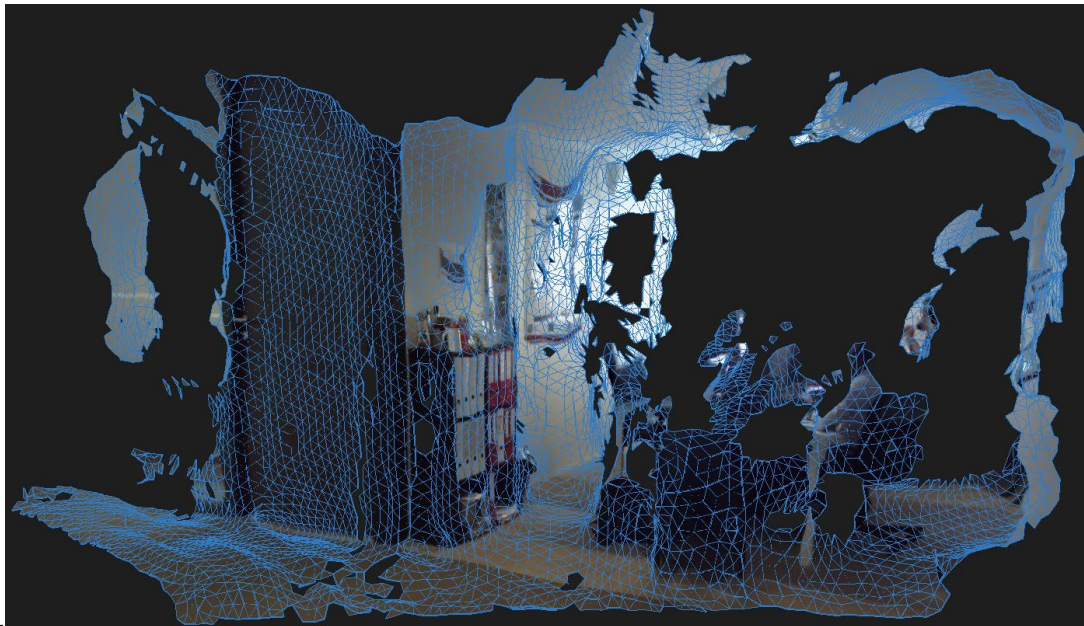


Figure 36 : Représentation du maillage en bleu, sur l'environnement reconstruit en 3D.

À partir de cette représentation 3D de l'environnement, nous souhaitons obtenir une carte 2D afin de permettre au robot mobile terrestre de pouvoir se déplacer dans son environnement. Pour cela nous réalisons une projection de la reconstruction 3D à l'altitude de la caméra stéréoscopique sur un plan 2D. De fait nous obtenons une carte 2D de l'environnement exploré ainsi que les obstacles détectés par le robot terrestre, c'est-à-dire les obstacles à la hauteur du robot qui empêcheraient le passage du robot terrestre. La Figure 33 représente l'interface avec laquelle nous observons le retour visuel du robot. À droite sont situées l'image couleur et la carte de profondeur, et à gauche la superposition du nuage de points 3D et de la carte 2D permettant au robot de planifier sa trajectoire. La région grise de la carte 2D correspond à la zone explorée par le robot et les régions noires aux obstacles détectés. Dans l'exemple de la Figure 33 nous avons disposé 2 obstacles proches du robot (un sac à dos et une pile de cartons) afin de vérifier que les obstacles étaient bien détectés et projetés sur la carte 2D.

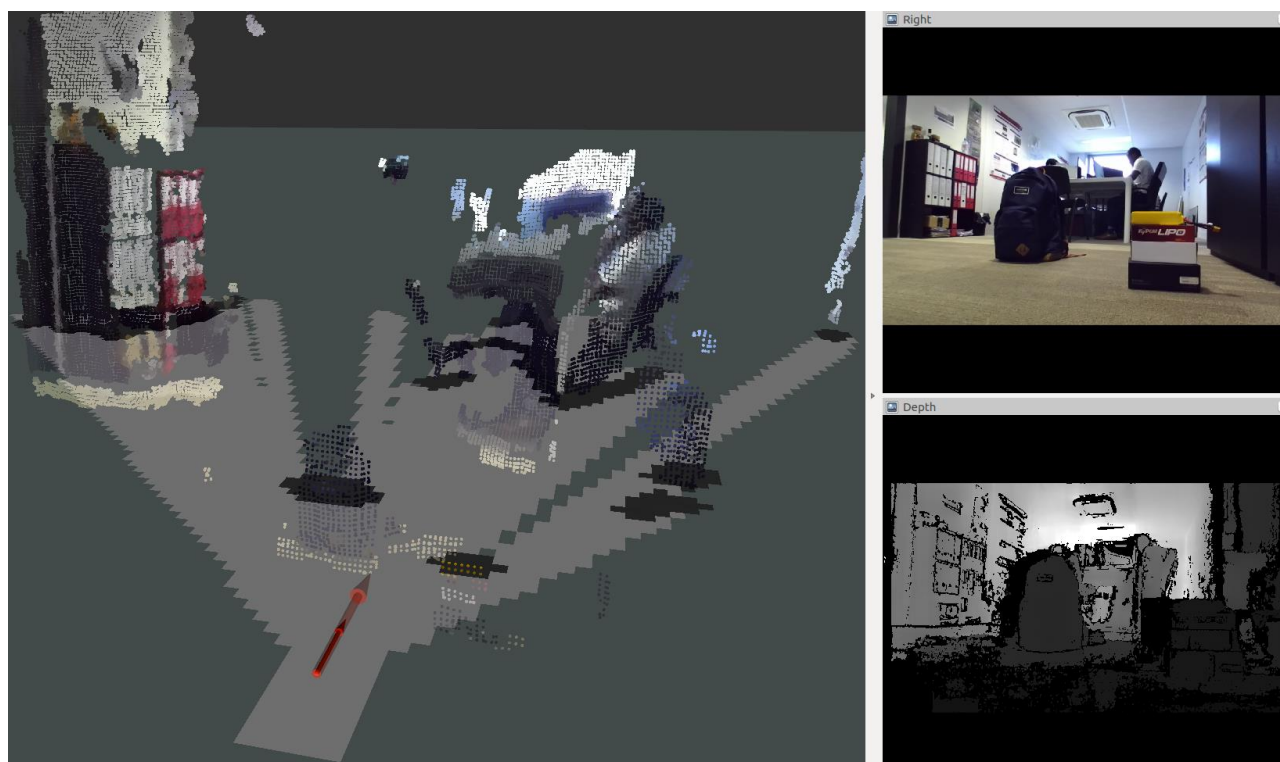


Figure 37 : Interface de visualisation de la reconstruction du robot. Image couleur en haut à droite. Carte de profondeur en bas à droite. Superposition de la carte 2D (en gris et noir) avec le nuage de points 3D à gauche.

Ainsi nous pouvons mettre en place la planification de la trajectoire du robot qui est réalisée de plusieurs façons. Le robot peut être contrôlé manuellement avec les touches d'un clavier (tout comme l'orientation de sa caméra). Il peut aussi planifier sa trajectoire en évitant les obstacles à partir d'une localisation sur la carte 2D définie par l'utilisateur (clic de la souris directement sur la carte 2D dans l'interface de visualisation). Et enfin, il peut aussi naviguer de façon autonome dans le but d'explorer un environnement en ciblant les plus grandes frontières entre la région explorée et inexplorée. Ce dernier mode étant activé par défaut jusqu'à l'intervention de l'utilisateur (soit avec le clavier, soit au travers de l'interface de visualisation).

Ainsi un POC est mise en place dans ce projet, qui permet de réaliser la reconstruction 3D d'un environnement réel. Cependant cette reconstruction est réalisée localement sur le robot ce qui ne permet pas à un utilisateur de le visualiser en temps réel à distance.

1.2 Faisabilité de la détection d'objets

Dans ce projet, nous nous intéressons à la détection d'objets et plus précisément à l'identification d'objets dans une scène. Ce projet est une première exploration pour permettre au robot d'évoluer dans des environnements plus difficiles et y effectuer des tâches aussi complexes. Ces tâches s'avèrent intéressantes dans le cas du projet Hypérion afin :

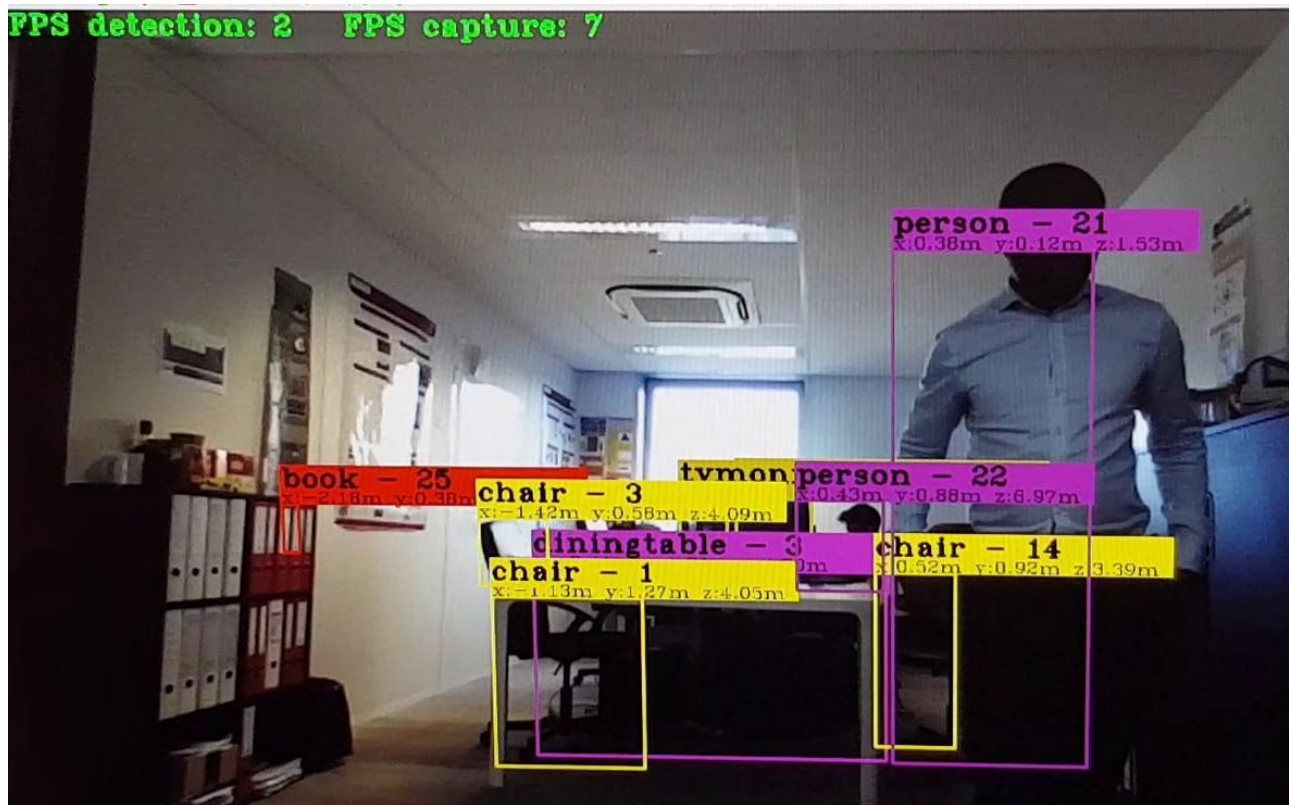
- D'affiner la reconstruction 3D des objets pouvant être identifiés ;
- De permettre de définir des tâches précises au cours de l'exploration des agents mobiles. Par exemple la détection de personne pour des applications de secourisme ;
- D'améliorer la compréhension de la scène qui entoure l'agent afin d'éviter les éléments pouvant nuire à sa mission. Par exemple l'utilisation du Lidar du projet ne permet pas de

détecter les vitres et ne donne pas une estimation précise de son environnement dans le cas de surface réfléchissantes. Nous supposons que pour pallier à ce problème, il est possible de fusionner les données du LiDAR avec celles d'un autre capteur (e.g. caméra) permettant d'identifier le contexte et les objets présents dans l'environnement.

Actuellement, la détection et l'identification d'objets sur des images sont largement utilisées en recherche et de plus en plus dans l'industrie. Ces projets emploient principalement des méthodes de deep learning afin d'obtenir des performances satisfaisantes. Cependant, les méthodes de deep learning demandent d'importantes ressources calculatoires afin d'être réalisables, ressources souvent indisponibles sur les systèmes embarqués. Des articles scientifiques commencent à voir le jour sur cette problématique que nous souhaitons aussi explorer à l'avenir.

Le choix s'est porté sur l'utilisation d'un réseau de neurones convolutifs afin de réaliser la détection des objets, méthode couramment utilisée dans le domaine de la vision par ordinateur. L'objectif étant de réaliser l'opération de détection d'objets sur le robot terrestre équipé de la caméra stéréoscopique ZED, et de la carte électronique Jetson TX2. Etant donné la capacité de calcul de la carte électronique, nous utilisons une architecture légère de réseau de neurones nommée YOLO (YOLO : You Only Look Once) et sa version encore plus légère. L'entraînement du modèle de détection est réalisé sur la base de données COCO, qui est une importante base de données pour la détection d'objets.

De cette façon, nous pouvons réaliser la détection des objets présents dans l'environnement proche de la caméra stéréoscopique ZED, embarquée sur la carte électronique Jetson TX2. Un exemple de la détection d'objet est illustré par la Figure 34, où on peut voir des boîtes englobant les objets détectés par le modèle et indiquant la distance à laquelle se situent ces objets (ce qui est un avantage de l'utilisation d'une caméra stéréoscopique, la tâche étant plus difficile avec une caméra monoculaire). Cependant la détection n'est pas précise (boîtes englobantes ne couvrant pas la totalité des objets) et la cadence d'analyse est très faible (2 images par seconde). Cette cadence d'acquisition est améliorée au détriment de la performance de détection du modèle en utilisant l'architecture plus légère de YOLO, qui permet d'atteindre une cadence d'analyse de 10 images par seconde. Bien qu'on soit encore éloigné du temps réel et pas encore couplé au réseau ROS, cette vérification de faisabilité d'intégrer la détection d'objets pour renforcer la reconstruction 3D reste prometteuse, même si nécessitant encore du travail. En particulier pour spécifier une architecture de réseau de neurones performante en temps réel, l'intégrer dans le processus de traitement de ROS, ainsi que pour intégrer les objets détectés dans la reconstruction 3D.



1.3 Projet amélioration du robot terrestre

Dernier point que nous avons abordé cette année, ce fût l'amélioration du robot terrestre et notamment le changement de la base roulante pour pouvoir supporter le poids engendré par l'ajout de matériel (support, Jetson TX2, caméra stéréoscopique, lidar, batteries).

En effet, la base roulante initiale avait été sélectionnée pour l'utilisation du lidar au travers d'une carte électronique Raspberry Pi3. Cependant, au cours de l'année 2018, nous avons remarqué que la capacité de traitement de la Raspberry était insuffisante, raison pour laquelle nous nous sommes tournés vers la carte électronique Jetson TX2.

La Jetson TX2 permettant de traiter une plus grande quantité d'information, elle nécessite par conséquent une source d'alimentation plus importante. Ce qui se traduit par l'ajout d'une batterie dédiée. L'ajout de tous ces composants diminue l'autonomie électrique du robot et rend aussi plus difficile sa mobilité, ce qui se traduit par une difficulté à manœuvrer pour le robot. La difficulté à manœuvrer de ce dernier se répercute ensuite sur la reconstruction 3D, et plus particulièrement sur la localisation du robot dans son environnement lorsque celui-ci doit effectuer des virages. Ce phénomène s'explique par l'écart entre une trajectoire courbe que le robot a planifié de réaliser, et celle qu'il réalise réellement, contraint par le poids qu'il embarque et qui l'empêche de réaliser une rotation correcte de ses roues.

Afin de remédier à ce problème nous avons changé la base roulante cette année. L'idée est de se diriger vers un prototype de base roulante de type voiture radio commandée (assemblage réalisé à partir de plusieurs composants : châssis, moteur, batterie, variateur de vitesse, roues, servo-moteur de contrôle des roues et carte électronique de commande des moteurs). Celle-ci étant suffisante pour transporter l'ensemble des composants actuellement équipés sur le robot, mais aussi pour les futurs ajouts de matériel que nous pourrions envisager (cf. Figure 35). De plus, l'autonomie de ce

robot permet d'augmenter l'autonomie à plusieurs heures de fonctionnement. Aussi des premiers tests ont été réalisés afin de fusionner l'information du lidar 2D et de la caméra stéréoscopique. Ces derniers se sont avérés concluants, ce qui préfigure de bonnes perspectives pour la suite.



Figure 38 : Photo de la nouvelle base roulante équipée de l'ensemble du matériel

1.4 Perspectives

- Transférer les données acquises par le robot terrestre mobile vers un serveur centralisant les données en utilisant le projet Docker ROS
- Reconstruire l'environnement 3D sur le serveur en utilisant les outils du projet de SLAM 3D
- Visualiser l'environnement 3D sur le serveur avec le casque de réalité virtuelle
- Comparer l'outil de SLAM 3D utilisé cette année avec d'autres outils de SLAM tels que « google cartographer » et « RTAB-Map » qui permettent la gestion de multi-agents et multi-capteurs (e.g. LiDAR et caméra stéréoscopique)
- Fusionner les données du Lidar 2D avec la caméra stéréoscopique pour améliorer la précision de la reconstruction 3D de l'environnement
- Reconstruire une carte 3D globale sur le serveur à partir de plusieurs cartes locales acquises par plusieurs robots (SLAM multi-agents)

2. Travaux 2020 : projet YOLO

Le cadre de travail YOLO (You Only Look Once) a été introduit en 2016 afin d'appuyer la performance de détection d'objets en temps réel. Depuis, il a connu plusieurs avancées qui ont été traduites par des versions labellisées : par rapport à la version originale, YOLOV2 a intégré Darknet-19 afin de modifier le réseau d'extraction des caractéristiques. Il a également utilisé la normalisation par lots, ou *batch*, lors de la phase d'apprentissage. YOLOV3 a ajouté le réseau résiduel profond Darknet-53 pour extraire des caractéristiques plus précises et donc pour réaliser des prédictions multi-scales. Ceci permet de mieux répondre aux exigences de la prédiction en temps réel en termes de temps et de précision. YOLOV4 est encore plus précis que son précédent et plus adapté au traitement vidéo. Il était basé sur l'augmentation des données mosaïques et l'amélioration de plusieurs paramètres tels que WRC, CSP et SAT [40]. La dernière version est YOLOV5. C'est un modèle pré-entraîné de plus en plus rapide. Son avantage réside dans les

multiplicateurs de mise à l'échelle de la largeur et de la profondeur du réseau. YOLO prend en input l'image complète qu'il divise en régions, par la suite, il prédit des cadres de délimitation englobant des détections candidates. Ces boîtes englobantes sont pondérées par les probabilités prédites par le réseau. Pour avoir une vision claire et non encombrante, les boîtes englobantes à score élevé sont présentées.

YOLO n'est pas uniquement efficace sur des images simples mais aussi bien sur des images complexes telles que les images 3D et les images incomplètes. Dans le cadre de notre projet, ces algorithmes sont exploitables pour avoir une meilleure perception de l'environnement en 3D peu importe les conditions de visibilité.

2.1 Détection des objets à petite échelle

Le cadre YOLOV3 a été utilisé dans [48] afin de proposer un algorithme appelé SSS-YOLO (Small Ship detection in SAR image). Cet algorithme est basé sur le réseau CNN (convolutional neural network) pour la détection de navires à petite échelle dont la forme n'est pas forcément complète. Les images qu'il exploite sont dites images SAR (Synthetic Aperture Radar) puisqu'elles sont prises par un radar à ouverture synthétique. Elles peuvent être d'une basse résolution ou d'une petite taille qu'elles paraissent juste comme un point lumineux. Dans ce cas le CNN seul engendre une perte des caractéristiques pertinentes dans l'image et dégrade considérablement la qualité de sa détection. L'idée de SSS-YOLO est donc de repenser le réseau d'extraction de caractéristiques en fonction des caractéristiques des cibles des navires dans ces images SAR. Autrement dit, il s'agit d'enrichir l'information sémantique et spatiale des petits navires. L'algorithme SSS-YOLO propose un réseau PAFN (Path Argumentation Fusion Network) permettant d'améliorer la fusion de différentes cartes de caractéristiques des images SAR (cf. Figure 39). Pour ce faire, YOLOV3 a été repensé en remplaçant les couches d'échantillon descendant de 32-stride par une Dilated-Res Unit. L'unité de résolution dilatée augmente une autre connexion de raccourci de deux couches convolutives dans différentes unités de résolution. De plus, la convolution dilatée a remplacé la convolution ordinaire. Une telle solution augmente non seulement le champ réceptif du réseau profond, mais réduit également la perte d'informations sur les petits navires causée par de multiples sous-échantillonnages, ce qui est propice à la détection de petites cibles.

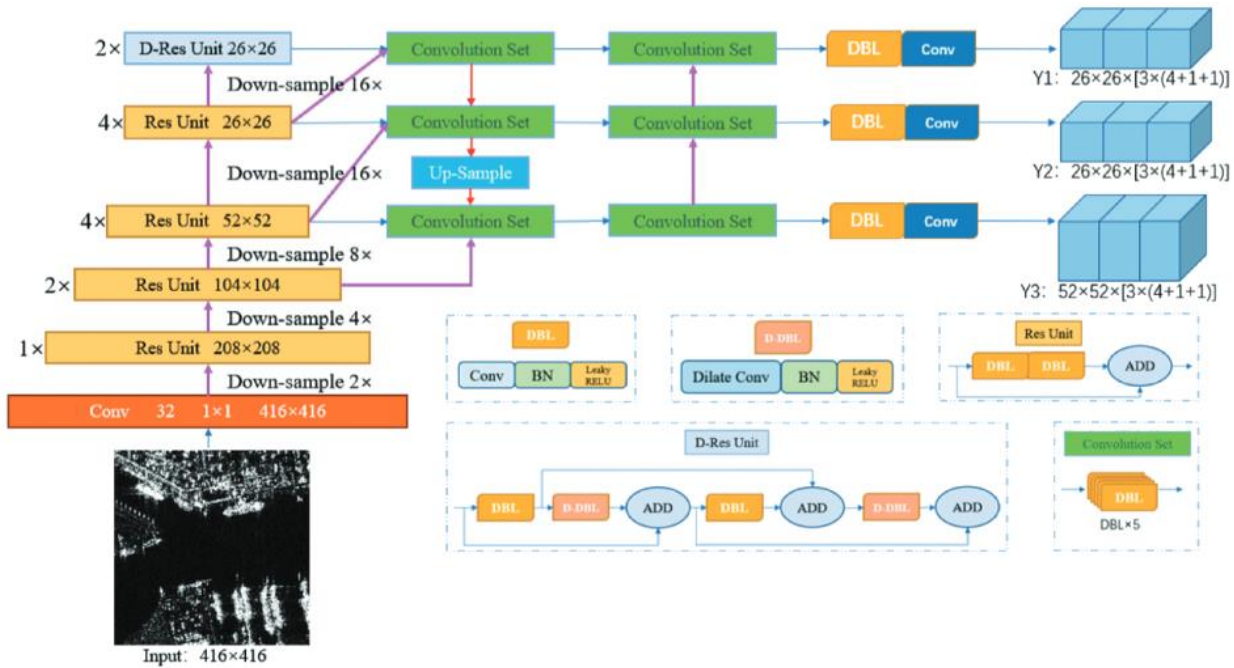


Figure 39 : Structure de SSS- YOLO-v3

Comparé à d'autres méthodes comme YOLOV3 et Faster RCNN, SSS-YOLO a prouvé une amélioration en termes d'efficacité (65.52, 62.86 et 67.24) et temps d'exécution (26.74, 93.29 et 25.84 ms respectivement).

La méthode UAV-YOLO proposée dans [44] est également basée sur YOLOV3 pour la détection des objets à petite échelle mais dans un environnement des véhicules aériens sans pilote (UAV). Cet algorithme a l'objectif d'améliorer les performances sur la détection de petits objets tout en gardant les performances inchangées sur un objet de taille normale. Le cadre YOLOv3 a été amélioré par deux aspects différents: l'entraînement du modèle et l'optimisation de la structure de base. YOLOv3 est d'abord entraîné avec l'augmentation des données par un ensemble d'images collecté selon trois catégories (normal, éloigné et jeux) en tenant compte des encombrements d'arrière-plan et de la distance entre la cible et la caméra. L'algorithme K-means est également appliqué pour optimiser le nombre et la taille de la boîte d'ancrage pendant la phase d'entraînement. Le réseau de base est optimisé pour satisfaire la détection de petits objets tout en gardant les performances inchangées dans des conditions normales. L'algorithme est basé sur l'optimisation du Resblock dans le réseau darknet en concaténant deux unités ResNet ayant la même largeur et hauteur. Ensuite, toute la structure du darknet est améliorée en augmentant l'opération de convolution à une couche précoce pour enrichir les informations spatiales. Ces deux optimisations peuvent élargir le champ réceptif. D'une manière plus technique (cf. Figure 40), l'apport du UAV-YOLO par rapport à YOLOV3 est l'optimisation du corps Res Unit sur le réseau de base Darknet afin d'élargir le champ réceptif. En effet, YOLOV3 est basé sur le bloc de structure de réseau résiduel Resblock_body noté Res_n . Res_n est composé de zéro padding, DBL et Res Unit noté aussi RU_1 . La variable n représente alors le nombre de répétitions de RU_1 . DBL est le composant de base du réseau Darknet53, Res_n indique qu'il y a n blocs résiduels. Chaque unité DBL est composée d'une couche convolutive (Conv2D), d'une couche de normalisation par lots (Batch Normalisation) et d'une fonction d'activation (Leaky ReLU). Un bloc résiduel comprend l'unité DBL, une connexion de raccourci et une opération d'addition par élément. UAV-YOLO est construit en connectant deux RU_1 afin de concaténer deux DBL dans différentes RU_1 . Le réseau de base dans UAV-YOLO est alors plus profond que celui de YOLOV3. Ceci permet d'obtenir des informations sémantiques plus

significatives et d'empêcher la diffusion de gradient en fusionnant plusieurs couches convolutives de même taille.

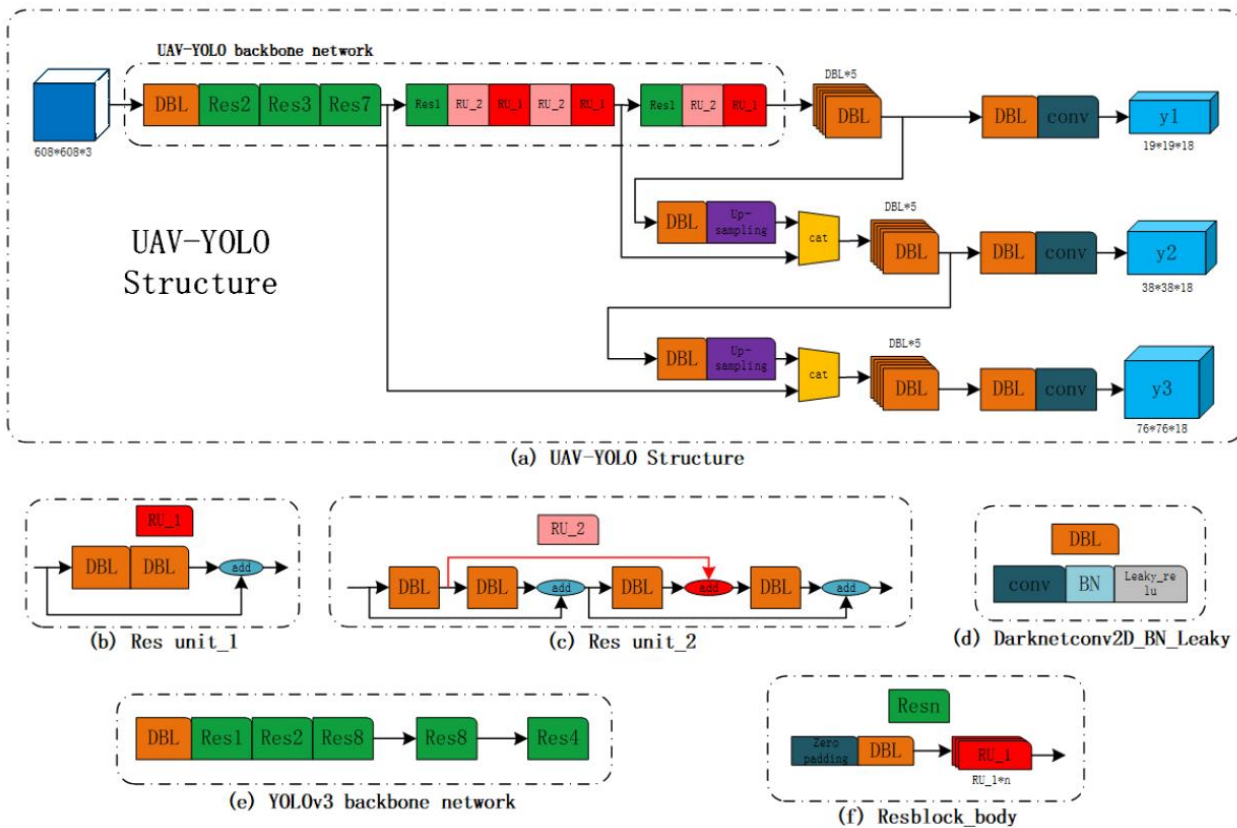


Figure 40 : Structure de UAV-YOLO

Le résultat est un champ réceptif plus large. UAV-YOLO a marqué des améliorations comparé à YOLOV3 considérant l'IOU (70.05 contre 68.43) et mAP (72.54 contre 72.21). Finalement, UAV-YOLO a consommé le même temps d'exécution que YOLOV3 mais était plus rapide que d'autres réseaux tels que SSD300 et SSD512.

2.2 Détection des objets en dimension 3D

YOLO a également été utilisé pour le traitement des images 3D.

a. Complex-YOLO

Le travail [47] a proposé une extension de la deuxième version de YOLO par une stratégie de régression complexe spécifique pour estimer des boîtes 3D multi-classes dans l'espace cartésien. Complex-Yolo est supporté par le réseau E-RPN (*Euler-Region-Proposal Network*) qui a été construit pour estimer l'orientation d'un objet en ajoutant deux fractions imaginaire et réelle pour chaque boîte. Le but est d'avoir un espace mathématique fermé sans singularités pour une généralisation précise des angles. Le modèle est capable de prédire des boîtes 3D exactes avec localisation précise des objets en temps réel, même si l'objet est basé sur quelques points.

L'architecture proposée par Complex-Yolo est présentée dans la Figure 41. Un pipeline a été utilisé pour des estimations rapides et précises de boîtes 3D sur les nuages de points. Une carte RGB est d'abord introduite dans un réseau CNN. Une grille E-RPN est ensuite exécutée simultanément sur la dernière carte des caractéristiques et prédit cinq cases pour chacune des cellules dans cette grille. Chacune des cases correspond aux paramètres de régression représentés par: x et y qui sont les coordonnées de l'entête de cet objet; l et w qui sont la longueur et la largeur de l'objet détecté et enfin les deux parties réelle et imaginaire qui définissent l'angle d'orientation de la boîte [46].

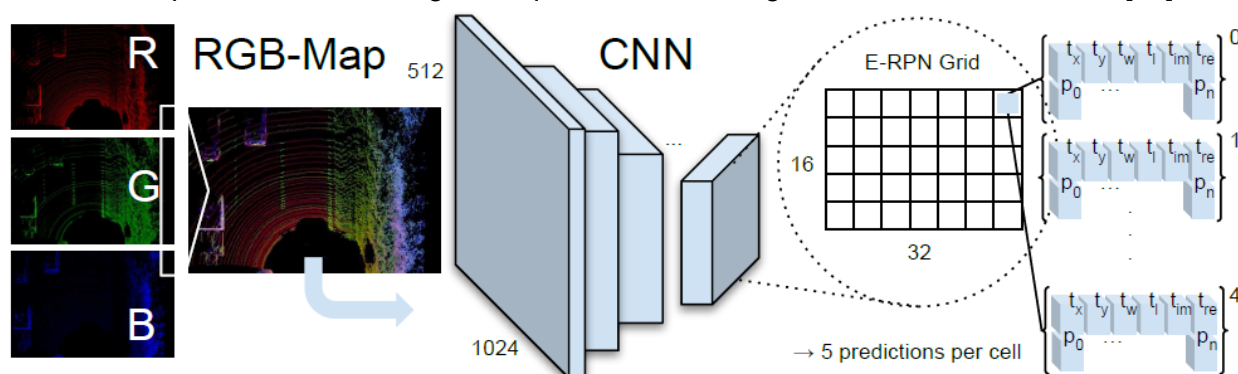


Figure 41 : Pipeline du Complex-YOLO

Chaque boîte de prédiction est composée de six paramètres de régression, de n scores de classe et d'une probabilité générale. La transition de la 2D à la 3D se fait par une hauteur prédéfinie en fonction de chaque classe.

Complex-Yolo a été testé sur trois classes: Véhicule, Piéton et Cycliste. Pour chacune des classes trois niveaux de complexité ont été définies: Simple, Moyenne et Complexe. Une étude comparative a été également menée afin de positionner Complex-Yolo par rapport à cinq autres algorithmes (cf. Tableau 1 : MV3D [41], F-PointNet [45], AVOD [42], AVOD-FPN [42] et VoxelNet [49]).

Cette étude comparative est basée sur un ensemble de 7518 images dont 75% représentent la classe Véhicule, 4% la classe Piéton et 15% la classe Cycliste. En général, la performance de Complex-Yolo n'est pas très satisfaisante.

Table 1: Tableau comparatif de la performance des méthodes de détection 3D basées sur un Lidar+Mono ou uniquement sur un Lidar

			Véhicules			Piéton			Cycliste		
Méth.	Mod.	FPS	F	M	C	F	M	C	F	M	C
MV3D	L+M	2.8	71.09	62.35	55.12	-	-	-	-	-	-
F-PointNet	L+M	5.9	81.20	70.39	62.19	51.21	44.89	40.23	71.96	56.77	50.39
AVOD	L+M	12.5	73.59	65.78	58.38	38.28	31.51	26.98	60.11	44.90	38.80
AVOD-FPN	L+M	10.0	81.94	71.88	66.38	46.35	39.00	36.58	59.97	46.12	42.36
Voxel-	L	4.3	77.47	65.11	57.73	39.48	33.69	31.51	61.22	48.36	44.37

Net											
Complex-YOLO	L	50.4	67.72	64.00	63.01	41.79	39.70	35.92	68.17	58.32	54.30

Les résultats montrent que Complex-Yolo était moins performant concernant la détection de la classe Piéton puisqu'elle représente une classe minimale. Également, l'algorithme était plus performant en ce qui concerne la classe majoritaire Véhicule. Cependant, la performance de Complex-Yolo a dépassé celle de Voxel-Net (basé uniquement sur un Lidar) en particulier sur les deux classes minoritaires. Comparé aux autres méthodes, complex-Yolo n'a pas marqué une amélioration considérable.

b. b. Modèle YOLO3D

Les auteurs dans [38] ont également utilisé la deuxième version de YOLO afin d'améliorer la performance de détection des objets 3D dans un contexte des véhicules autonomes. Un algorithme YOLO3D est défini basé sur les capteurs LiDAR pour fournir la reconstruction d'un nuage de points 3D de l'environnement. La principale caractéristique du LiDAR est sa capacité physique à percevoir la profondeur avec une grande précision. L'architecture du YOLO3D suit la méta-architecture de YOLOV2 avec une adaptation et un réglage du réseau afin de correspondre à la nature de l'entrée LiDAR. Contrairement à YOLOV2 qui est basé généralement sur cinq sorties de régression et Complex-Yolo qui en génère six, YOLO3D en comprend huit : le centre de la boîte englobante de l'objet en 3D (x, y, z), les dimensions 3D (longueur, largeur et hauteur), l'orientation dans l'espace, la confiance et l'étiquette de la classe d'objets. Deux cartes de grille à partir de la projection du nuage de points ont été créées pour chaque vue. La première carte de caractéristiques contient la hauteur maximale, où chaque valeur de cellule de grille (pixel) représente la hauteur du point le plus élevé associé à cette cellule. La deuxième carte représente la densité des points qui permettrait d'évaluer l'importance de chaque cellule.

L'extension de YOLOV2 afin de construire YOLO3D se focalise alors sur l'inclusion de l'orientation et les coordonnées 3D de la boîte englobante comme étant une tâche de régression directe. Les coordonnées 3D sont déduites à partir de la coordonnée z du centre de l'objet et la hauteur maximale de la boîte déjà calculée dans la première carte. L'orientation des boîtes englobantes est comprise entre $-\pi$ et π et normalisée entre -1 à 1. Le modèle de "Régression de l'angle de Yaw" est ensuite appliqué sur ces valeurs normalisées afin de prédire l'orientation de la boîte. Le modèle a été adapté afin de prédire directement l'orientation de la boîte englobante via un seul nombre régressé. La Figure 42 montre la grille quand elle a été étendue en 3D.

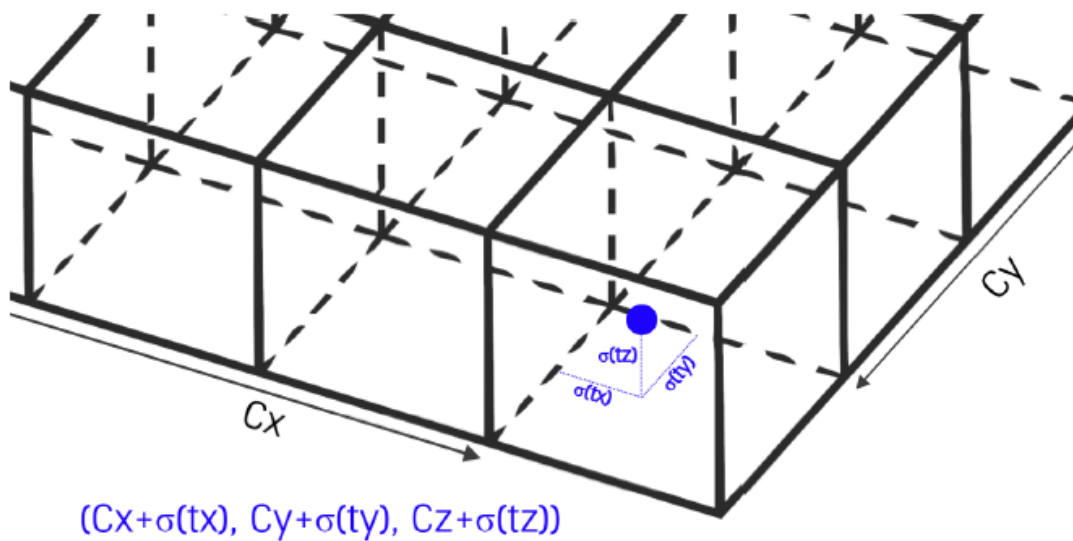


Figure 42 : Pipeline du Complex-YOLO

Alors que les coordonnées x et y sont régressées en prédisant une valeur entre 0 et 1 à chaque cellule de la grille, localisant où le point se trouve dans cette cellule, la valeur de z est uniquement mappée pour se situer dans une cellule de grille verticale puisque la variabilité de cette valeur est beaucoup plus petite que celle des x et y . Enfin, la boîte 3D pour chaque classe d'objets est dessinée en calculant les dimensions moyennes de la boîte 3D au lieu de la technique k -means. En effet, dans ce contexte où les cartes en grille sont créées de vue à vol d'oiseau, il n'y a pas de très grande variabilité dans les dimensions des boîtes au sein d'une même classe d'objets ce qui rend inefficace les méthodes de clustering. Ce modèle a été évalué sur le benchmark *KITTI* en considérant également les trois classes, Piéton, Véhicule et Cycliste. Environ 40% des images d'apprentissage ont été utilisées. Les résultats obtenus sont résumés dans le tableau 2 en considérant les deux métriques d'évaluation Rappel et précision.

Table 2: Tableau comparatif de la performance des méthodes de détection 3D basées sur un Lidar+Mono ou uniquement sur un Lidar

Label	Précision	Rappel	F-score
Piéton	44.0%	39.2%	41.46%
Cycliste	65.13%	51.1%	57.26%
Véhicule	94.07%	83.4%	88.59%

Pour avoir une visibilité plus claire sur l'efficacité du modèle nous avons calculé la F-score. Le modèle était uniquement efficace pour la classe Véhicule. Concernant les deux autres classes, les résultats n'étaient pas satisfaisants.

c. Modèle RGB-D pour une localisation 3D

Le modèle RGB-D proposé par [43] est basé sur la troisième version de YOLO et est destiné à la localisation 3D des humains. Au lieu d'un Lidar, RGB-D exploite des données issues d'une caméra

RGB-D qui fournit simultanément une image couleur et une carte de profondeur caractérisant la distance des objets vus.

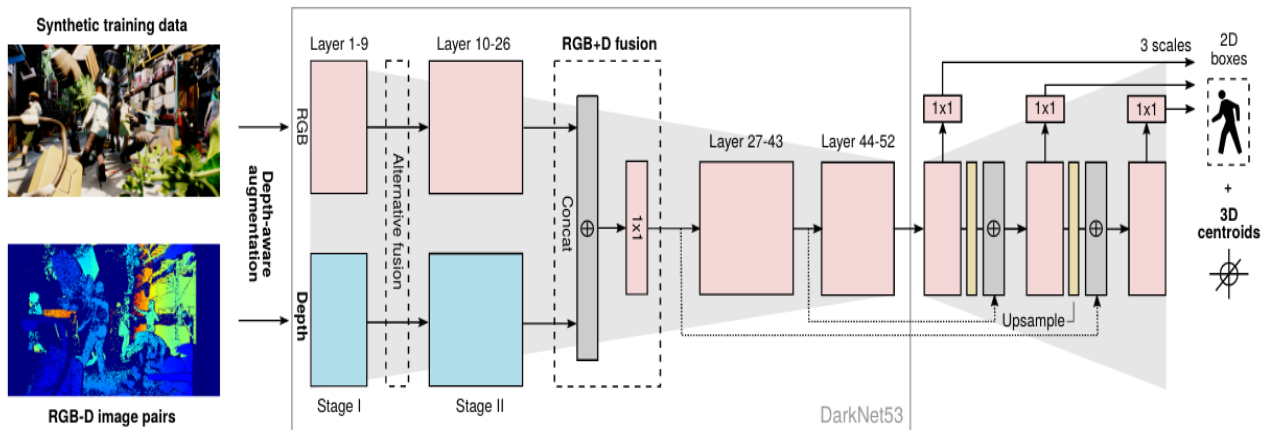


Figure 43 : Architecture du modèle RGB-D pour la reconnaissance 3D des personnes

L'architecture de ce modèle est présentée dans la Figure 43. Pendant l'entraînement, la profondeur des objets a été mise à l'échelle en utilisant une opération de zoomage [39] permettant le redimensionnement de leur recadrage en fonction de la dimension d'entrée. D'abord, le réseau Darknet-53 a été étendu pour pouvoir exploiter les informations de profondeur. Ensuite, les couches sont dupliquées jusqu'à l'obtention d'un point de fusion entre la profondeur et l'image RGB. Deux points de fusion de niveau intermédiaire différents seront évalués et placés à la fin des étapes résiduelles dans l'architecture Darknet-53 (dans ce cas après la couche 9 ou la couche 26). De plus, les modalités sont aussi fusionnées en concaténant les caractéristiques de sortie des deux backbones en utilisant une convolution 1×1 . Ceci permet de réduire de moitié le nombre de canaux par rapport à la dimension de canal d'origine. Les dernières convolutions 1×1 des trois couches de sortie sont étendues pour prédire un centroïde (c_x, c_y, c_z) pour chaque boîte englobante. La hauteur et les coordonnées du centre vont permettre une localisation 3D des personnes sur des images. Les coordonnées c_x et c_y sont d'abord prédites en coordonnées d'image, puis rétro-projetées à l'aide de c_z et de la matrice de la caméra. De plus, les cibles de régression sont reformulées par rapport aux coordonnées du coin supérieur gauche de la boîte englobante et les deux variables représentant le hauteur et la largeur en pixels. La hauteur et les coordonnées du centre vont permettre la localisation 3D des personnes.

Le modèle a été testé sur un ensemble aléatoire contenant 15000 images des personnes figurant dans vingt-trois positions différentes. Deux métriques d'évaluation ont été utilisées qui sont la précision moyenne de 3D (3D AP) et le peak F-score (Peak F-1). Chaque métrique a été testée sur deux rayons d'évaluations différents valant 0.25m et 0.5m.

Les résultats d'évaluation du modèle RGB-D sont présentés dans le tableau 3. Nous remarquons que ce modèle n'a pas surpassé le modèle naïf 3D quand il a été testé sur des données réelles ou synthétiques séparément. Il était aussi plus performant en considérant un rayon d'évaluation plus élevé. Sinon, les résultats restent satisfaisants mais pas très performants surtout que ce modèle a été testé sur une seule classe (personne) et sur des images dont la qualité a été améliorée en amont.

Table 3: Tableau comparatif du rapport rappel précision du modèle RGBD testé

sur des données réelles (R) et des données synthétiques (S)

Modèle	3D AP: 0.25m	3D AP: 0.5m	Peak-F1: 0.25m	Peak-F1: 0.25m
Naive YOLOv3 (RGB+D)	58.1	79.8	72.8	86.6
RGB, S+R	57.5	95.2	69.8	93.8
RGB-D, R	39.0	82.1	56.7	87.3
RGB-D, S	59.9	93.7	72.5	93.5
RGB-D, R+S	68.7	96.5	78.6	95.3

En résumé, en nous basant sur cet état de l'art concernant les dernières avancées sur la détection 3D, nous pouvons confirmer que les modèles existants ne sont pas encore suffisants pour ce genre d'applications. La performance se dégrade considérablement dans les cas complexes tels que les objets chevauchés, les objets partiellement incomplets ou dans des conditions d'éclairage difficiles.

Références bibliographiques

- [1] F. Danilewsky, «Réalité virtuelle : un marché de 895 millions de dollars,» 13 04 2016. [En ligne]. Available: <http://www.idboox.com/smartphone/realite-virtuelle-un-marche-de-895-millions-de-dollars/>. [Accès le 23 05 2016].
- [2] R. Loukil, «La réalité virtuelle, un marché de 30 milliards de dollars en 2020 ?,» 10 04 2015. [En ligne]. Available: <http://www.usine-digitale.fr/editorial/la-realite-virtuelle-un-marche-de-30-milliards-de-dollars-en-2020.N323666>. [Accès le 23 05 2016].
- [3] Nicolas, «Business Model – Le marché de la réalité virtuelle et augmentée,» 10 05 2016. [En ligne]. Available: <http://www.realite-virtuelle.com/business-model-rv-ra>. [Accès le 23 05 2016].
- [4] «Pourquoi le marché des drones professionnels va exploser,» 18 05 2015. [En ligne]. Available: <http://www.capital.fr/bourse/actualites/energie-transports-securite-immobilier-pourquoi-le-marche-des-drones-professionnels-va-exploser-1039150>. [Accès le 23 05 2016].
- [5] B. Arnaldi, P. Fuchs et J. Tisseau, «Chapitre 1 du volume 1 du traité de la réalité virtuelle,» chez Le traité de la réalité virtuelle, Les Presses de l'Ecole des Mines de Paris, 2003.
- [6] J.-M. Burkhardt, Réalité virtuelle et ergonomie: quelques apports réciproques, Presses Universitaires de France, 2003.
- [7] «Guide & Comparatif des casque de réalité virtuelle et augmentée,» [En ligne]. Available: <http://www.realite-virtuelle.com/guide-comparatif-casque-realite-virtuelle>. [Accès le 13 05 2016].
- [8] «Oculus Rift,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Oculus_Rift. [Accès le 13 05 2016].
- [9] J.-K. Lauret et C. Bechade, «Test / HTC Vive : une nouvelle dimension à portée de main,» 22 04 2016. [En ligne]. Available: <http://www.lesnumeriques.com/casque-realite-virtuelle/htc-vive-p30731/test.html>. [Accès le 13 05 2016].
- [10] «PlayStation VR,» [En ligne]. Available: https://fr.wikipedia.org/wiki/PlayStation_VR. [Accès le 13 05 2016].
- [11] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier et B. MacIntyre, «Recent Advances in Augmented Reality,» Computer Graphics and Applications, IEEE, vol. 21, n° 16, pp. 34-47, 2001.
- [12] R. T. Azuma, «A survey of augmented reality,» Presence: Teleoperators and virtual environments, vol. 6, n° 4, pp. 355-385, 1997.
- [13] P. Milgram et F. Kishino, «A Taxonomy of Mixed Reality Visual Displays,» IEICE Transactions on Information Systems, Vols. 77-D, n° 112, 1994.
- [14] H. Tarik, «Top des utilisations marketing de la réalité augmentée,» 19 04 2016. [En ligne]. Available: <http://www.realite-virtuelle.com/top-marketing-realite-augmentee>. [Accès le 17 05 2016].
- [15] J. Bergounhoux, «Réalité mixte : la Nasa réduit ses risques et ses coûts avec HoloLens,» 17 02 2016. [En ligne]. Available: <http://www.usine->

digitale.fr/article/realite-mixte-la-nasa-reduit-ses-risques-et-ses-couts-avec-hololens.N379958. [Accès le 02 06 2016].

[16] «Google Glass,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Google_Glass. [Accès le 13 05 2016].

[17] «Microsoft HoloLens,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Microsoft_HoloLens. [Accès le 13 05 2016].

[18] M. Mercky, «BAE Systems : la réalité augmentée pour les soldats de demain,» 08 03 2014. [En ligne]. Available: <http://www.generation-nt.com/bae-systems-realite-augmentee-pour-soldats-demain-actualite-1859152.html>. [Accès le 13 05 2016].

[19] L. Barnier, «ATR sélectionne officiellement la ClearVision d'Elbit,» 06 07 2015. [En ligne]. Available: <http://www.air-cosmos.com/atr-selectionne-officiellement-la-clearvision-d-elbit-38920>. [Accès le 13 05 2016].

[20] P. Axelsson, «Processing of laser scanner data-algorithms and applications,» ISPRS Journal of Photogrammetry and Remote Sensing, vol. 54, n° 12, pp. 138-147, 1999.

[21] «Scanner tridimensionnel,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Scanner_tridimensionnel#Scanner_par_temps_de_vol. [Accès le 19 05 2016].

[22] M. Dujardin, «Le scanner laser 3D: reconnaissance de formes et modélisation de déformations,» 2013.

[23] O. Wulf et B. Wagner, «Fast 3D scanning methods for laser measurement systems,» International conference on control systems and computer science, pp. 2-5, 2003.

[24] L. Nuninger, C. Fruchart et R. Opitz, «LiDAR : quel apport pour l'analyse des paysages ?,» Bulletin AGER, pp. 34-43, 2010.

[25] L. Benjamin, «Automatisation de l'acquisition et des post-traitements en numérisation 3D,» 2009.

[26] D. Bechmann, J.-M. Dischler et M. Fournier, «Filtrage adaptatif des données acquises par un scanner 3D et représentées par une transformée en distance volumétrique,» Journées AFIG, 2006.

[27] P. Maur, «Delaunay triangulation in 3D,» 2002.

[28] J. Salvi, C. Matabosch, D. Fofi et J. Forest, «A review of recent range image registration methods with accuracy evaluation,» Image and Vision computing, vol. 25, n° 15, pp. 578-596, 2007.

[29] J. R. Shewchuk, «Delaunay refinement algorithms for triangular mesh generation,» Computational geometry, vol. 22, n° 11, pp. 21-74, 2002.

[30] «Triangulation de Delaunay,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Triangulation_de_Delaunay.

- [31] C. Harvey E et W. E. Lorensen, «Marching cubes: A high resolution 3D surface construction algorithm,» ACM siggraph computer graphics, vol. 21, n° %14, pp. 163-169, 1987.
- [32] «Marching cubes,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Marching_cubes.
- [33] L. Tian et F. Chen, «Quantitative analysis of molecular surface based on improved Marching Tetrahedra algorithm,» Journal of Molecular Graphics and Modelling, vol. 38, pp. 314-323, 2012.
- [34] «Marching tetrahedra,» [En ligne]. Available: https://fr.wikipedia.org/wiki/Marching_tetrahedra.
- [35] D. Christophe, «Contributions pour les applications de réalité augmentée. Suivi visuel et recalage 2D. Suivi d'objets 3D représentés par des modèles par points,» 2008.
- [36] D. Roussel et M. Mallem, Réalité augmentée: principes, technologies et applications, Techniques de l'ingénieur. Télécoms TE5920, 2008.
- [37] C. Qi, «Airborne lidar data processing and information extraction,» Photogrammetric engineering and remote sensing, vol. 73, n° %12, p. 109, 2007.
- [38] Waleed Ali, Sherif Abdelkarim, Mahmoud Zidan, Mohamed Zahran, and Ahmad El Sallab. Yolo3d: End-to-end real-time 3d oriented object bounding box detection from lidar point cloud. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, pages 0–0, 2018.
- [39] Alex M Andrew. Multiple view geometry in computer vision, by richard hartley and andrew zisserman, cambridge university press, cambridge, 2000, xvi+ 607 pp., isbn 0-521-62304-9 (hardback,£ 60.00). Cambridge University Press, 2001.
- [40] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934, 2020.
- [41] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 1907–1915, 2017.
- [42] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven L Waslander. Joint 3d proposal generation and object detection from view aggregation. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 1–8. IEEE, 2018.
- [43] Timm Linder, Kilian Y Pfeiffer, Narunas Vaskevicius, Robert Schirmer, and Kai O Arras. Accurate detection and 3d localization of humans using a novel yolo-based rgb-d fusion approach and synthetic training data. In 2020 IEEE International Conference on Robotics and Automation (ICRA), pages 1000–1006. IEEE, 2020.
- [44] Mingjie Liu, Xianhao Wang, Anjian Zhou, Xiuyuan Fu, Yiwei Ma, and Changhao Piao. Uav-yolo: Small object detection on unmanned aerial vehicle perspective. Sensors, 20(8):2238, 2020.

- [45] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 918–927, 2018.
- [46] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7263–7271, 2017.
- [47] Martin Simony, Stefan Milzy, Karl Amendey, and Horst-Michael Gross. Complex-yolo: An euler-region-proposal for real-time 3d object detection on point clouds. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, pages 0–0, 2018.
- [48] Jingpu Wang, Youquan Lin, Jie Guo, and Long Zhuang. Sss-yolo: towards more accurate detection for small ships in sar image. Remote Sensing Letters, 12(2):122–131, 2021.
- [49] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4490–4499, 2018.