Vers l'immersion mobile en réalité augmentée : une approche basée sur le suivi robuste de cibles naturelles et sur l'interaction 3D

Abdelkader Bellarbi

NNT : 2017SACLE005

THESE DE DOCTORAT
DE
L'UNIVERSITE PARIS-SACLAY
PREPAREE A
L'UNIVERSITE EVRY VAL D'ESSONNES

ÉCOLE DOCTORALE N°580
Sciences et technologies de l'information et de la communication (STIC)

Spécialité de doctorat : Informatique

Par

M. Abdelkader Bellarbi

Vers l'immersion mobile en réalité augmentée : une approche basée sur le suivi robuste de cibles naturelles et sur l'interaction 3D

Thèse présentée et soutenue à Evry, le 26/04/2017 :

Composition du Jury :

M. Guillaume Moreau, Professeur, Ecole Centrale de Nantes, Président.

M. Peter Sturm, Directeur de recherche, INRIA Grenoble Rhône-Alpes, Rapporteur.

Mme Luce Morin, Professeur, INSA de Rennes, Rapporteur.

Mme Indira Thouvenin, Enseignant-chercheur -HDR, UTC, Compiègne, Rapporteur.

M. Samir Otmane, Professeur, Université d'Evry Val d'Essonne, Directeur de thèse

Mme Nadia Zenati, Maitre de recherche A, CDTA, Algérie, Co-directrice de thèse.

Remerciement

Cette thèse est le fruit de la collaboration entre deux institutions qui ont permis son accomplissement. Je souhaite donc remercier, l'UEVE (Université d'Evry Val d'Essonne) d'une part pour m'avoir permis d'effectuer cette thèse et le CDTA (Centre de Développement des Technologies Avancés) d'Alger d'autre part, pour m'avoir offert le cadre de travail nécessaire.

Je remercie très chaleureusement mon directeur de thèse, monsieur le professeur Samir Otmane, qui, malgré ses nombreuses occupations, a accepté de prendre la direction de cette thèse, transformant ainsi les difficultés rencontrées en une expérience enrichissante. Je lui suis également reconnaissant de m'avoir accordé généreusement le temps nécessaire pour partager avec moi sa grande expérience et de m'avoir assuré un encadrement rigoureux, tout en me donnant toutefois la possibilité de trouver par moi-même mon cheminement personnel.

Mes sincères remerciements s'adressent également à ma co-directrice de thèse, madame Nadia Zenati, qui m'a toujours soutenu et encouragé. Son écoute, ses critiques et ses conseils constructifs m'ont guidé tout au long de cette thèse. Je tiens à saluer son dynamisme et son énergie qui ont déteint sur moi durant ces quelques années de travail ensemble. Je tiens à vous remercier madame particulièrement pour votre générosité et pour tous le soutient que vous m'avez apporté.

J'exprime tous mes remerciements à l'ensemble des membres de mon jury : Madame Luce Morin, Madame Indira Thouvenin, Monsieur Peter Sturm et Monsieur Guillaume Moreau. Je vous suis reconnaissant pour avoir accepté de rapporter et d'examiner ce travail.

J'adresse ma gratitude à toute l'équipe IRVA (Interaction en Réalité Virtuelle et Augmentée) du CDTA à commencer par Samir Benbelkacem collègue et ami, que je remercie pour ses conseils, ses encouragements et pour avoir été présent pour moi. Un merci à mon ami d'enfance et collègue Mahfoud, à Assia, ainsi que tous les autres membres, qu'ils me pardonnent de ne pas tous les avoir expressément nommés, merci pour votre soutien et vos encouragements, et surtout, merci pour m'avoir supporté dans les moments de stresse. Je remercie également le personnel technique et administratif que j'ai côtoyé pendant ces quelques années et qui a contribué à alléger mes charges en rapport direct ou indirect avec cette thèse.

De plus, comment ne pas citer aussi les membres de l'équipe IRA2 (Interaction, Réalité Augmentée et Robotique Ambiante) du laboratoire IBISC. Merci à Amine Chellali qui a été un ami tout au long de cette période, merci pour tes conseils et tes orientations, ton aide m'a été précieuse. Je cite aussi, Jean-Yves Didier et Frédéric Davesne, merci pour votre disponibilité, votre

accueil et votre sympathie. Un merci à l'ensemble des doctorants pour l'ensemble des moments partagés.

J'adresse ma chaleureuse reconnaissance à une personne qui est à la fois mon amie, ma collègue et ma femme. Qui m'a accompagné et supporté durant toute cette aventure. Je te remercie infiniment Hayet pour ta patience et ton soutien sans failles. Ainsi que toute ta formidable famille.

Enfin, les mots les plus simples étant les plus forts, j'adresse toute mon affection à ma famille, je vous remercie pour votre compréhension, pour le support et l'aide que chacun de vous m'a apporté à sa manière. Un merci particulier à mes parents : maman, papa vous êtes pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple du dévouement, rien au monde ne vaut les efforts fournis jour et nuit pour mon éducation et mon bien être.

Pour finir, merci à toute personne qui a participé de près ou de loin à l'accomplissement de ce travail.

Sommaire

Liste des figures

Liste des tables

Glossaire

AGAST	Adaptive and Generic Accelerated Segment Test
BRIEF	Binary Robust Independent Elementary Features
BRISK	Binary Robust Invariant Scalable Keypoints
DLT	Direct Linear Transformation
DoG	Difference of Gaussian
ESM	Efficient Second order Minimisation
EV	Environnement Virtuel
FAST	Fast Accelerated Segment Test
FOV	Field Of View
FREAK	Fast Retina Key-point
HOMER	Hand-centered Object Manipulation Extending Ray-casting
KLT	Kanade & Lucas Tracket
LDA	Linear Discriminant Analysis
LDE	Local Discriminant Embedding
MOBIL	Moments-based BInary differences for Local description
NUI	Natural User Interface
PCA (ACP)	Principal Component Analysis
P-nP	Perspective n Point
POSIT	Pose From Orthography And Scaling With Iterations
PTAM	Parallel tracking And Mapping
RA	Réalité Augmentée
RV	Réalité Virtuelle
SIFT	Scale-invariant feature transform
SLAM	Simultaneous Localization And Mapping
SSD	Sum of Squared Differences
SURF	Speeded Up Robusts Features
SVD	Singular Value Decomposition

Introduction et Problématique

1. Contexte de la thèse

Cette thèse s'inscrit dans le cadre de la collaboration entre l'Université d'Evry-Val-d'Essonne (UEVE) et le Centre de Développement des Technologies Avancées (CDTA) d'Alger en Algérie. En effet, les travaux ont été menés conjointement entre deux équipes de recherches : Interaction, Réalité Augmentée & Robotique Ambiante (IRA2) du laboratoire IBISC de l'UEVE et Interaction en Réalité Virtuelle & Augmentée (IRVA) de la Division Productique et Robotique (DPR) du CDTA.

Ces travaux concernent principalement le domaine de la réalité augmentée (RA) dont les problématiques se situent au croisement de plusieurs disciplines notamment : le traitement d'images, la vision par ordinateur et l'interaction homme machine. En effet, la réalité augmentée regroupe un ensemble de techniques et technologies permettant d'associer un monde réel avec un monde virtuel. L'objectif est de parvenir à rompre la frontière entre l'ordinateur et le monde réel afin de permettre à l'utilisateur de bénéficier de moyens informatiques, tout en restant au contact de son environnement réel.

Du point de vue scientifique, la RA se base sur un ensemble de méthodes de traitements informatiques notamment en vision par ordinateur et en interaction homme machine, afin d'assurer la cohérence spatio-temporelle des deux mondes (réel et virtuel). Il s'agit également de fournir à l'utilisateur des moyens lui permettant d'interagir de manière spontanée dans son environnement augmenté.

D'un point de vue technologique, la RA a beaucoup évolué. Depuis le premier casque proposé par Sutherland en 1968 jusqu'à aujourd'hui, nous avons pu voir une multitude de dispositifs de visualisation et/ou d'interaction évoluer : casques, tablettes, smartphones, kinect, Leap motion…

Ces avancées technologiques ont permis l'émergence d'un nouveau concept, celui de « l'immersion mobile en réalité augmentée ». Ce dernier consiste à rendre possible l'immersion de l'utilisateur dans un environnement mixte réel/virtuel comme s'il s'agissait d'un « seul monde » en lui permettant de se déplacer et d'interagir en toute liberté dans son environnement augmenté tout en conservant une cohérence sensorielle (visuelle, auditive et haptique). L'immersion mobile en réalité augmentée combine donc les technologies de la réalité augmentée

et celles des interfaces homme-machine en utilisant des dispositifs mobiles de plus en plus puissants et légers. Ainsi, au fur et à mesure que la technologie évolue, de même que les techniques de traitement d'images et de vision par ordinateur, les systèmes de RA deviennent de plus en plus mobiles et à la fois immersifs en exploitant les capacités sensorielles de l'utilisateur. Les interactions seront alors naturelles et multimodales (impliquant l'audio, le visuel et l'haptique) et les augmentations deviendront omniprésentes.

Actuellement, des limitations subsistent encore que ce soit sur le plan, technologique, dans le domaine de la vision par ordinateur ou encore au niveau des techniques et dispositifs d'interaction.

Dans cette thèse, nous nous intéressons plus particulièrement aux questions suivantes :

Comment peut-on immerger un utilisateur dans un monde de réalité augmentée tout en garantissant sa mobilité ?

Comment maintenir la cohérence visuelle dans l'espace et dans le temps entre l'action d'un utilisateur vers le système (en entrée) et la réaction du système vers l'utilisateur (en sortie) dans un environnement augmenté ?

Pour répondre à ces questions, il faut s'intéresser d'une part à la chaine du système de vision pour la réalité augmentée à savoir la **détection** et la **description** des caractéristiques visuelles de l'image, l'estimation de pose. D'autre part, au processus d'interaction notamment aux techniques **de sélection et de manipulation** des objets virtuels.

Dans cette optique, l'objectif de mes travaux de thèse est de concevoir un système de réalité augmentée mobile permettant d'une part, un suivi stable et précis de la scène réelle, afin d'assurer une augmentation cohérente et réaliste. Et d'autre part, une interaction naturelle et intuitive avec les objets virtuels insérés.

2. Organisation du mémoire

Ce mémoire est organisé en deux parties. Une première partie « état de l'art », composée de deux chapitres, est consacrée à la recherche de concepts, méthodes et outils pouvant répondre à notre problématique. La seconde partie « contributions » est composée également de deux chapitres, présente les contributions de cette thèse.

Le premier chapitre présente les techniques de vision pour la réalité augmentée. Durant ce chapitre, nous commençons par introduire les éléments de base qui forment un système de vision par ordinateur pour la réalité augmentée. Nous présentons aussi, les différentes techniques et méthodes existantes pour l'estimation de pose, à savoir les techniques se basant sur un modèle

3D et les techniques se basant sur des informations coplanaires. Nous abordons également, les différentes approches de reconnaissance et de suivi d'objets ou de cibles naturelles en particulier les techniques de détection et de description des caractéristiques de l'image.

Le second chapitre porte sur l'interaction naturelle de l'utilisateur avec les objets virtuels insérés. Nous définissons donc les concepts liés à l'interaction (techniques, paradigme et métaphore d'interaction) ainsi que les modalités d'interaction naturelle et les différentes tâches d'interaction. Nous abordons aussi la classification des techniques d'interaction 3D en fonction des tâches d'interaction à savoir les techniques de sélection, les techniques de manipulation et les techniques de contrôle d'application. Nous présentons aussi les techniques et technologies existantes d'acquisition et de reconnaissance de gestes de la main pour l'interaction en RA.

A la fin de cette première partie du mémoire, nous présentons un bilan qui résume les différentes approches, techniques et travaux présentés durant les deux premiers chapitres et nous permet ainsi d'aborder au mieux la seconde partie de ce mémoire qui concerne nos contributions.

Le troisième chapitre présente notre contribution dans la partie vision pour la RA. En effet, nous commençons par présenter notre nouveau détecteur de points d'intérêt. Ce dernier hybride le détecteur AGAST avec la mesure de coins de Shi-Tomasi. Des tests et des comparaisons du détecteur proposé avec d'autres détecteurs connus sont également présentés. Nous introduisons par la suite, notre nouveau descripteur binaire appelé MOBIL (Moments-based BInary differences for Local description) (Bellarbi et al. 2014b). Ce dernier se base sur la comparaison binaire des moments géométriques autour du point d'intérêt (le patch), et présente une robustesse contre les différents types de transformations et changement d'intensité afin d'assurer une augmentation stable et précise des scènes réelles.

Nous présentons par la suite deux améliorations apportées au descripteur MOBIL dans le but d'améliorer sa robustesse. A savoir MOBIL_2B et POLAR_MOBIL. MOBIL_2B (Bellarbi et al. 2015) affecte pour chaque test binaire deux bits au lieu d'un seul afin d'améliorer la distinction entre les patchs. De son côté, POLAR_MOBIL (Bellarbi et al. 2017a) applique les tests binaires sur l'image polaire du patch afin d'améliorer l'invariance du descripteur contre les transformations affines et le changement de point de vue. Des tests des descripteurs proposés et des comparaisons avec les descripteurs de l'état de l'art sont présentés par la suite.

Ces descripteurs sont utilisés durant la deuxième partie de ce chapitre afin de mettre en œuvre un système de réalité augmentée mobile. En effet, nous avons utilisé une technique qui se base sur le descripteur POLAR_MOBIL afin d'assurer le recalage réel/virtuel, en offrant ainsi à l'utilisateur la possibilité de se déplacer tout en restant immergé dans son environnement augmenté.

Le quatrième chapitre présente notre contribution dans le domaine de l'interaction 3D en RA en introduisant une nouvelle technique d'interaction « Zoom-In » (Bellarbi et al. 2017b). Cette technique combine la technique de manipulation « Virtual Hand » et le zoom de la caméra afin de répondre aux problèmes de sélection et de manipulation des objets virtuels distants dans un environnement augmenté. Ainsi, cette technique permet de répondre autrement au problème de mobilité en conservant la cohérence spatio-temporelle en respectant les contraintes de recalage de ces objets (Bellarbi et al. 2017c). Une étude expérimentale est menée à la fin de ce chapitre, afin d'évaluer notre technique d'interaction.

Une conclusion générale viendra clôturer cette thèse avec une mise au point sur ce qui a été présenté dans ce rapport et des perspectives à envisager comme suite de ce travail.

3. Liste des publications

La section suivante regroupe l'ensemble de mes travaux effectués durant cette thèse.

- **Abdelkader Bellarbi**, Nadia Zenati, Samir Otmane and Hayet Belghit, "Learning Moment-based Fast Local Binary Descriptor", **Journal of Electronic Imaging** 26, no. 2 (2017): 02300601-02300611.

- **Abdelkader Bellarbi**, Nadia Zenati, Samir Otmane and Hayet Belghit, "Design and Evaluation of Zoom-based 3D Interaction Technique for Augmented Reality", **ACM VRIC 2017**.

- **Abdelkader Bellarbi,** Nadia Zenati-Henda, Hayet Belghit, Mahfoud Hamidia, Samir Benbelkacem, and Samir Otmane. "An improved MOBIL descriptor for markerless augmented reality." In Control, Engineering & Information Technology (**CEIT'2015**), 2015 3rd International Conference on, pp. 1-5. IEEE, 2015.

- **Abdelkader Bellarbi**, Samir. Benbelkacem, Hayet Belghit, Nadia Zenati, Samir Otmane, "Design and Evaluation of a low cost interaction marker based technique". The 4th International Conference on Image Processing Theory, Tools and Applications **IPTA'2014**, Paris, France. pp.393—397.

- **Abdelkader Bellarbi**, Samir Otmane, Nadia Zenati, Samir BENBELKACEM. MOBIL: A Moment Based Local Binary Descriptor. The IEEE International Symposium on Mixed and Augmented Reality (**ISMAR'2014**), 10-12 September 2014, Munich. p. 251-252.

- **Abdelkader Bellarbi,** Christophe Domingues, Samir Otmane, Samir Benbelkacem and Alain Dinis, *"Augmented Reality for Underwater activities with the use of the DOLPHYN",* the 2013 IEEE International Conference on Networking, Sensing and Control **ICNSC'2013**, Paris, France.

- **Abdelkader Bellarbi,** Hayet BELGHIT, Samir BENBELKACEM, Nadia ZENATI-HENDA, Mahmoud BELHOCINE*, "Hand Gesture Recognition Using Contour based Method for Tabletop Surfaces",* the 2013 IEEE International Conference on Networking, Sensing and Control **ICNSC'2013**, Paris, France.

Autres publications comme co-auteur :

- Assia Messaci, Nadia Zenati, **Abdelkader Bellarbi**, and Mahmoud Belhocine. "3D interaction techniques using gestures recognition in virtual environment." In 2015 4th International Conference on Electrical Engineering (ICEE), pp. 1-5. IEEE, 2015.

- Hayet Belghit, **Abdelkader Bellarbi**, Samir BENBELKACEM, Nadia ZENATI, Samir OTMANE, "Vision-based Collaborative & Mobile Augmented Reality", In ACM VRIC, Laval, France, pp 23-26. 2015.

- Nadia Zenati, Mahfoud Hamidia, **Abdelkader Bellarbi**, and Samir Benbelkacem. "E-maintenance for photovoltaic power system in Algeria." In Industrial Technology (ICIT), 2015 IEEE International Conference on, pp. 2594-2599. IEEE, 2015.

- Nadia Zenati, **Abdelkader Bellarbi**, Samir Benbelkacem, and Mahmoud Belhocine. "Augmented reality system based on hand gestures for remote maintenance." In Multimedia Computing and Systems (ICMCS), 2014 International Conference on, pp. 5-8. IEEE, 2014.

- Samir Benbelkacem, Mahmoud Belhocine, **Abdelkader Bellarbi**, Nadia Zenati-Henda, and Mohamed Tadjine. "Augmented reality for photovoltaic pumping systems maintenance tasks." Renewable energy 55 (2013): 428-437.

- Nadia Zenati, Samir Benbelkacem, Mahmoud Belhocine, **Abdelkader Bellarbi,** "A new AR Interaction for Collaborative E-maintenance System." 7th IFAC Conference on Manufacturing Modelling, Management, and Control (2013), pp 619-624.

PARTIE I. Etat de l'art

Techniques d'estimation de pose et d'interaction 3D en réalité augmentée

Chapitre I.

Estimation de pose pour la réalité augmentée

1.1 Introduction

Ce présent chapitre a pour but de traiter un des problèmes de base de la réalité augmentée qui est l'estimation de la pose. En effet, la réalité augmentée (RA) étant définie comme le fait de rehausser l'environnement réel par des éléments virtuels en laissant l'utilisateur en contact avec son environnement réel. Elle permet d'insérer harmonieusement des objets virtuels dans une séquence d'images.

L'alignement des deux mondes réel et virtuel est le plus souvent nécessaire pour que l'environnement augmenté soit cohérent et pour qu'il ait un sens. La solution à ce problème est donc liée à une estimation de la pose, autrement dit à un procédé de localisation de la caméra.

Il est clair que cette problématique a suscité beaucoup d'intérêt auprès de la communauté scientifique. Pour y remédier, de nombreux types de capteurs ont été considérés : mécaniques, ultrasons, magnétiques, inertiels, GPS, boussole, gyroscope, accéléromètre, sans oublier les capteurs optiques.

Néanmoins, la caméra reste le capteur le plus attrayant pour la RA car l'estimation de la pose en utilisant uniquement la caméra simplifie la procédure d'augmentation contrairement à l'utilisation d'un autre capteur qui nécessite forcement une procédure supplémentaire de calibrage pour déterminer la pose de la caméra. De ce fait, les techniques basées vision sont privilégiées dans ce domaine.

Dans ce sens, ce chapitre nous permet de faire un tour d'horizon des techniques existantes qui permettent l'estimation de pose. Ainsi, durant ce chapitre nous allons tout d'abord aborder les concepts et définitions de base de la réalité augmentée, ainsi que le problème de localisation que nous décrivons d'un point de vue mathématique, ensuite, nous présentons les approches basées vision les plus importantes qui permettent la localisation de l'utilisateur dans son environnement augmenté.

1.2 Réalité augmentée : définition et concepts

Le terme «Augmented Reality» (en français : réalité augmentée) a été utilisé pour la première fois en 1992, par les deux chercheurs de Boeing, Thomas Caudell et David Mizell (Caudell & Mizell 1992) pour décrire un casque semi-transparent, utilisé par les électriciens d'aéronautique et qui visualise des informations virtuelles sur l'image réelle.

Etymologiquement, Le dictionnaire Petit Robert[1] donne la définition suivante :

- **Réalité** n. f. bas lat. realitas->rien. Caractère de ce qui est réel, de ce qui ne constitue pas seulement un concept, mais une chose, un fait. ... . Ce qui est réel, actuel, donnée comme tel à l'esprit.
- **Augmentée** vr tr. Augmenter. Rendre plus grand, plus considérable, par addition d'une chose de même nature.

Les premières définitions de la réalité augmentée (RA) étaient restreintes à l'utilisation de dispositifs de visualisation tête-haute semi-transparents (See-through HMD). Milgram et Kishino ont introduit en 1994 une définition plus étendue de la réalité augmentée (RA) : « *la réalité augmentée a pour but d'augmenter la rétroaction naturelle de l'opérateur avec le monde réel à l'aide d'indices virtuels*» (Milgram & Kishino 1994). Cette définition a permis de sortir du cadre de l'utilisation des seuls dispositifs de visualisation tête-haute.

Mackay (Mackay 1998) précise qu'il est possible de considérer la réalité augmentée comme une interface entre l'homme et la machine. Celle-ci permet d'interagir de manière naturelle avec le monde réel tout en bénéficiant des capacités évoluées qu'offre l'ordinateur. A partir de là, comme pour tout IHM, un système de RA doit prendre en compte les paramètres liés à l'utilisateur et à son environnement.

Ronald Azuma a développé une définition de la RA en 1997 (Azuma 1997), qu'il a complété dans son état de l'art en 2001. Selon lui « *un système de réalité augmentée est un système qui complète le monde réel avec des objets virtuels (générés par ordinateur) de telle sorte qu'ils semblent coexister dans le même espace que le monde réel »* (Azuma et al. 2001). Ainsi, il a défini trois règles de base d'un système de RA :

1. Combiner des objets réels et virtuels dans un environnement réel,
2. Être interactifs en temps-réel,
3. Recaler (aligner) les objets réels et virtuels.

[1] Petit Robert, http://www.lerobert.com/ 2016.

Mais toujours selon Azuma, la RA peut potentiellement s'appliquer à tous les sens tel que le toucher ou l'ouïe.

Cette définition a été critiquée par certains chercheurs tels que Didier (Didier 2005) ou Hugues (Hugues 2011). En effet, selon Didier (Didier 2005), cette définition serait très restrictive, vu qu'elle exclue la réalité augmentée en post-production, et considère le recalage (réel/virtuel) comme condition nécessaire pour la RA. Ainsi, d'après Didier, il suffit qu'un objet virtuel soit sémantiquement lié avec la scène réelle (sans alignement) pour le considérer comme une augmentation et donc la troisième condition de Azuma (Azuma et al. 2001) serait à éliminer.

Fuchs et Moreau (Fuchs & Moreau 2006) ont présenté une définition de la réalité augmentée plus élargie : « *La réalité augmentée regroupe l'ensemble des techniques permettant d'associer un monde réel avec un monde virtuel, spécialement en utilisant l'intégration d'Images Réelles (IR) avec des Entités Virtuelles (EV) : images de synthèse, objets virtuels, textes, symboles, schémas, graphiques, etc. D'autres types d'association entre mondes réels et virtuels sont possibles par le son ou par le retour d'effort* ». Cette définition semble la plus équilibrée.

Selon Dubois (Dubois 2009), la RA est un paradigme d'interaction qui est né de la volonté de fusionner les capacités de traitement informatique et l'environnement physique. L'objectif est de parvenir à rompre la frontière entre l'ordinateur et le monde réel afin de permettre à l'utilisateur de bénéficier de moyens informatiques, tout en restant au contact de son environnement réel. La RA vise donc à faire sortir l'ordinateur de son cadre d'utilisation habituel (écran, clavier et souris), pour laisser une part de plus en plus grande à l'interaction avec l'environnement physique de l'utilisateur, c'est-à-dire le monde réel.

Otmane (Otmane 2010) a défini la RA par sa finalité « *La finalité de la réalité augmentée est de permettre à une personne (ou plusieurs) des interactions multi sensorielles (audio, vidéo et haptique) avec un environnement qui fait coexister les deux mondes virtuel et réel* ».

Quand à Hugues (Hugues 2011) il remet complétement en question l'appellation « Réalité Augmentée ». Selon lui « *le terme est un non-sens puisque, à strictement parlé, cela paraît délicat d'augmenter … la réalité vue qu'elle représente tout ce qui existe* ». Hugues parle alors de « Perception Augmentée ». Toutefois, l'expression « Réalité Augmentée » s'est vue démocratisée auprès du grand public, il est donc inutile de remettre en cause cette appellation.

Face aux avancées des technologies de l'interaction, la RA constitue un sujet d'étude d'importance dans le domaine de l'Interaction Homme-Machine, elle permet d'accroître les capacités de l'utilisateur à percevoir les informations, à exécuter des tâches, ou encore à communiquer en privilégiant les moyens d'interaction naturels, le tout sans coupure avec son environnement habituel.

D'une manière générale, la réalité augmentée représente tout système qui permet la fusion du réel et du virtuel en offrant à l'utilisateur une perception augmentée de la réalité. Celle-ci peut s'étendre à tous les sens de l'utilisateur (le visuel, le toucher ou encore l'ouïe). En fonction de la nature de l'association du réel et du virtuel (Fuchs & Moreau 2003), cette fusion peut lier sémantiquement le réel et le virtuel et/ou prendre en compte le recalage des deux mondes réel et virtuel. Aussi avec les technologies émergentes, différents dispositifs peuvent être envisagés pour, l'estimation de pose, la visualisation (tablettes, casques,…) ou encore l'interaction. Ainsi, tous ces choix potentiels ont un impact certain sur l'immersion de l'utilisateur dans son environnement augmenté.

1.3 Quelques applications de la réalité augmentée

Les systèmes de RA ont beaucoup évolué durant cette dernière décennie avec l'avancement technologique et l'évolution des algorithmes de traitement d'images qui répondent jusqu'à un certain point aux besoins de ce genre d'applications.

De par son originalité et sa pluridisciplinarité en matière de technologie, la réalité augmentée retrouve son application dans plusieurs domaines, on pourra citer notamment les applications potentielles dans les domaines suivants : médical, l'architecture et l'urbanisme, industrie/maintenance, robotique, militaire, etc.

Dans ce sens, les approches présentées dans ce chapitre peuvent être utilisées dans différents domaines d'application de la RA. Nous présentons dans ce qui suit quelques applications de la RA qui mettent en œuvre les outils présentés précédemment.

De nombreux travaux de RA ont été réalisés dans le domaine médical, nous pouvons citer ceux de Haouchine et associés qui proposent dans (Haouchine et al. 2013a) une méthode pour augmenter la vue laparoscopique pendant la résection de la tumeur hépatique. En utilisant des techniques de réalité augmentée, les vaisseaux, les tumeurs et plans de coupe calculés à partir des données préopératoires peuvent être superposés sur la vidéo laparoscopique.

La figure 1 montre les caractéristiques visuelles qui sont suivi (image gauche), le modèle biomécanique des éléments finis du lobe du foie déformé en utilisant les points de contrôle (image du centre), et le réseau vasculaire augmenté en bleu (image de droite).

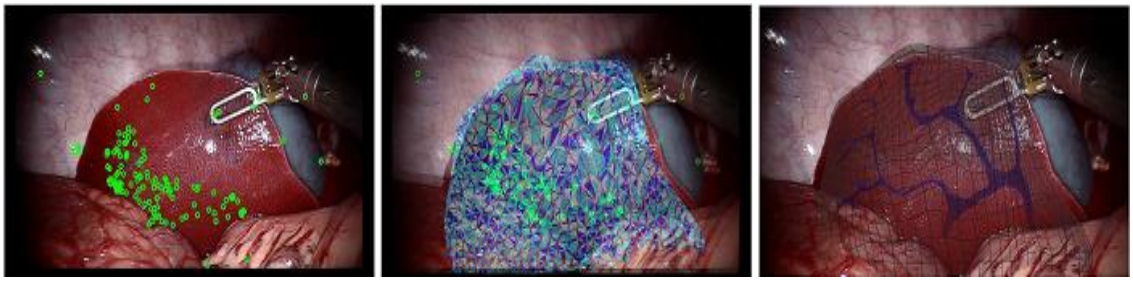


Figure 1. Superposition des vaisseaux sanguins sur la vidéo laparoscopique (Haouchine et al. 2013a).

Dans le domaine du marketing, nous pouvons citer l'application IKEA Catalog de IKEA[1]. Cette application offre la possibilité de placer des meubles virtuels dans votre propre maison, par la numérisation des pages sélectionnées dans le catalogue de IKEA (disponible pour iOS et Android), ou en parcourant les pages dans le catalogue numérique sur votre smartphone ou tablette (Figure 2). Il suffit ensuite de placer le catalogue IKEA imprimé où vous voulez mettre les meubles dans votre chambre ou salon, puis choisir un produit à partir d'une sélection de la gamme IKEA et voir à quoi ressemblera la pièce avec l'élément ajouté.

Figure 2. Visualisation de meubles virtuels dans un salon, application commerciale d'IKEA[1].

Le domaine de la maintenance industrielle a profité également des avantages qu'offre la réalité augmentée. Plusieurs travaux (Didier et al. 2005), (Benbelkacem et al. 2013), (Zenati et al. 2013), (Zenati-Henda et al. 2014) et (Zenati et al. 2015) ont été proposés dans cette optique afin d'aider les techniciens à réparer et/ou maintenir leurs équipements (figure 3).

[1] IKEA, http://info.ikea-usa.com/Catalog/

Figure 3. Un technicien utilise le système d'aide à la maintenance par la réalité augmentée proposé dans (Benbelkacem et al. 2011) pour maintenir son véhicule.

Lancé depuis juillet 2016, le jeu de réalité augmentée Pokémon Go[1] est devenu très répandu à travers le monde. Ce jeu a été développé conjointement par The Pokémon Company, Nintendo et Niantic, une ancienne filiale de Google. Son principe est de chercher et [2]attraper des Pokémons qui sont visibles sur les caméras des smartphones des utilisateurs en réalité augmentée. Le jeu permet également une interaction avec ces objets virtuels (les Pokémons) en les attrapant par des PokéBalls (figure 4).

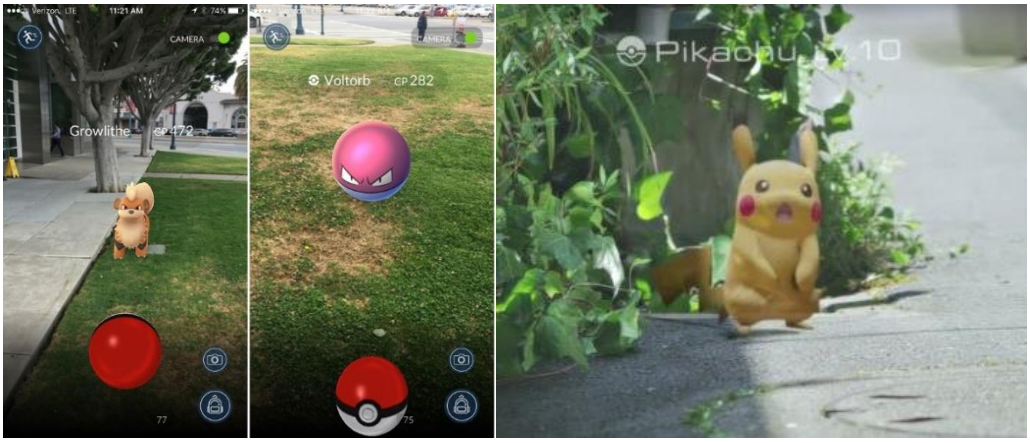


Figure 4. Le jeu de RA Pokémon Go[1].

Vue la complexité des nouveaux véhicules, Volkswagen a fait appel à Metaio pour simplifier le travail de ses techniciens. Cette collaboration a donné naissance à MARTA (Mobile Augmented Reality Technical Assistance) (Stanimirovic et al. 2014). MARTA est un système d'assistance qui permet de guider les techniciens dans leurs tâches en utilisant la réalité

[1] http://www.pokemongo.com/fr-fr/

augmentée. Le système traque la voiture en se basant sur son modèle 3D ce qui permet de positionner les objets virtuels de manière précise (figure 5).

Figure 5. MARTA (Mobile Augmented Reality Technical Assistance) (Stanimirovic et al. 2014).

Les fonds marins ont été également enrichis par des informations virtuelles grâce à la réalité augmentée lors du projet européen DigitalOcean[1]. Lors de ce dernier, des poissons virtuels, des informations multimédia et des sous-marins virtuels ont été insérés dans des piscines réelles (figure 6), dans le but de simuler aux nageurs une plongée dans un océan réel (Bellarbi, Domingues, et al. 2013).

Figure 6. Un nageur joue en réalité augmentée dans une piscine, en utilisant le Dolphyn (Bellarbi et al. 2012).

Afin d'assurer l'alignement des objets virtuels, Certaines applications se basent sur différents types de capteurs, tel que le GPS, la boussole ou encore le gyroscope. L'application Pokemon Go par exemple se base principalement sur le GPS. Néanmoins, ce type de capteurs ne fournit pas un suivi stable et précis de la scène réelle.

[1] Projet européen FP7-SME-262160, http://cordis.europa.eu/project/rcn/97193_en.html

La vision par ordinateur reste l'approche la plus attrayante pour la RA car l'estimation de la pose en utilisant uniquement la caméra simplifie la procédure d'augmentation, et offre un alignement stable des objets virtuels dans un environnement réel. Ainsi, La partie suivante, sera consacrée aux différentes approches de vision par ordinateur pour utilisées en RA.

1.4 Description du problème de localisation de la caméra

L'alignement des objets virtuels avec le monde réel peut être fait en alignant les caméras réelles et virtuelles. Afin d'obtenir un monde augmenté cohérent combinant à la fois le virtuel et le réel, nous devons attribuer à la caméra virtuelle les mêmes propriétés (extrinsèques et intrinsèques) que ceux de la caméra réelle. Par conséquent, nous devons déterminer en temps réel pour chaque image la position et l'orientation de la caméra dans la scène réelle. La Figure suivante (Figure 7) illustre le problème de recalage 2D-3D.

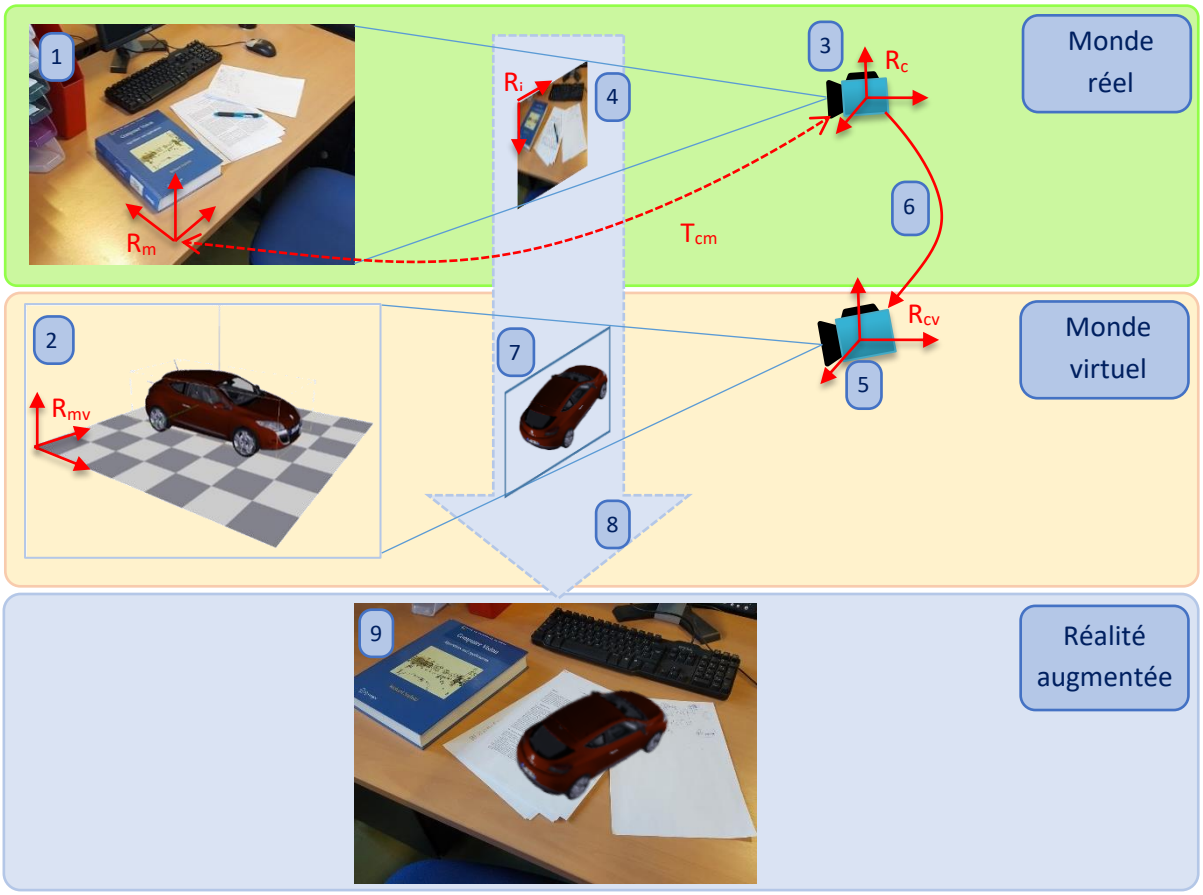


Figure 7. Principe général du processus du recalage en réalité augmentée. 1) Environnement réel. 2) Environnement virtuel. 3) Caméra. 4) Image capturée. 5) Caméra virtuelle. 6) Alignement de la caméra virtuelle avec la caméra réelle. 7) Espace de projection. 8) Mixage réel/virtuel. 9) Réalité augmentée.

(R_m, R_C, R_{mv}, R_{cv}, R_i) représentent respectivement (le repère du monde réel, le repère caméra, le repère du monde virtuelle, le repère de la caméra virtuelle et le repère image). Afin de parvenir à une composition cohérente du monde réel et virtuel, les deux caméras réelle et virtuelle doivent avoir la même position et les mêmes paramètres (focale, champ de vision (FOV) ... etc) par rapport aux repères des deux mondes réel et virtuel (R_m, R_{mv}). Ainsi, la seule inconnue est la pose de la caméra réelle par rapport au repère du monde réel.

Soit P un point dans la scène réelle de coordonnées $(X_m, Y_m, Z_m)^T$ par rapport à R_m et $(X_c, Y_c, Z_c)^T$ par rapport à R_c, , la transformation qui permet le passage de R_m vers R_c est décrite comme suit (Lepetit 2001) (Equation 1):

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = \mathrm{r} \begin{pmatrix} X_m \\ Y_m \\ Z_m \end{pmatrix} + \mathrm{t} = (\mathrm{r}\ \mathrm{t}) \begin{pmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{pmatrix} \tag{1}$$

(r t) représente la transformation entre les deux repères (monde et caméra). Celle-ci définit le vecteur de translation (t) et la matrice de rotation (r) de R_c relativement à R_m.

Soit Q la projection perspective de P sur le plan image. Les coordonnées de cette projection peuvent être calculées comme suit (Lepetit 2001) (Equation 2):

$$s \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathrm{A}} \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = \mathrm{A}\ \underbrace{(\mathrm{r}\ \mathrm{t})}_{\mathrm{T}} \begin{pmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{pmatrix} \tag{2}$$

Où « A » représente la matrice des paramètres intrinsèques (α_u, α_v : le rapport entre la distance focale et la taille horizontale et verticale du pixel, u_0, v_0: l'intersection de l'axe optique avec le plan image) et T la matrice des paramètres extrinsèques. Nous supposons que "A" est connue, ce qui fait qu'on obtient l'équation suivante (Equation 3) :

$$\mathrm{q} = \mathrm{A}^{-1}\mathrm{Q} \approx \mathrm{TP} \tag{3}$$

Comme nous l'avons dit précédemment, afin d'insérer un objet virtuel dans une scène réelle d'une manière cohérente, nous devons connaitre la pose de la caméra que nous représentons ici par la matrice « T ». Ainsi, si nous avons un ensemble de points $P_i(X_i, Y_i, Z_i)$ et leurs projections $q_i(x_i, y_i)$, nous pouvons déterminer la transformation T.

Nous présentons dans ce qui suit les différentes approches qui permettent de déterminer la pose de la caméra, ou en d'autres termes de résoudre l'équation suivante (Equation 4):

$$\mathrm{q_i} = \mathrm{TP_i} \tag{4}$$

1.5 Techniques d'estimation de la pose en RA

Nous présentons dans cette section un bref état de l'art sur les approches d'estimation de la pose que nous répartissons en deux catégories, en fonction des données disponibles (modèle 3D ou planaire).

1.5.1 Estimation de la pose basée sur un modèle 3D

Considérons le cas général où un modèle 3D est soit disponible soit calculable en ligne. Dans le premier cas, nous sommes face au problème classique dit P-nP (perspective n points). Dans le second cas, nous pouvons obtenir le modèle grâce aux techniques SLAM (Simultaneous Localization And Mapping). Aussi, dans certains cas les données 3D peuvent être mesurées directement et donc l'alignement par rapport au modèle 3D peut être fait directement.

L'estimation de pose peut se faire avec un minimum de 3 points 3D. En effet, la pose peut être représentée par six paramètres (3 angles de rotation et 3 translations) et donc 3 points seraient suffisant pour résoudre l'équation (4), ce qui correspond au problème P-3P (Perspective 3 Points).

Les approches qui permettent de résoudre ce problème se basent généralement sur deux étapes : La première étape consiste à estimer le Z_i^c pour chaque point par rapport au repère R_c via le théorème d'Al-Kashi (Katz 2007) (loi des cosinus) en utilisant le triangle CP_iP_j (avec C origine du repère R_c). Une fois que nous avons les coordonnées des 3 points, la seconde étape consiste à estimer la transformation T qui permet d'effectuer le passage de R_m vers R_c.

Comme seconde alternative nous citons la méthode des moindres carrés qui donne une solution ambiguë et requière un quatrième point pour avoir une solution unique. Celle-ci se base principalement sur la décomposition en valeurs singulière dite SVD (Singular Value Decomposition).

Plus récemment Kneip et associés (Kneip et al. 2011) ont proposé une nouvelle solution au problème de P-3P qui calcule T directement en une seule étape, sans estimation des coordonnées des points par rapport au repère de la caméra R_c. Ceci est rendu possible par l'introduction d'un nouveau repère de caméra R_c' et d'un nouveau repère du monde R_m'. Les positions et les orientations relatives de ces deux repères intermédiaires sont exprimés en utilisant uniquement deux paramètres. Ce qui simplifie la projection des points de Rm' vers Rc' en réduisant le problème à deux conditions. Le calcul de la transformation intermédiaire se fait par une équation quartique. Enfin, la pose est calculée par une simple substitution de variables.

Bien que les approches qui se basent sur le problème P-3P offrent de bonnes solutions au problème d'estimation de pose, néanmoins les approches P-nP sont préférables vue que la précision de la pose calculée augmente avec le nombre de points.

Quan et Lan (Quan & Lan 1999) sont parmi les pionniers dans le domaine à avoir étendu leur algorithme de P-3P à P-4P puis P-5P pour finalement aboutir au P-nP. Dans l'approche EP-nP (Lepetit et al. 2009), les points de coordonnées 3D sont exprimées sous forme d'une somme pondérée de quatre points de contrôle virtuels. Le problème de la pose est alors réduit à l'estimation des coordonnées de ces points de contrôle dans le repère de la caméra. Le principal avantage de cette approche est qu'elle réduit la complexité de calcul.

Dans les approches à une étape, la transformation linéaire directe (Direct Linear Transform, DLT) est certainement la plus ancienne (Hartley & Zisserman 2005). Bien que pas très précise, cette solution et ses dérivées ont largement été pris en compte dans les applications de RA.

P-nP est un problème non-linéaire ; néanmoins une solution reposant sur la solution d'un système linéaire peut être considérée. En effet, à partir de l'équation (5) :

$$\mathrm{D} \times \mathrm{C} = \begin{pmatrix} \cdot \\ \cdot \\ D_i \\ \cdot \\ \vdots \\ \cdot \end{pmatrix} \times \mathrm{C} = 0 \tag{5}$$

Où: $D_i = \begin{pmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -x_i X_i & -x_i Y_i & -x_i Z_i & -x_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -y_i X_i & -y_i Y_i & -y_i Z_i & -y_i \end{pmatrix}$

Et $C = (r_1 \quad t_x \quad r_2 \quad t_y \quad r_3 \quad t_z)^T$

La solution de ce système homogène est le vecteur propre de D correspondant à la valeur propre minimale (calculée par une décomposition en valeurs singulières de D). Malheureusement, cette solution est très sensible au bruit et donc il est préférable d'opter pour une approche qui prend en considération la non-linéarité du système.

Parmi les solutions qui prennent en considération la non-linéarité du système, nous pouvons citer POSIT. Proposée par Dementhon et Davis (Dementhon & Davis 1995), l'idée consiste à utiliser un système de projection orthogonale afin que le problème devienne linéaire, ensuite d'une manière itérative on revient au système de base dit projection perspective.

Selon Marchand et associés. (Marchand et al. 2016), la meilleure solution au problème P-nP consiste à estimer la transformation T en minimisant l'erreur de re-projection en utilisant une approche de minimisation non linéaire tel que l'algorithme de Levenberg-Marquardt (More 1978). Dans ce sens, nous pouvons citer l'approche proposée par Olsson et associés. (Olsson et al. 2009) ou encore celle de Lu et associés (Lu et al. 2000).

Dans le cas où le nombre de points augmente considérablement, il n y'a pas de solution à complexité linéaire pour le problème P-nP. Comme solutions possibles pour ce cas, nous citons les approches suivantes : EPnP (Lepetit et al. 2009), OPnP (Zheng et al. 2013), GPnP (Kneip et al. 2013), et UPnP (Kneip et al. 2014).

D'autres méthodes se basent sur le suivi d'un modèle pour l'estimation de pose. L'idée est de définir une distance entre le point d'un contour dans l'image et la projection de la ligne 3D correspondant au modèle 3D. Nous supposons ici que nous avons une estimation de la pose. Le modèle 3D est tout d'abord projeté selon la pose, le contour est échantillonné puis on effectue une recherche des échantillons tout le long du contour normal. Ensuite, la pose est estimée par le biais d'une approche non-linéaire d'optimisation qui consiste à minimiser l'erreur entre les points sélectionnés et les contours projetés. (Figure 8).

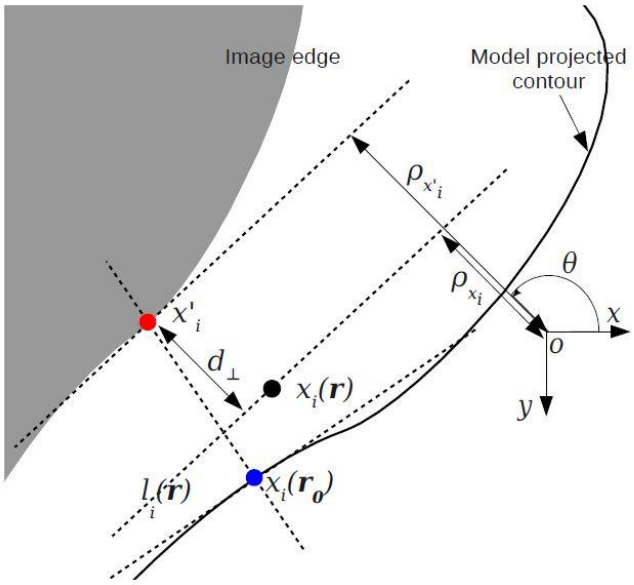


Figure 8. Suivi de contour pour l'estimation de pose, extrait de (Petit et al. 2013). À partir de la pose initiale r_0, une recherche monodimensionnelle est réalisée le long de la normale au contour projeté sous-jacent au point de mesure $x_i(r_0)$. Et minimisation de la distance d entre le point x'_i et la ligne $l_i(r)$.

Comport et associés (Comport et al. 2006) ont proposé un algorithme de suivi basé sur un modèle 3D. Une estimation non linéaire de la pose est formulée à l'aide d'une approche d'asservissement visuel virtuel. La robustesse est obtenue en intégrant un estimateur dans la loi du contrôle visuel par l'intermédiaire d'une mise en œuvre itérative de la méthode des moindres carrés pondérés. Cette approche est ensuite étendue pour résoudre le modèle 3D. Les résultats ont montré que la méthode est robuste à l'occlusion, et aux changements d'éclairage.

Jusqu'à présent, nous avons abordé des approches qui se basent sur des modèles 3D disponibles, ce qui n'est pas toujours le cas. Dans ce sens, d'autres approches nommées VSLAM

(Vision based Simultaneus Localization and Mapping) ont vu le jour tel que : (Davison 2003), (Eade & Drummond 2006), (Agrawal & Konolige 2008). L'idée est d'estimer en même temps la structure de la scène, et la pose de la caméra.

Les premières approches VSLAM se basaient sur le filtre bayésien. Davison (Davison 2003) a proposé l'utilisation du filtre de Kalman étendu pour l'intégration des données. Par contre, Eade et Drummond (Eade & Drummond 2006) ont utilisé le filtre à particule. Dans ce genre d'approches, les données sont intégrées de manière séquentielle dans le filtre. Les différentes mises à jours (la position de la caméra, sa vitesse, et la structure de la scène) sont faites séquentiellement, l'une après l'autre. Toutes les poses précédentes sont mises de côté, et donc le nombre de paramètres à estimer croit avec la taille de la carte (map).

D'autres approches se basent sur la minimisation de l'erreur de re-projection. Ainsi, les approches dites « Bundle Adjustment (BA) » permettent d'estimer le mouvement de la caméra en minimisant l'erreur entre les points prédits et les points observés et de construire ainsi la carte (mapping). Parmi les travaux se basant sur cette méthode nous citons (Mouragnon et al. 2006) et (Sibley et al. 2010).

Malgré que certains travaux (Nist & Bergen 2004), (Mouragnon et al. 2006) ont démontré la possibilité d'utiliser les SLAM en RA, néanmoins ce genre d'approches fait défaut pour ce qui est de la localisation absolue, et reste complexe et couteux en temps de calcul.

Pour répondre à ce problème, l'approche PTAM (Parallel Suivi And Mapping) (Klein & Murray 2007) (Klein & Murray 2009) vient dissocier le suivi du mapping. Elle consiste à mettre en place en parallèle la partie mapping (BA) et une méthode de suivi qui se base uniquement sur les points déjà reconstruits du map pour localiser la caméra. Cette approche a connu le succès dans différents domaines d'applications notamment en RA (voir figure 9) (Wagner et al. 2008), (Ventura & Höllerer 2012).

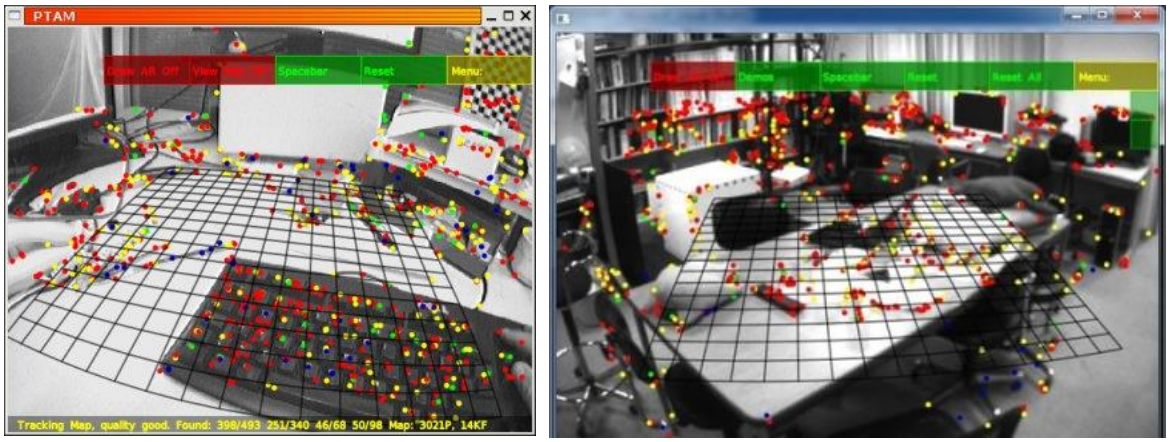


Figure 9. Exemple d'application de la technique PTAM en RA. (Klein & Murray 2009)

Contrairement aux approches VSLAM abordées précédemment qui utilisent quelques pixels de l'image pour l'estimation de la pose et la structure de la scène, ils existent d'autres approches dites Dense SLAM qui utilisent tous les pixels de l'image. Nous pouvons citer DTAM (Newcombe et al. 2011) ou LSD-SLAM (Engel et al. 2014). Cependant, ces techniques sont beaucoup plus couteuses en temps de calcul.

Jusque-là, nous avons abordé les approches mono-caméra. Toutefois, ils existent plusieurs travaux qui se basent sur l'utilisation de plusieurs caméras ou encore d'autres types de capteurs, tel que la Kinect et la technique KinectFusion (Izadi et al. 2011). Ces types de systèmes permettent de déterminer directement la position 3D des points (figure 10). Cependant, ces approches requièrent une étape d'apprentissage et de reconstruction lente.

Figure 10. Interaction des particules virtuelles avec une scène réelle reconstruite par KinectFusion (Izadi et al. 2011)

1.5.2 Estimation de pose à base d'informations coplanaires

Il n'est pas toujours évident d'avoir des données 3D pour l'estimation de pose. Considérer une scène planaire simplifie beaucoup le problème. Dans ce sens, l'estimation de pose revient à un processus d'estimation de mouvement de la caméra.

Il est donc possible de passer outre les données 3D et d'avoir d'autres de contraintes par rapport à la scène observée. Dans cette partie du chapitre nous abordons les approches qui se basent sur l'extraction d'information 2D de l'image et les informations géométriques de la scène planaire. L'objectif étant d'estimer le déplacement de la caméra entre deux images au lieu d'estimer la pose ; le modèle 3D est de ce fait remplacé par une image de référence.

1.5.2.1 Approches géométriques

L'approche géométrique consiste à estimer le mouvement via une mise en correspondance des points. L'objectif est d'estimer le mouvement 3D de la caméra entre deux acquisitions en utilisant uniquement les informations 2D extraites des deux images. Le calcul de l'homographie (la transformation entre les deux images) entre les deux images est souvent utilisé dans ce cas. En effet, passer du cas général à un cas restreint (cas d'une scène planaire) simplifie beaucoup l'estimation de pose.

D'un point de vue général, il n'existe pas un modèle de mouvement 2D utilisable pour une scène 3D. Toutefois, quand il s'agit d'une scène planaire, un modèle de mouvement 2D suffit pour déterminer le mouvement 3D de la caméra.

Soit x_1 un point appartenant à l'image I_1 et x_2 un point de l'image I_2 (I_1, I_2 deux images de la même scène avec un point de vue diffèrent). Les deux points (x_1, x_2) sont liés par l'homographie H_1^2 comme suit (Equation 6) (Marchand et al. 2016):

$$x_2 = H_1^2 x_1 \tag{6}$$

H_1^2 peut être estimée via l'algorithme DLT (Direct Linear Transformation) (Hartley & Zisserman 2005). Une fois l'homographie estimée, la pose peut être calculée en décomposant l'homographie (Faugeras & Lustman 1988). Dans le cas d'une scène planaire, la pose peut être calculée directement si la position 3D de certains points du plan est connue, et leur projection sur l'image capturée est également connue (Oberkampf et al. 1996), (Zhang 2000).

La simplicité de cette approche a rendu son utilisation en RA un standard. Ainsi, plusieurs systèmes d'identification de cibles codées se basant sur cette approche ont été présentés dans la littérature. L'idée est de placer des marqueurs différents (cibles codées) par leurs couleurs ou leurs formes dans l'environnement réel (Belghit et al. 2012). En se basant sur le fait que les marqueurs sont connus a priori, on peut alors décomposer l'estimation de pose en trois phases (Zendjebil 2010) :

1. Détection et identification des marqueurs : cela consiste à extraire de l'image les marqueurs visibles par la caméra et de les identifier.

2. Mise en correspondance 2D/3D : à chaque marqueur identifié dans l'image est associée une position 3D.
3. Calcul de la pose de la caméra : estimation de la pose à partir des appariements 2D/3D.

Toutefois, les marqueurs planaires ont été démocratisés avec l'avènement de la bibliothèque ARToolkit (Hirokazu & Billinghurst 1999), où les marqueurs utilisés ont une forme rectangulaire, des bords noirs sur un fond blanc et ont un code permettant de les identifier. Cette bibliothèque a connu de nombreuses propositions d'améliorations tel que Artag (Fiala 2004) et ArtoolkitPlus (Wagner & Schmalstieg 2007). De nombreuses techniques et librairies de marqueurs avec des formes différentes ont été proposées afin d'assurer un recalage stable et précis en RA. Cependant, l'inconvénient majeur des marqueurs (cibles codées) est l'hétérogénéité avec la scène à augmenter. La figure 11 montre quelques librairies de cibles codées.

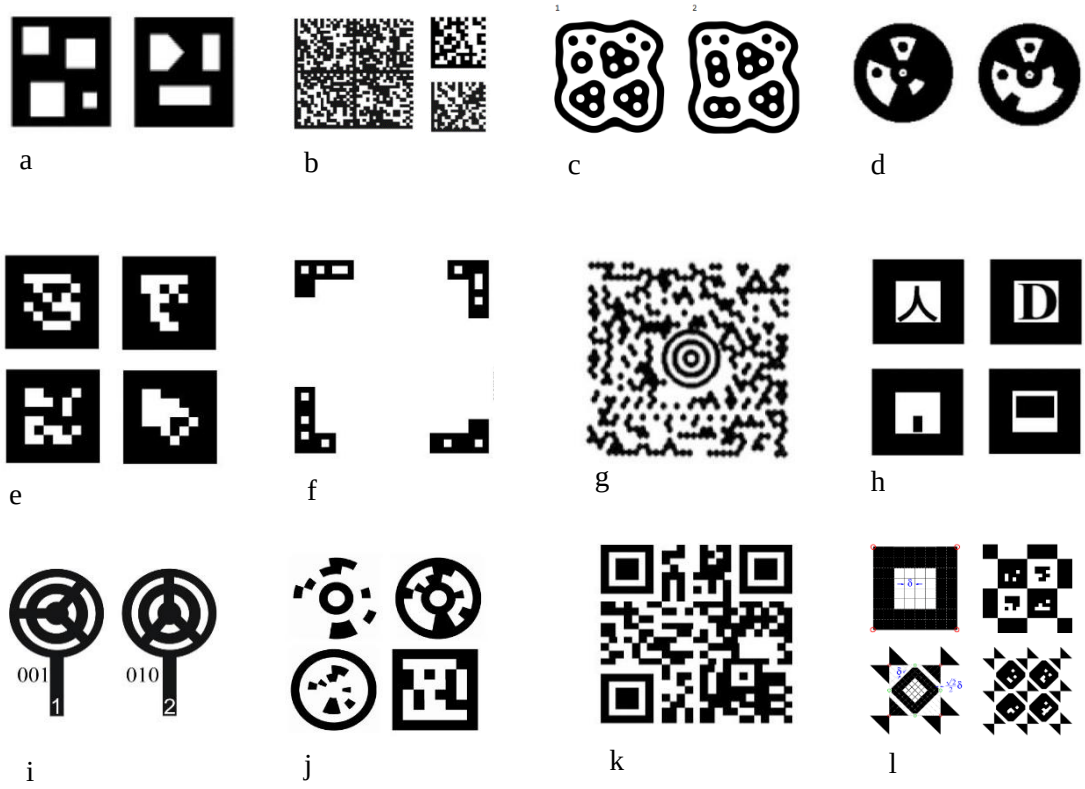


Figure 11. Quelques exemples de librairies de marqueurs. a : ARSTUDIO (Malik et al. 2002), b : Data matrix (Naimark & Foxlin 2002), c : ReacTIVision (Bencina & Kaltenbrunner 2005), d: Intersense (Boulanger 2004), e: Artag (Fiala 2004), f : L-Split Marker (Han & Zhao 2016), g : Maxicode (Wang & Ye 2000), h : ARToolkit (Hirokazu & Billinghurst 1999), i : LARICS marker (Mutka et al. 2008), j : Cantag markers (Rice et al. 2006), k : QRCode (Kan et al. 2009), l : Caltag (Atcheson et al. 2010).

Récemment, DeTone et associés (DeTone et al. 2016) ont considéré le problème d'estimation de l'homographie entre deux images comme étant un problème d'apprentissage. Ils ont appliqué de ce fait un réseau de neurones convolutionnel (Convolutional Neural Network, CNN) afin de le résoudre. Cependant, ce type de réseau de neurones est couteux en temps de calcul.

1.5.2.2 Approches basées apparence

Les approches précédentes se basent sur des méthodes géométriques. Une autre alternative est d'intégrer l'estimation du mouvement dans le traitement d'images. Considérons le modèle 2D comme une image de référence, l'objectif est d'estimer le mouvement entre l'image capturée et l'image de référence à l'échelle du pixel. Le modèle étant défini par un ensemble de pixels, nous devons retrouver leurs nouvelles positions dans l'image. Au lieu d'utiliser l'homographie pour déterminer la pose, l'alignement peut être défini directement comme un problème de minimisation des dissimilarités ou de maximisation des similarités entre l'apparence de la zone d'intérêt dans l'image de référence et celle de la zone d'intérêt dans l'image capturée, ce qu'on appelle les approches basées apparence.

Si par exemple, l'apparence est définie comme l'intensité des pixels appartenant à un patch, la différence ou bien la dissimilarité est considérée comme la somme des carrés des différences dites SSD (Sum of Squared Differences), ce qui revient à l'algorithme KLT proposé dans (Lucas & Kanade 1981).

L'algorithme de suivi proposé par Benhimane & Malis (Benhimane & Malis 2004) est également basé sur la minimisation de la SSD entre un modèle donné et l'image courante en appliquant l'algorithme ESM (Efficient Second order Minimisation). Ce dernier possède les mêmes propriétés de convergence que la méthode de Newton, mais avec un temps de calcul plus rapide.

1.5.3 Discussion

Dans cette partie du chapitre, nous avons abordé les différentes approches permettant l'estimation de la pose selon les données disponibles (3D ou 2D), allant du problème P-nP jusqu'aux SLAM, sans oublier le cas d'une scène planaire. Pour ceux qui souhaitent détailler ces approches nous vous recommandons de consulter le Survey proposé par Marchand et associés (Marchand et al. 2016), qui regroupe la plupart des approches d'estimation de pose pour la RA. Nous présentons également dans ce qui suit une classification des techniques d'estimation de pose (Figure 12).

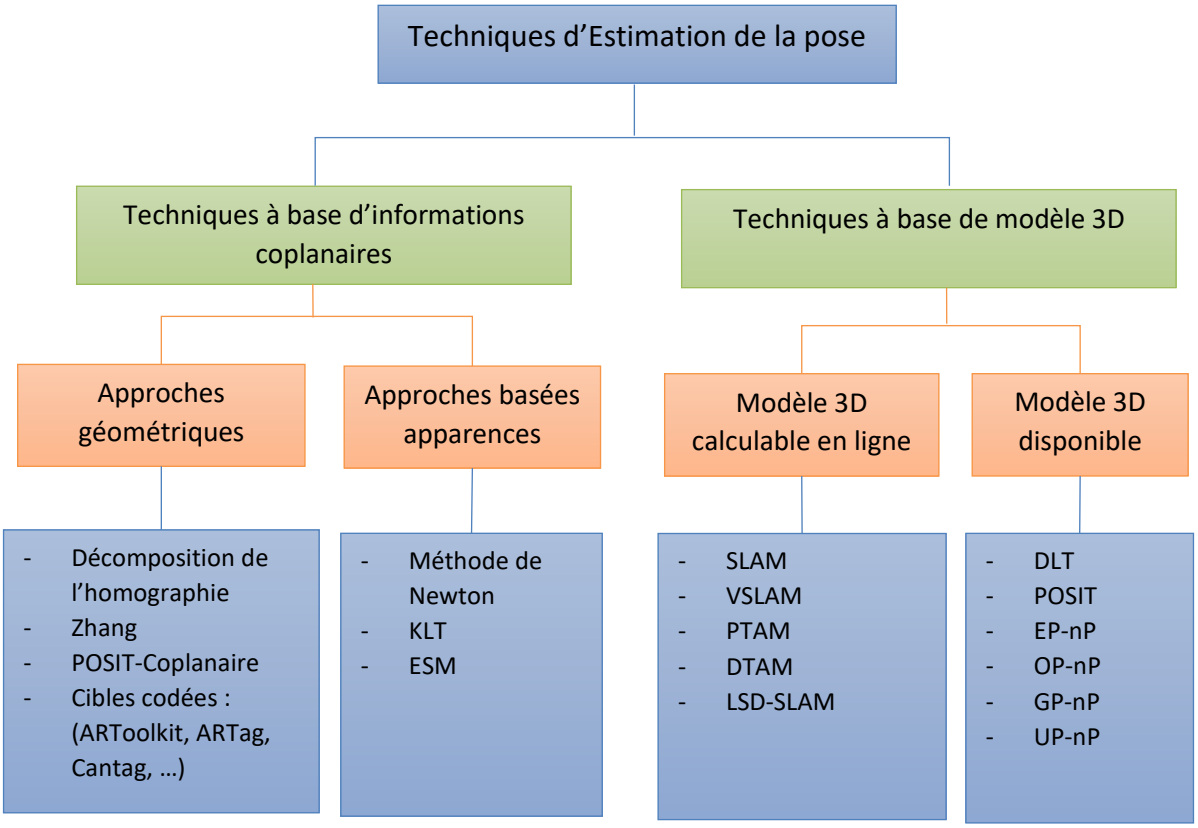


Figure 12. Classification des techniques d'estimation de la pose

D'un point de vue pratique la détection et la description des caractéristiques (points d'intérêt) est nécessaire pour l'estimation de pose. Dans ce sens, la section suivante regroupe les différentes approches utilisées.

1.6 Détection et description des caractéristiques

La réalité augmentée implique le repérage spatial de l'objet à augmenter par rapport à la caméra. D'autres parts, les progrès récents en vision par ordinateur rendent possible la détection, la reconnaissance et le suivi d'objets en exploitant uniquement leurs caractéristiques propres. Cette technique est devenue très répandue en réalité augmentée. Nous présentons dans ce qui suit les différentes techniques de détection et de description des caractéristiques de l'image.

1.6.1 Détection de caractéristiques visuelles

Obtenir des points d'intérêt stables est une étape essentielle pour tous les processus de vision par ordinateur basée sur les points. Récemment, la détection de coins est devenue la technique la plus utilisée, en raison de ses bonnes performances en termes de répétabilité et de temps de traitement. Un coin est défini comme étant le point d'intersection de deux lignes ou

contours. Mathématiquement, il se réfère au point où existent deux orientations de gradient dominantes mais différentes.

On peut classifier les différentes approches de détection de coins selon trois catégories (Li et al. 2015) :

1. **Détection de coins basée gradient** : ces techniques se basent sur le calcul du gradient, tels que, Harris (Harris & Stephens 1988), KLT (Lucas & Kanade 1981), et Shi-Tomasi (Shi & Tomasi, 1994). Ces techniques sont robustes mais couteuses en temps de calcul.
2. **Détection de coins basée contours** : ces techniques étudient la forme des contours afin d'identifier les coins. A savoir les détecteurs DoG-curve (Zhang et al. 2010), ANDD (Willis & Sui 2009) et Hyperbola fitting (Shui & Zhang 2013). Cependant, ces techniques sont très sensibles aux bruits.
3. **Détection de coins basée modèles** : ces techniques se basent sur la comparaison des pixels autour d'un modèle. SUSAN (Smith & Brady 1997) et FAST (Rosten & Drummond 2006) utilisent cette approche. Récemment, des modèles ont été combinés avec des techniques d'apprentissage automatique (machine learning), comme les arbres de décision, pour une détection plus rapide de coins. Tels que le détecteur AGAST. Ces techniques sont les plus utilisées pour les applications en temps réel.

Etant le premier détecteur de coins basé modèles, FAST (Fast Accelerated Segment Test) représente une percée dans l'évolution des détecteurs de coins. Il est basé sur le principe du test de segments accéléré ou AST (Accelerated Segment Test) qui est une modification du détecteur SUSAN. AST classe un point p (avec une intensité I_p) comme un coin, si n pixels adjacents dans un cercle de rayon 3 autour de p sont plus lumineux que $I_p + t$ ou plus sombre que $I_p - t$, avec t un seuil prédéfini. On attribue à chaque coin un score s, défini comme étant le seuil pour lequel p peut être classé comme un coin (Figure 13). Un arbre de décision est ensuite appris afin d'accélérer le classement des points candidats.

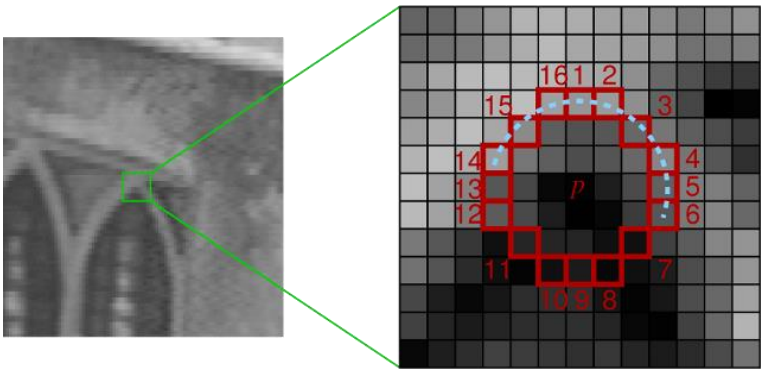


Figure 13. Exemple de détection d'un coin par le détecteur FAST (Rosten & Drummond 2006).

Le détecteur FAST-ER (Rosten et al. 2010), qui est une version améliorée de FAST, augmente l'épaisseur du modèle circulaire afin d'augmenter la stabilité des coins détectés ; cependant, il est devenu plus lent par rapport à FAST.

Le détecteur AGAST (Adaptive and Generic Corner Detection Based on the Accelerated Segment Test) (Mair et al. 2010) qui est basé également sur la technique AST, a considérablement amélioré la construction et l'utilisation des arbres de décision utilisés dans l'AST, AGAST utilise des arbres de décision binaires pour terminer le test des segments accélérés. Deux arbres sont construits, le premier pour les régions homogènes et le second pour les régions structurées. En combinant les deux arbres de décision, AGAST s'adapte automatiquement pour fournir l'arbre de décision le plus efficace pour la région de l'image étudiée. AGAST n'a pas besoin d'une phase d'apprentissage et il préserve la même détection et répétabilité que le détecteur FAST.

Les détecteurs basés-AST ne fournissent pas la détection multi-échelle. Dans leur détecteur ORB, Rublee et associés (Rublee et al. 2011) ont utilisé FAST sur une image à espace d'échelle. Les points détectés ont été affinés par la métrique du détecteur Harris (Harris & Stephens 1988). Cependant, des points redondants sur différents niveaux de la pyramide ont été détectés, depuis, la suppression non-maximale n'est pas utilisée entre les échelles.

Leutenegger et associés ont proposé le détecteur BRISK (Leutenegger et al. 2011), ce dernier se base sur le détecteur AGAST et raffiné par la métrique de mesure de coins de FAST. Le détecteur BRISK identifie les points d'intérêts dans une pyramide d'images à espace d'échelle et effectue ensuite l'élimination des points entre les niveaux de la pyramide.

Bien que ce détecteur soit connu par sa rapidité, néanmoins sa répétabilité reste limitée, vu qu'il se base sur le même principe de mesure de coins (AST) pour la détection et le raffinement.

1.6.2 Description des caractéristiques (points d'intérêt)

L'utilisation des caractéristiques visuelles de l'objet (contours, coins, points d'intérêt…) permet la reconnaissance et le suivi d'objets. En général, la reconnaissance se fait en comparant des patchs extraits d'un côté de l'image de référence contenant l'objet à reconnaitre, et de l'autre côté de l'image acquise par la caméra.

Étant donné deux patchs (régions autour des points d'intérêts) extraits de deux images distinctes, la comparaison se fait en mesurant la similitude des deux patchs. Comment pouvons-nous alors déterminer cette similitude ?

Nous pouvons mesurer la similitude de pixel à pixel en calculant leur distance euclidienne, mais cette mesure est très sensible au bruit, rotation, translation et changements d'éclairage. Dans la plupart des applications, nous requérons des résultats robustes face à de tels changements.

D'où l'intérêt des descripteurs. Un descripteur est une fonction qui est appliquée sur le patch afin de le décrire d'une manière invariante pour tous changements sur l'image (par exemple la rotation, l'éclairage, le bruit, etc.).

Ainsi, pour comparer deux images ou bien deux parties d'images, nous calculons leurs descripteurs et nous mesurons par la suite leur similarité par la mesure de la similitude du descripteur, qui à son tour se fait en calculant leur distance entre les descripteurs.

Le pipeline commun pour l'utilisation de descripteurs est le suivant :

1. Sélectionner des régions (patchs) autour des points d'intérêt détectés dans l'image. Ces patchs sont de forme carré ou circulaire selon les propriétés du descripteur à appliquer.
2. Décrire chaque région (le patch) sous forme d'un vecteur de caractéristiques, en utilisant ce descripteur.
3. Calculer la distance entre les vecteurs en utilisant une mesure de similarité

Dans la littérature, la majorité des travaux se focalisent sur la description des points d'intérêt. Les techniques de descriptions ou les descripteurs ont été regroupés en deux grandes familles, les techniques à base de virgule flottante (Floating point descriptors), et les descripteurs à base binaire (binary descriptors). Dans ce qui suit, nous allons donner un aperçu des deux familles.

1.6.2.1 Descripteurs à base de virgule flottante

En 1999, Lowe a proposé un descripteur invariant appelé SIFT (Lowe 1999) (Scale-invariant feature transform). La méthode proposée par Lowe comprend deux étapes : la détection de points d'intérêt et le calcul de descripteurs ; les points d'intérêt sont calculés par une différence de gaussienne (DoG). Pour chaque point, on détermine une orientation intrinsèque qui sert à la construction d'un histogramme des orientations locales des contours. Cet histogramme qui est sous forme d'un vecteur de dimensions 128 constitue le descripteur SIFT du point d'intérêt. Ces descripteurs présentent l'avantage d'être invariants à l'orientation et à la résolution de l'image, et peu sensibles à son exposition, à sa netteté ainsi qu'au point de vue 3D (Figure 14).

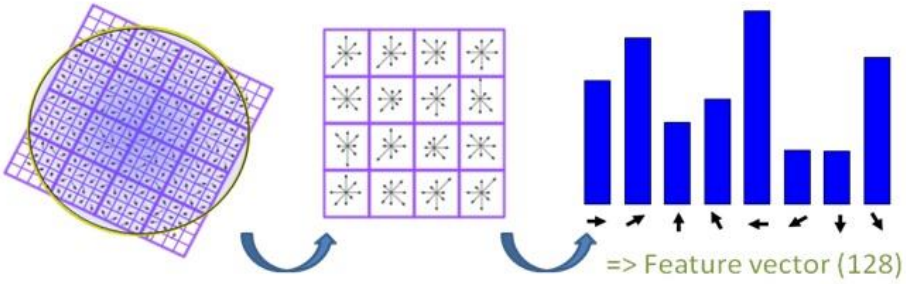


Figure 14. Le principe du descripteur SIFT (Lowe 2004).

Une version améliorée de cette méthode est proposée en 2004 (Lowe 2004) afin d'améliorer sa robustesse, mais l'inconvénient majeur est que cette méthode est très gourmande en temps de calcul.

Des variantes du descripteur SIFT ont été proposées afin de réduire le temps de calcul, tels que, PCA-SIFT (Ke & Sukthankar 2004) qui applique l'analyse en composantes principales (ACP) sur le gradient normalisé du patch au lieu d'utiliser des histogrammes pondérés de SIFT, ce qui a réduit la taille du descripteur de 128 à 36. D'autres variantes de SIFT afin de réduire sa dimension et le rendre plus rapide, sont proposées dans (Kordelas & Daras 2009; Liao et al. 2013; Valenzuela et al. 2014). Nous pouvons trouver dans (Wu et al. 2013) une étude comparative du descripteur SIFT avec ses variantes.

Mikolajczyk et Schmid (Mikolajczyk & Schmid 2005) ont comparé les performances des dix descripteurs et ils défendent leur GLOH (Gradient Localisation and Orientation Histogram) qui est une extension du descripteur SIFT. Le descripteur GLOH a appliqué l'ACP sur le descripteur SIFT afin de réduire sa dimension à 64. Il surpasse SIFT et d'autres descripteurs par sa robustesse et son caractère distinctif.

Se basant sur les propriétés de SIFT, Bay et associés (Bay et al. 2006) ont proposé la méthode SURF (Speeded Up Robusts Features). Cette dernière se base sur les points d'intérêt à partir du DoG, et utilise les ondelettes de HAAR pour la description afin de minimiser le temps de calcul. Cependant, cette technique reste lente et montre un résultat moins robuste que SIFT. Surf a été hybridé avec des techniques de suivi (suivi), comme SURF avec flow optique (Noguchi & Yanai 2012) et SURF avec Kalman (Ta et al. 2009). Ces approches ont été largement utilisées pour le suivi en réalité augmentée. Cependant, elles présentent certaines anomalies comme la divergence du trackeur, et l'accumulation de l'erreur.

L'équipe IRVA du CDTA ont proposé également une amélioration du descripteur SURF (Hamidia et al. 2014). Cette amélioration permet de minimiser la zone de recherche par SURF. L'idée de base est d'appliquer autour de l'objet détecté une région d'intérêt (voir Figure 15), afin de minimiser la zone de recherche pour le prochain image. Cette méthode a permis un gain en temps de calcul. Cependant, elle génère des résultats erronés lors les mouvements brusques.

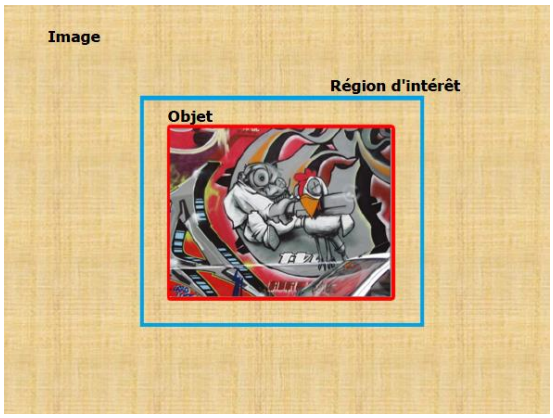


Figure 15. Application d'une région d'intérêt autour de l'objet détecté (Hamidia et al. 2014).

Tola et associés (Tola et al. 2010) ont proposé un descripteur appelé DAISY, dérivée de SIFT et GLOH avec une configuration de la forme d'une marguerite « Daisy en anglais ». Ce descripteur hybride peut être utilisé pour estimer la profondeur à partir de deux larges bases d'images. Cependant, ce descripteur est couteux en temps de calcul.

Le laboratoire de recherche de Google a aussi inventé son propre descripteur appelé CONGAS (COmpact Normalized GAbor Sampling) (Zheng et al. 2009) qui est basé sur les ondelettes de Gabor de différentes échelles. Alcantarilla et associés. (Alcantarilla et al. 2012) ont proposé une technique de détection et de description appelé KAZE, qui détecte et décrit un patch par une approche de filtrage de diffusion non linéaire.

L'apprentissage automatique (machine learning) a été introduit également dans les descripteurs. Dans (Simonyan et al. 2014), (Ji et al. 2012) et (Winder & Brown 2007) des descripteurs basés sur l'histogramme de gradient orienté (HOG-Like) comme SIFT et GLOH, ont été entrainés afin de réduire leurs dimensions et améliorer leurs caractères discriminatifs. Cependant, ils sont devenus plus lents.

Les techniques présentées jusqu'à présent utilisent un codage basé sur la virgule flottante pour le calcul de vecteurs descriptifs. Ce qui les rend couteuses en temps de calcul et en mémoire.

Afin de pallier ces problèmes, des techniques employant des calculs binaires des vecteurs descriptifs ont été proposées. En outre, ces techniques utilisent la distance de Hamming en tant que mesure de distance entre deux chaînes binaires, ce qui accélère le matching des vecteurs, et ceci est le point fort des descripteurs binaires.

1.6.2.2 Descripteurs binaires

En général, les descripteurs binaires sont composés généralement de trois parties essentiels :

- La compensation de l'orientation du patch.

- La modélisation du pattern (sampling pattern)
- Le choix des paires.

Ainsi, nous pouvons construire un descripteur binaire comme suit : considérons tout d'abord un patch autour d'un point d'intérêt. Après avoir tourné le patch vers son orientation dominante, l'étape suivante alors est de modéliser le pattern. C.à.d. choisir une stratégie de division du patch. Par exemple, des sous-régions égales ou bien des cercles concentriques de tailles différentes, comme dans le cas de BRISK (Leutenegger et al. 2011) et FREAK (Alahi et al. 2012) (voir Figure 16).

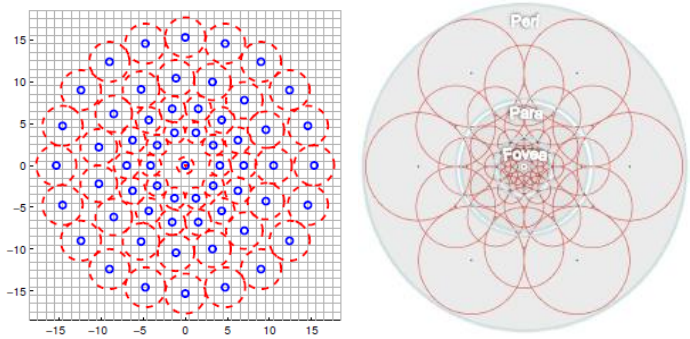


Figure 16. Modélisation du pattern dans un descripteur binaire. De gauche à droite : BRISK (Leutenegger et al. 2011), FREAK (Alahi et al. 2012).

On choisit ensuite N paires de points (512 paires de points dans le cas de BRISK) suivant une technique de sélection des paires. On compare par la suite, sur toutes les paires des points par le test binaire τ la valeur d'intensité du premier point $I(p_1)$ (ou la sous-région du patch) de la paire avec la valeur de l'intensité du deuxième point $I(p_2)$ de la paire, suivant le l'équation (7).

$$\tau = \begin{cases} 1 & si\ I(p_1) \geq I(p_2) \\ 0 & sinon \end{cases} \quad (7)$$

Après avoir testé les N paires. Nous aurons alors une chaîne binaire de dimension N, qui code les informations locales autour du point d'intérêt. La mise en correspondance ensuite des vecteurs binaires se fait par la distance de Hamming.

Plusieurs techniques de description binaire ont été proposées. Dans ce qui suit, nous allons donner un aperçu sur les descripteurs binaires les plus connus.

Présenté en 2010, BRIEF (Binary Robust Independent Elementary Features) (Calonder et al. 2010) a été le premier descripteur binaire publié. Il ne dispose ni d'une méthode de modélisation du patch élaborée ni d'un mécanisme de compensation de l'orientation, ce qui le rend plus facile à comprendre et à implémenter.

Pour construire un descripteur BRIEF de longueur *n*, on doit déterminer *n* paires (Xi, Yi) de points dans un patch de taille *S*S*, lissé par un filtre gaussien. Notons X et Y les vecteurs du point Xi et Yi, respectivement.

Calonder et associés (Calonder et al. 2010) ont proposé cinq possibilités pour déterminer les vecteurs X et Y (figure 17) :

1. X et Y sont choisis d'une manière purement aléatoire.
2. X et Y sont échantillonnés de façon aléatoire en utilisant une distribution gaussienne, ce qui signifie que des emplacements qui sont plus proches du centre du patch sont préférés.
3. X et Y sont prélevés au hasard en utilisant une distribution gaussienne où les X sont échantillonnés avec un écart type de 0,04 * S^2. Les Yi sont prélevés en utilisant une distribution gaussienne par rapport à Xi avec un écart type de 0,01 * S^2.
4. X et Y sont échantillonnés au hasard à partir d'un emplacement proche du centre du patch.
5. Pour chaque i, Xi est (0, 0) et Yi prend toutes les valeurs possibles sur une grille polaire.

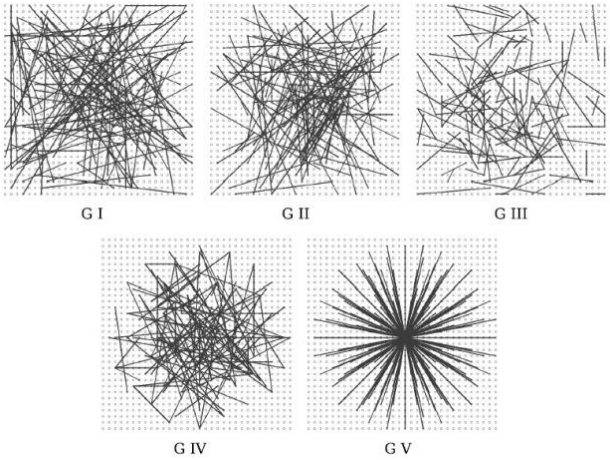


Figure 17. Différentes possibilités de choix de paires pour BRIEF (Calonder et al. 2010).

Selon les auteurs de BRIEF, la 4eme configuration pour le choix des paires a donné de meilleurs résultats par rapport aux autres. Cependant, ce descripteur n'est pas invariant à rotation. En outre, sa sensibilité au bruit est considérable.

Le probleme de l'invariance à la rotation de BRIEF a été traité par Rublee et associés, en introduisant le détecteur-descripteur ORB (Rublee et al. 2011). Ce dernier se base sur le descripteur BRIEF, avec quelques différences :

1. ORB utilise son propre détecteur FAST-orienté (Oriented-FAST).
2. ORB utilise un mécanisme de compensation de l'orientation, qui le rend invariant à la rotation.

3. Dans ORB, les paires de points optimales sont apprises par une technique d'apprentissage automatique afin de sélectionner des tests binaires pertinents et non corrélés, alors que dans BRIEF, le choix des paires est aléatoire.

Pour la compensation de l'orientation, Rublee et associés ont adopté la technique basée sur la mesure du centre de gravité de l'intensité proposée dans (Rosin 1999) afin de calculer la rotation du patch.

Le descripteur BRISK (Binary Robust Invariant Scalable Keypoints) proposé par Leutenegger et associés (Leutenegger et al. 2011) est composé de cercles concentriques (Figure 18). Lors de la sélection de chaque point, on prend un petit patch autour du point et on applique un filtre gaussien. Les cercles rouges dans la figure ci-dessous montrent la taille de l'écart type du filtre gaussien appliqué à chaque point sélectionné.

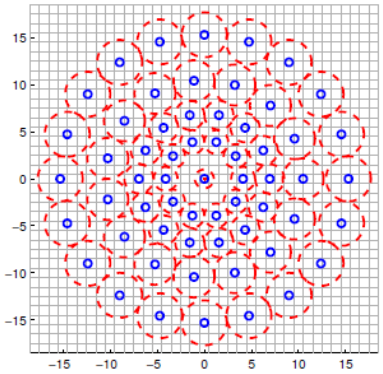


Figure 18. Modèle du pattern du descripteur BRISK (Calonder et al. 2010).

Lors de la comparaison binaire des sous-régions, BRISK regroupe les distances des paires selon un seuil prédéfini. Les paires à longues distances sont utilisées pour déterminer l'orientation, alors que les paires à courtes distances sont utilisées pour les comparaisons d'intensité qui construisent le descripteur binaire.

Pour l'invariance à la rotation, BRISK estime l'orientation du patch en calculant les gradients locaux entre les paires de points qui ont une longue distance.

La figure ci-dessous (Figure 19) montre un exemple d'utilisation de BRISK pour la mise en correspondance entre des images avec un changement de point de vue. Les cercles rouges sont les points d'intérêt détectés. Les lignes vertes sont des correspondances valides.

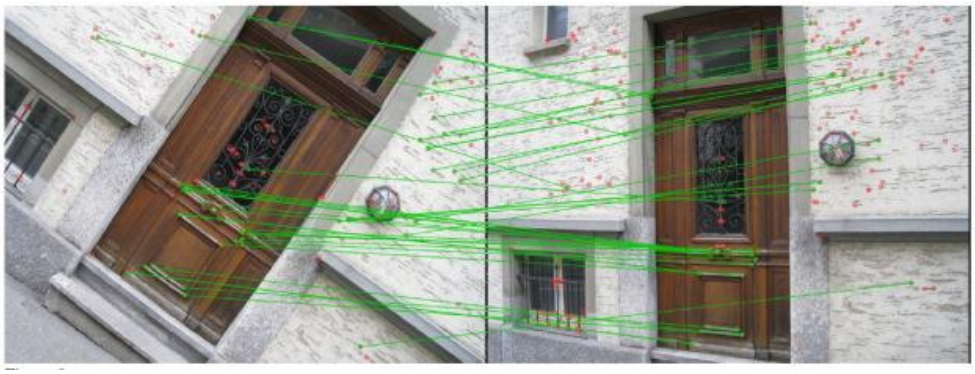


Figure 19. Exemple d'application du descripteur BRISK (Leutenegger et al. 2011).

Alahi et associés ont proposé le descripteur FREAK (Alahi et al. 2012) qui est similaire à BRISK en ayant un modèle d'échantillonnage et également similaire à ORB en utilisant des techniques d'apprentissage automatique pour sélectionner l'ensemble optimal de paires. FREAK possède également un mécanisme d'orientation qui est similaire à celui de BRISK.

FREAK a proposé une modélisation du pattern (Sampling Pattern) inspirée de la compréhension du modèle de la rétine humaine (Figure 20.c) (d'où vient son nom FREAK : Fast Retina Key-point) qui est circulaire avec une densité plus élevée de points près du centre (comme la distribution des cellules ganglionnaire sur la rétine, figure 20.b). La densité de points baisse de façon exponentielle pour les cercles extérieurs comme illustré dans la figure ci-dessous (Figure 20.a). Chaque point sélectionné est lissé avec un noyau gaussien où le rayon du cercle illustre la taille de l'écart-type du noyau.

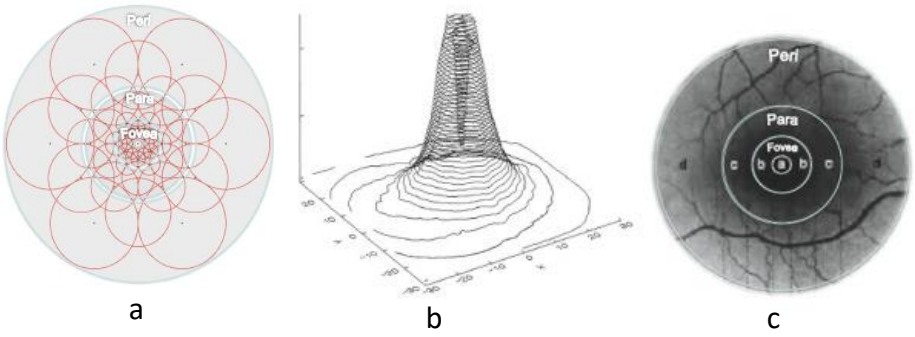


Figure 20. Le descripteur FREAK (Alahi et al. 2012). a) la modélisation du pattern. b) la distribution des cellules ganglionnaires sur la rétine. c) la rétine humaine.

Récemment, Nguyen et associés (Nguyen et al. 2016) ont proposé un détecteur- descripteur hybride appelé « Hessian ORB - Overlapped FREAK (HOOFR) ». Ce dernier hybride le détecteur d'ORB filtré par la matrice Hessienne avec une version modifiée du descripteur FREAK, dans laquelle la dimension du descripteur est de 256 bits au lieu de 512 bits pour la version originale de FREAK. Cette technique hybride a présenté un gain en mémoire. Cependant, la matrice Hessienne utilisée est couteuse en temps de calcul.

Yang et Cheng (Yang & Cheng 2014) ont proposé un descripteur binaire appelé Local Difference Binary (LDB). Ce dernier compare les intensités, ainsi que les orientations des gradients des sous-régions du patch et génère trois valeurs binaires à l'issue de chaque comparaison (voir figure 21). Les résultats de ces descripteurs soient comparables au ceux du descripteur original (SIFT). Cependant, ils sont couteux en temps de calcul, car ils doivent calculer à chaque fois les gradients de l'image.

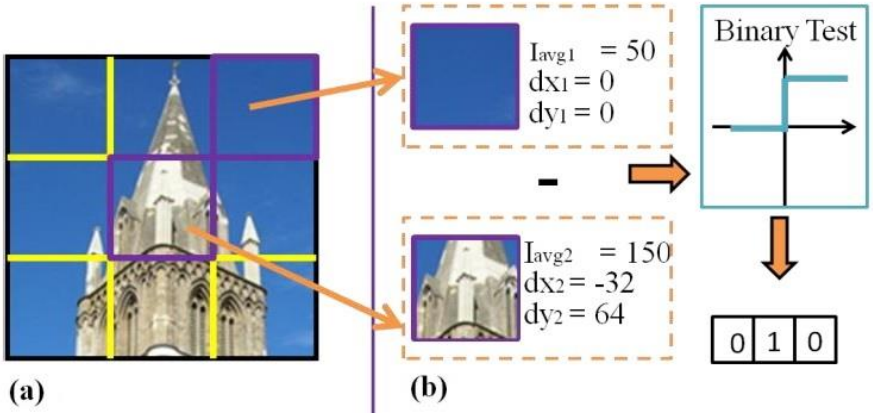


Figure 21. Le descripteur binaire LDB. a) La modélisation du pattern : division en 3x3 sous-régions. b) Tests binaires entre deux sous-régions. (Yang & Cheng 2014)

Levi et Hassner (Levi & Hassner 2016) ont introduit le descripteur binaire LATCH (Learned Arrangements of Three Patch Codes). LATCH compare la différence de l'intensité de trois sous-régions (appelées l'ancre et ses deux compagnons) dans le patch pour produire un seul bit du descripteur (figure 22). La comparaison de similitude entre la sous-région d'ancrage et les deux autres sous-régions est donnée par leur norme de Frobenius (Levi & Hassner 2016).

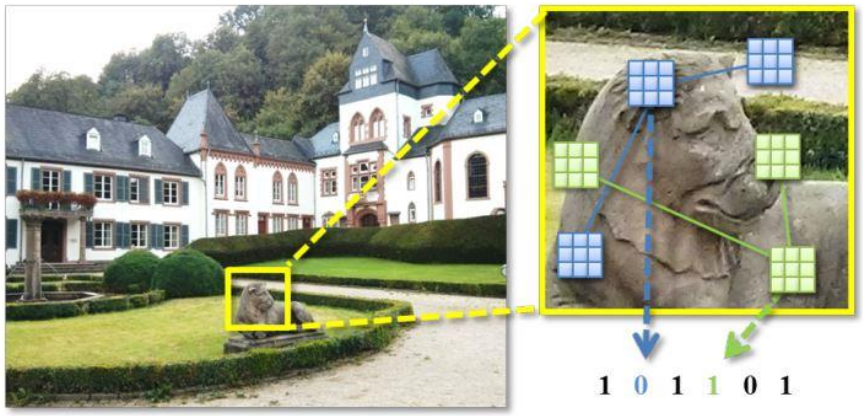


Figure 22. Le principe du descripteur binaire LATCH (Levi & Hassner 2016). LATCH compare la différence de l'intensité d'une sous-région avec deux autres régions.

Le descripteur LATCH offre ainsi, un taux de reconnaissance élevé. Cependant, il est plus lent que la plupart des descripteurs binaires de même taille. Parker et associés (Parker et al. 2016)

ont proposé récemment une version CUDA du descripteur binaire LATCH afin d'accélérer l'extraction du descripteur à l'aide des processeurs graphiques (GPU).

D'autres types de descripteurs binaires ont été également proposés. Leur principe est de binariser les descripteurs à base de virgule flottante comme SIFT et HOG, afin d'accélérer la mise en correspondance en utilisant la distance de Hamming. Zhou et associés (Zhou et al. 2015) ont proposé BSIFT (Binarized SIFT). Ce dernier binarise le vecteur de valeurs réelles de SIFT en comparant ces valeurs avec leur valeur médiane, ce qui produit un vecteur binaire déjà normalisé. Chen et Hsieh (Chen & Hsieh 2015) ont proposé une binarisation du descripteur SIFT en comparant successivement ses valeurs. Ainsi, lors de chaque test binaire on compare deux valeurs successives du vecteur de valeurs réelles de SIFT.

De nombreux efforts ont tenté également, d'encoder les descripteurs à base de virgule flottante en codes binaires en faisant appel à des techniques de hachage. La technique de hachage local « Local Sensitive Hashing (LSH) » (Gionis et al. 1999) et ses variantes (Bawa et al. 2005), (Andoni & Indyk 2006) sont fréquemment utilisées pour trouver des représentations binaires efficaces de vecteurs à virgule flottante de grande dimension en conservant leur similarité dans le nouvel espace, comme (Raginsky & Lazebnik 2009), (Heise et al. 2015) et (Kim & Choi 2015). Ces approches sont réalisées en multipliant d'abord des vecteurs de description par une matrice de projection, puis en établissant un seuillage des vecteurs des valeurs réelles à des chaines binaires.

De même, l'analyse discriminative linéaire LDA (LDA-HASH) (Strecha et al. 2012), K-means Hashing (He et al. 2013), Random Forest Hashing (Li et al. 2013), Bilinear Projections (Gong et al. 2013), Multi-linear Projections (Liu et al. 2016), Sparse Projection (Yan Xia et al. 2015) et Convex Optimisation (Simonyan et al. 2014) sont également utilisés pour binariser un descripteur à base de virgule flottante. Cependant, ces approches sont coûteuses du point de vue temps de calcul, parce que le descripteur en virgule flottante d'origine doit être calculé avant que le hachage puisse se produire. En outre, une dégradation des performances apparait à cause de la limitation des techniques de projection et de compression utilisées.

L'apprentissage profond (ou Deep Learning) est aussi présent dans les descripteurs binaires. Liong et associés (Liong et al. 2015) ont entrainé leur descripteur binaire grâce à un réseau de neurones profond (Deep Neural Network) afin de reduire sa dimension et augmenter sa distinction. Lin et associés (Lin et al. 2016) ont proposé DeepBit, un descripteur binaire dont les tests binaires ont été selectionnés par une approche d'apprentissage non-supervisé en utilisant un réseau de neurones convolutionnel (Convolutional Neural Network CNN). Les resultats obtenus sont convaincants. Cependant, les réseaux de neurones ne sont pas adéquats pour des applications en temps réel.

Nous pouvons trouver dans (Zeng et al. 2016), (Madeo & Bober 2016) et (Miksik & Mikolajczyk 2012) des comparaisons et des évaluations de ces descripteurs binaires présentés ainsi que d'autres descripteurs proposés.

1.6.3 Discussion

Afin d'avoir une idée sur le temps de calcul de certains descripteurs connus, nous avons calculé le temps moyen d'une description d'un patch pour chacun de ces descripteurs (Tableau 1). Ainsi, nous avons remarqué que la plupart de ces descripteurs ne sont pas adaptés pour des applications en temps réel (minimum 15 images par second), sauf pour BRISK et FREAK qui ont atteint respectivement ~27 et ~21 images/sec respectivement

Notons que la description est faite sur 500 points/image et le nombre d'images par second est calculé en fonction du temps de la description uniquement (sans rajouter le temps de la détection et de la mise en correspondance).

Tableau 1. Temps moyen de description d'un patch (ms) et le nombre d'images/sec.

Descripteurs	Temps de description d'un patch (ms)	Nombre d'images par second (avec description de 500 points par image)
SIFT	3.121	0.64
SURF	1.488	1.34
LDA-HASH	4.21	0.47
BRISK	0.072	27.77
FREAK	0.094	21.27
ORB	0.146	13.69
LDB	0.139	14.38
LATCH	0.437	4.57

En résumé, nous avons proposé une classification des descripteurs présentés avec leurs détecteurs (Tableau 2). Nous avons classifié ces descripteurs selon certains critères jugés nécessaires pour la mise en œuvre d'un système de réalité augmentée. Nous avons noté les descripteurs selon chaque critère, à savoir le temps de calcul, le taux de reconnaissance et l'espace mémoire, en utilisant une échelle de 1 à 5 (de + à +++++) montrée comme suit :

- Temps de calcul :
  - + : Très lent.
  - +++++ : Très rapide.
- Taux de reconnaissance :
  - + : Moins robuste.
  - +++++ : Robuste.
- Mémoire :
  - + : Volumineux.
  - +++++ : Léger.

Tableau 2. Classification des descripteurs.

Descripteur	Détecteur suggéré	Type	Temps de calcul	Taux de reconnaissance	Espace mémoire
SIFT (Lowe 2004)	DoG	Réel	+	+++++	++
PCA-SIFT (Ke & Sukthankar 2004)	DoG (SIFT)	Réel	++	++++	+++
GLOH (Mikolajczyk & Schmid 2005)	DoG (SIFT)	Réel	+	+++++	++
SURF (Bay et al. 2006)	Determinant of Hessian	Réel	+	++++	++
LDE (Hua et al. 2007)	DoG	Réel	++	+++++	+++
CONGAS (Buddemeier & Hartmut 2008)	LoG	Réel	++	++++	++
Daisy (Tola et al. 2010)	DoG (SIFT)	Réel	+	+++++	++
BRIEF (Calonder et al. 2010)	Détecteur de SURF	Binaire	+++	+++	++++
ORB (Rublee et al. 2011)	FAST	Binaire	++++	++++	++++
BRISK (Leutenegger et al. 2011)	AGAST+FAST	Binaire	+++++	++++	++++
FREAK (Alahi et al. 2012)	Détecteur de BRISK	Binaire	+++++	+++	++++
ALOHA (Saha & Démoulin 2012)	Détecteur de SURF	Binaire	+++	++++	+++
LDA-HASH (Strecha et al. 2012)	DoG (SIFT)	Binaire	+	+++++	+
KAZE (Alcantarilla et al. 2012)	Matrice Hessienne + Scharr filter	Réel	++	+++	+++
D-BRIEF (Trzcinski & Lepetit 2012)	DoG	Binaire	++	++++	++
BinBoost (Trzcinski et al. 2013)	DoG	Binaire	++	++++	++
A-KAZE (Alcantarilla et al. 2013)	Détecteur de KAZE	Binaire	++++	++++	+++
BRIGHT (Iwamoto et al. 2013)	DoG	Binaire	++	+++	++
LDB (Yang & Cheng 2014)	DoG (SIFT)	Binaire	+++	++++	+++
OSRI (Xu et al. 2014)	DoG (SIFT) / Hessian / Harris-Affine	Binaire	+++	++++	++++
BAMBOO (Baroffio et al. 2014)	Non spécifié	Binaire	+++	+++	+++
USB (Zhang et al. 2014)	DoG	Binaire	+++	++++	++++
PRO (Desai et al. 2014)	Multi-scale Harris	Binaire	+++	++++	++++
BSIFT (Zhou et al. 2015)	SIFT	Binaire	+++	++++	+++
BOLD (Balntas et al. 2015)	Haris-Laplace	Binaire	+++	++++	++++

Deep Hashing (Liong et al. 2015)	Image complète	Binaire	++	+++++	+++
DeepDesc (Simo-Serra et al. 2015)	DoG	Réel	+	+++++	++
SYBA (Desai et al. 2016)	Détecteur de SURF	Binaire	+++	++++	+++
LATCH (Levi & Hassner 2016)	Multi-scale Harris	Binaire	++	++++	+++
CUDA-LATCH (Parker et al. 2016)	FAST	Binaire	+++++	++++	+++
DeepBit (Lin et al. 2016)	Image complète	Binaire	++	+++++	+++
BDSB (Oszust 2016)	Determinant of Hessian	Binaire	+++	+++	++++
PN-Net (Balntas et al. 2016)	Harris-Affine	Réel	+	+++++	++
DELF (Noh et al. 2016)	CNN (convolutional neural network)	Réel	+	+++++	++
LIFT (Yi et al. 2016)	CNN (convolutional neural network)	Réel	++	+++++	+

Selon le tableau 2, nous avons constaté que les nouveaux descripteurs se basant sur l'apprentissage profond (Deep learning) comme DeepBit (Lin et al. 2016), DELF (Noh et al. 2016) ou encore LIFT (Yi et al. 2016) donnent de meilleurs résultats en termes de taux de reconnaissance. Cependant, leur inconvénient majeur est le temps de calcul. De même, les descripteurs classiques à base de virgule flottante, comme SIFT, GLOH, LDE ou DAISY montrent également des performances non négligeables en termes de taux de reconnaissance. Néanmoins, leur consommation de mémoire et de temps de calcul présente un obstacle pour une utilisation dans un système de réalité augmentée en temps réel.

En revanche, nous avons remarqué que malgré l'avantage majeur des descripteurs binaires pour la réduction du temps de calcul et de l'espace mémoire, leur robustesse et leur pouvoir discriminatif et distinctif sont considérablement limités par rapport à ceux des descripteurs à base de virgule flottante. Cela est dû au fait que la plupart de descripteurs binaires ne se basent que sur l'intensité des pixels de l'image.

Dans cette optique, nous allons introduire dans le troisième chapitre de ce présent manuscrit, notre nouveau descripteur binaire appelé MOBIL, qui compare les propriétés géométriques et statistiques des sous-régions du patch à savoir les moments géométriques. Afin de construire un vecteur binaire rapide avec une dimension réduite, et un pouvoir discriminatif et distinctif élevé. Nous allons détailler également l'utilisation de ce descripteur afin d'estimer la position de l'utilisateur dans un environnement augmenté.

1.7 Conclusion

Maintenir une cohérence visuelle dans l'espace et dans le temps entre le point de vue de l'utilisateur et son environnement augmenté est fortement lié aux problèmes d'estimation de pose et de vision par ordinateur.

A cet effet, nous avons tenté à travers ce chapitre de donner une vision globale du problème d'estimation de pose pour les applications de réalité augmentée. Nous nous sommes basés dans un premier temps sur l'aspect géométrique de l'estimation de pose en présentant les différentes méthodes qui permettent de répondre géométriquement à ce problème, puis nous avons abordé les approches qui intègrent l'estimation du mouvement. Nous avons classifié ces approches en fonction des informations disponibles : modèle 3D ou scène planaire.

Ensuite, nous avons présenté l'aspect extraction et description des caractéristiques qui est primordiale dans le processus d'estimation de pose. Nous avons constaté que la première génération de descripteurs (SIFT, SURF, …) a fait ses preuves en termes de robustesse. Cependant, l'inconvénient majeur de ces techniques est la lenteur d'exécution. D'autre part, l'apparition des descripteurs binaires au début des années 2010, a tenté de résoudre ce problème mais en diminuant la robustesse.

Un bon recalage des objets virtuels facilite l'immersion de l'utilisateur dans son environnement augmenté. En outre, vu que la qualité de l'extraction et de la description des caractéristiques visuelles influe énormément sur l'estimation de pose, nous présenterons dans le troisième chapitre nos contributions dans le but d'aboutir à une extraction et description fiable des caractéristiques visuelles (points d'intérêt).

D'autre part, l'interaction naturelle avec les objets virtuels est un aspect nécessaire qui contribue à l'immersion mobile de l'utilisateur en réalité augmentée. Celle-ci, constitue, la seconde partie de notre problématique, à savoir « *Comment interagir avec les objets virtuels tout en respectant l'homogénéité et la cohérence entre les deux mondes ?* »

Afin d'y répondre, nous allons aborder dans le chapitre suivant (deuxième chapitre) un état de l'art, sur les techniques et technologies d'interaction en réalité augmentée, qui nous permettra de faire le point sur l'existant.

Chapitre II.

Techniques et technologies d'interaction en RA

2.1 Introduction

L'immersion mobile en réalité augmentée est un concept qui a pour objectif d'offrir à l'utilisateur un environnement augmenté qu'il s'approprie facilement dans le sens où il interagit de manière naturelle avec les objets virtuels.

En RV et RA, la notion d'interaction est utilisée pour désigner un ensemble de règles et de techniques permettant à l'utilisateur d'accomplir des tâches d'interaction au sein d'un environnement virtuel ou augmenté (Otmane 2010). L'interaction 3D est donc, une composante motrice de la RV et de la RA, elle permet aux utilisateurs d'interagir avec les objets virtuels.

Le développement technologique ouvre de nouvelles portes à la RA. Avec le concept d'immersion mobile en réalité augmentée, la démarche d'aujourd'hui consiste à permettre à l'utilisateur en situation de mobilité d'interagir directement avec les objets virtuels. L'objectif étant d'offrir un moyen d'interaction à la fois naturel et intuitif, le contrôle d'application tend à se faire par l'un des canaux de communication du corps humain (voix, touché, geste…etc.).

Dans ce sens, la littérature est plus que riche dans le domaine de l'interaction et offre un ensemble de techniques qui permettent de répondre au besoin d'interaction avec des objets virtuels. Ce présent chapitre nous permet d'énumérer les différentes techniques d'interaction, notamment celles utilisées en réalité augmentée. Nous définissons donc les concepts liés à l'interaction (techniques, paradigme et métaphore d'interaction) ainsi que les modalités d'interaction naturelle et les différentes tâches d'interaction.

Nous abordons aussi la classification des techniques d'interaction 3D en fonction des tâches d'interaction à savoir les techniques de sélection et de manipulation et les techniques de contrôle d'application. Nous présentons aussi les techniques et technologies existantes d'acquisition et de reconnaissance de gestes de la main pour l'interaction en RA.

2.2 Définitions

2.2.1 Interface utilisateur

D'après Hix et Hartson (Hix & Hartson 1993), une interface consiste en le matériel et les logiciels qui assurent la médiation de l'interaction entre l'utilisateur et la machine et comprend également des dispositifs d'entrée et de sortie.

Fuchs et associés (Fuchs & Moreau 2006) ont défini l'interface comme un moyen qui permet à l'utilisateur de communiquer avec son environnement virtuel/augmenté. Ils ont déterminé également trois catégories d'interfaces : sensorielles, motrices et sensori-motrice. La première informe l'utilisateur par ses sens de l'évolution du monde virtuel/augmenté, la seconde informe l'ordinateur des actions motrices de l'utilisateur sur le monde virtuel/augmenté quant à la troisième elle jumelle les deux catégories précédentes.

2.2.2 Notion d'interaction

L'interaction est considérée comme étant un langage de communication entre l'homme et la machine correspondant à un ensemble d'actions/réactions réciproques entre l'homme et l'ordinateur par l'intermédiaire d'un certain ensemble d'interfaces (Ouramdane et al. 2009).

2.2.3 Technique, paradigme et métaphore d'interaction

Le besoin d'interagir avec la machine est de plus en plus présent, notamment dans le cas de la réalité mixte (RM : virtuelle/augmentée). Dans ce contexte particulier de la RM, une technique d'interaction désigne la méthode qui permet d'effectuer une tâche d'interaction dans un environnement virtuel/augmenté (Hachet 2003). Selon Ouramdane et associés (Ouramdane et al. 2009), elle peut être définie aussi comme le scénario qui utilise l'interface motrice d'une application donnée pour traduire les mouvements de l'utilisateur en actions dans le monde virtuel.

Quant à la notion de paradigme d'interaction, elle est utilisée pour désigner un ensemble de règles et de techniques permettant à l'utilisateur d'accomplir des tâches d'interaction au sein d'un environnement mixte (Mine 1995), (Poupyrev & Ichikawa 1999) et (Bowman 1999).

Pour ce qui est de la métaphore d'interaction, elle regroupe un ensemble de techniques d'interaction qui utilisent le même outil virtuel ou le même concept pour interagir avec les objets de l'espace (Sternberger 2006). Pour Fuchs et associés (Fuchs et al. 2003), une métaphore

d'interaction est une image symbolique d'une action ou d'une perception utilisée pour réaliser une tâche précise dans un environnement virtuel.

Otmane a proposé dans (Otmane 2010) de redéfinir l'interaction 3D par sa finalité. Ainsi, la finalité de l'interaction 3D est de permettre l'utilisation de dispositifs matériels et de techniques logicielles adaptées en vue d'une utilisation préformante et crédible des tâches d'interaction 3D.

2.3 Modalités d'interaction naturelle

Avec l'avènement des NUIs les modalités d'interaction ont pris un autre sens. Selon O'hara et associés (O'hara et al. 2013) le mot "Natural" dans NUI dénote à la fois l'aspect intuitif et facile à utiliser ou à apprendre de ces interfaces. Ce terme ne se réfère pas à l'interface elle-même et encore moins à sa technologie mais à la façon dont l'utilisateur interagit avec le système et la manière dont il le sent et l'utilise (Wigdor & Wixon 2011). Ainsi, plusieurs travaux se sont intéressés à l'exploitation des sens de l'homme dans la définition des techniques d'interaction naturelles notamment dans le contexte de la RA/RV.

Le geste de la main est la modalité la plus employée dans l'interaction naturelle. L'interaction directe par la main est considérée comme le moyen d'interaction le plus naturel étant donné que dans le monde réel, nous interagissons avec les objets physiques par la main.

L'interaction par la main peut se faire avec contact (cas des surfaces tactiles ou tangibles) ou sans contact avec la surface de visualisation. Dans le second cas on se base davantage sur les techniques de reconnaissance de gestes.

La voix, le regard, le mouvement du corps, l'expression faciale ou encore l'activité cérébrale sont également exploités comme moyens d'interaction naturels. Chacune de ces modalités présente des avantages selon le cas d'utilisation par exemple l'interaction par le regard peut être intéressante dans le cas d'applications destinées à des personnes atteintes de paralysie (Lim & Kim 2012). Djelil et associés présentent dans (Djelil et al. 2013) en détail les modalités d'interaction naturelles et leurs applications.

2.4 Tâches d'interaction

Dans les environnements mixtes (virtuels ou augmentées), l'interaction est primordiale. Nous pouvons concevoir ou imaginer des multitudes de tâches à réaliser pour des applications spécifiques de RA ou de RV, néanmoins considérer des tâches d'interaction de base est un moyen pratique qui permet de réaliser des tâches complexes dans toute application.

Bowman (Bowman 1999) a identifié ces tâches universelles et les a classifiées en quatre catégories : navigation, sélection, manipulation et contrôle d'application.

La navigation permet à l'utilisateur d'explorer, de rechercher et/ou de manœuvrer dans l'espace virtuel. Les composantes principales de la navigation sont le déplacement et la recherche d'itinéraire. Le déplacement représente la composante motrice de la navigation et consiste en le contrôle du mouvement du point de vue. C'est là où l'utilisateur positionne et oriente son point de vue dans l'environnement. La recherche d'itinéraire correspond à la composante cognitive de la navigation. Elle permet aux utilisateurs de déterminer un chemin d'accès basé sur des repères visuels, la connaissance de l'environnement et d'autres guides tels qu'une carte ou bien une boussole.

La sélection consiste à désigner de façon explicite un (des) objet (s) virtuel (s) afin d'effectuer une action donnée. Quant à la Cette tâche manipulation, elle consiste à modifier les attribues de l'objet sélectionné (position, orientation, l'échelle, couleur, texture).

Le contrôle de l'application englobe les autres commandes auxquelles l'utilisateur peut faire appel pour accomplir son travail dans l'application.

Pour chacune de ces tâches plusieurs techniques d'interactions ont été proposées. Ainsi nous pouvons interagir différemment dans une même tâche, par exemple l'utilisateur peut sélectionner de manière indirecte un objet virtuel en le choisissant sur une liste d'objets, comme il peut le sélectionner d'une manière directe en déplaçant sa main virtuelle de sorte à ce qu'il touche l'objet qu'il souhaite sélectionner (Bowman 1999).

De ce fait, nous présentons dans ce qui suit une classification des différentes techniques d'interaction.

2.5 Classification des techniques d'interaction 3D

Plusieurs techniques d'interaction ont été développées pour les environnements 3D. Celles-ci sont classifiées en général selon les tâches d'interaction. L'état de l'art étant riche en articles de synthèse qui récapitulent les différentes techniques d'interaction pour les environnements virtuels (Bowman et al. 2002) et (Jankowski & Hachet 2013), nous nous en inspirons dans ce qui suit pour présenter une classification adaptée à la réalité augmentée.

Dans les environnements virtuels, les techniques d'interaction sont classées par tâches d'interaction (navigation, sélection et manipulation, contrôle d'application). Les techniques d'interaction définies pour la tâche de navigation concernent en fait le déplacement, autrement

dit le contrôle du mouvement du point de vue de l'utilisateur, qui est l'un des fondements de base des environnements virtuels. Bowman (Bowman 1999) a défini le déplacement comme étant le contrôle du mouvement du point de vue de l'utilisateur dans un environnement 3D. Il proposa une taxonomie qui répartit les techniques de navigation en trois parties et/ou catégories selon le type de déplacement (recherche, exploration, manœuvre, etc.) :

1. Choix de la direction ou de la cible : Désigne les méthodes où la direction ou bien la cible visée par le déplacement est spécifiée.
2. Choix de la vitesse/accélération du mouvement : permet à l'utilisateur de varier sa vitesse de déplacement.
3. Choix des conditions d'entrée : Désigne les entrées requises par le système pour démarrer, continuer ou arrêter le déplacement.

En réalité augmentée, l'utilisateur perçoit en même temps son environnement réel et des objets virtuels insérés dans la scène. De ce fait, la navigation en RA se fait généralement de la même façon que l'utilisateur navigue dans sa vie quotidienne. Toutefois, son point de vue doit être calculé en permanence pour positionner les objets virtuels correctement dans la scène (voir chapitre 1).

De ce fait, nous présentons dans ce qui suit les techniques d'interaction pour la sélection et la manipulation, et les techniques qui permettent le contrôle d'application.

2.5.1 Techniques de sélection et de manipulation

Etant donné que l'utilisateur est dans un environnement augmenté, nous devons lui permettre d'interagir avec cet environnement. Dans ce sens, la sélection permet d'impliquer un ou plusieurs objets virtuels dans le processus de manipulation. Il existe une variété de techniques qui permettent à l'utilisateur de sélectionner et de manipuler des objets virtuels.

Bowman (Bowman 1999) a classifié ces techniques en trois catégories : extension du bras (Arm-extension), ray-casting et plan image (image plane). Les techniques d'extension du bras traitent le problème de la portée limitée de l'utilisateur et permettent à l'utilisateur d'étendre sa main virtuelle beaucoup plus loin que sa main physique. Les techniques ray-casting permettent de sélectionner des objets lointains en élargissant la métaphore du bureau 2D. Ainsi, on peut pointer un rayon de lumière virtuelle dans la scène et sélectionner un objet virtuel (Mine 1997). Enfin, les techniques plan-image sont une combinaison des interactions 2D et 3D. La sélection d'objet se fait sur un plan de vue (view plane) sans prendre en considération la profondeur par exemple l'utilisateur peut sélectionner un objet en l'occultant avec sa main virtuelle.

Il existe également la classification par décomposition en tâches (Ouramdane et al. 2009). Celle-ci se base sur un ensemble de blocs qui constituent une tâche de sélection ou de manipulation. Chaque bloc exécute une action élémentaire par le biais d'un certain nombre de ses composants. Cette approche permet donc de structurer l'espace de conception et de modélisation des techniques d'interaction afin qu'on puisse mettre en place de nouvelles techniques d'interaction en utilisant les composants déjà existants.

Une autre classification des techniques de sélection et de manipulation est la classification par métaphore (Poupyrev & Ichikawa 1999). En effet, la plupart des techniques d'interaction se basent sur des métaphores de base ou une combinaison de ces métaphores. Chaque métaphore constitue le modèle fondamental d'une technique d'interaction. Ainsi, les techniques de sélection et de manipulation ont été classifiées en deux grandes familles en fonction de la position de l'utilisateur et de la distance entre l'utilisateur et l'objet virtuel : les techniques exocentriques et les techniques egocentriques. Il existe également une troisième catégorie de techniques hybrides qui combine des techniques des deux catégories.

2.5.1.1 Les techniques exocentriques

Dans cette catégorie de techniques, l'utilisateur interagit avec le monde virtuel depuis l'extérieur et n'aura donc pas la sensation d'être totalement immergé dans l'environnement 3D. L'utilisateur est considéré comme un acteur qui ne fait pas partie de la scène, toutefois, il peut agir sur les entités virtuelles.

Dans ce contexte, une des premières métaphores est « le monde en miniature » ou « World In Miniature WIM ». Proposée par Stoakley et associés (Stoakley et al. 1995), celle-ci utilise une représentation miniature de l'environnement 3D pour permettre à l'utilisateur d'agir indirectement sur les objets virtuels (figure 23) . De ce fait, chacun des objets peut être sélectionné et/ou manipulé en utilisant la métaphore de la main virtuelle simple. L'utilisateur manipule indirectement les objets via une représentation miniature de l'environnement 3D. Ainsi, toute action sur un objet dans le monde en miniature provoque une action similaire sur l'objet du monde initial. L'inconvénient majeur de cette technique est la sélection et la manipulation des objets virtuels qui sont petits à l'origine.

Figure 23. La technique monde en miniature en RV (Argelaguet & Andujar 2013).

La figure suivante (figure 24) montre l'utilisation de cette technique dans une application de réalité augmentée pour la navigation en intérieur. Dans ce système, Mulloni et associés (Mulloni et al. 2012) ont utilisé des marqueurs (cibles) placés sur le sol d'un centre commercial et augmentés par un modèle 3D de l'immeuble en miniature. L'utilisateur peut sélectionner à travers l'écran du smartphone son point de destination.

Figure 24. Application de réalité augmentée avec monde en miniature pour la navigation en intérieur (Mulloni et al. 2012).

Une autre technique exocentrique, nommée « Scaled-World grab », est dotée d'un principe simple : à la sélection d'un objet, l'échelle de l'utilisateur ou du monde 3D est augmentée de façon à ce que la main de l'utilisateur touche l'objet sélectionné (Mine et al. 1995).

Ces techniques ont été proposées pour les environnements virtuels. En ce qui concerne leur utilisation en réalité augmentée, elles sont inappropriées vue que le recalage des objets virtuels ne peut être assuré durant la tâche d'interaction.

2.5.1.2 Les techniques égocentriques

Contrairement à la catégorie précédente, les techniques egocentriques correspondent beaucoup mieux aux applications de réalité augmentée. Cela est dû au fait que dans ce genre de techniques, l'utilisateur fait partie de l'environnement où il est considéré comme une composante de ce dernier. Ces techniques sont représentées en trois types : main virtuelle, pointeur virtuel et plan image (Figure 25).

Figure 25. Exemples de techniques égocentriques en RA, gauche : main virtuelle (Ha et al. 2014), droite: pointeur virtuel (Oda & Feiner 2012).

2.5.1.2.1 Main virtuelle

L'idée est de permettre à l'utilisateur d'utiliser sa propre main pour sélectionner des objets virtuels. En effet, il s'agit de la métaphore de la main virtuelle (Virtual hand) proposée par Jacoby et associés (Jacoby et al. 1994). Cette technique est la plus naturelle et intuitive, l'utilisateur touche l'objet virtuel avec sa main pour le désigner et confirme la sélection en fermant le poignet ou en restant en contact avec l'objet un certain temps. Les mouvements de la main de l'utilisateur sont transmis à l'avatar pour manipuler l'objet (figure 26).

Bien que naturelle et intuitive, cette technique pose problème pour les objets distants, l'utilisateur doit se déplacer vers ces objets jusqu'à ce qu'ils soient à sa portée.

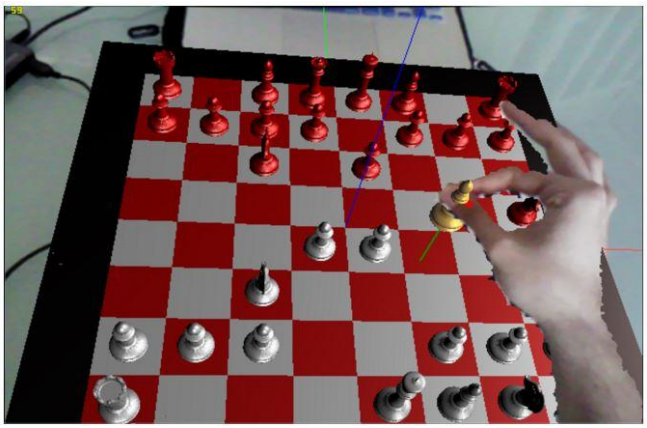


Figure 26. Exemple d'interaction par la main en RA, en utilisant la métaphore Virtual Hand (Bikos et al. 2016).

2.5.1.2.2 La technique Go-Go

La technique Go-Go vient palier à ce problème. Proposée par (Poupyrev & Billinghurst 1996), cette technique appelée également technique d'extension du bras, permet la sélection et la manipulation d'objets éloignés, ceci grâce à une relation non-linéaire entre la main virtuelle et la main réelle. En d'autres termes, le déplacement dans le monde réel se traduit par un grand déplacement dans le monde virtuel après un certain seuil.

La technique Go-Go présente un inconvénient quant à la sélection d'objets petits et distants. Les mouvements de faible amplitude de la main réelle sont traduits par des mouvements de grande amplitude par la main virtuelle. Une extension de la technique Go-Go a été proposée toujours par Poupyrev et Billinghurst appelée Stretch Go-Go (Poupyrev and Billinghurst 1996). Dans cette technique, le bras virtuel peut s'agrandir ou rétrécir selon sa position dans le monde virtuel. Ainsi, lorsque la main est dans une zone proche, celle-ci s'approche de l'objet avec une vitesse constante. Dans le cas où on est dans une zone intermédiaire, la main est au repos, et en zone distante, le bras s'allonge à une vitesse constante. Une autre variante, qui est le Fast Go-Go (Bowman 1999) où la position du bras est calculée par une fonction non linéaire comme pour la technique Go-Go, sauf que la valeur du seuil est égale à zéro, ce qui fait que le bras s'allonge rapidement dès le début du mouvement.

2.5.1.2.3 Pointeur virtuel

Les techniques basées sur la métaphore du pointeur virtuel, utilisent aussi une représentation virtuelle de la main, sauf que l'utilisateur sélectionne un objet à l'aide d'un pointeur laser, sans toucher l'objet par la main. La plupart de ces techniques sont des techniques de sélection.

Une des premières techniques du genre est le Ray-Casting, introduite par Bolt dans (Bolt 1980) puis reprise par plusieurs auteurs (Zhai et al. 1994), (AMICIS R et al. 2001). Cette technique de pointage est basée sur la métaphore du rayon virtuel. Un rayon laser infini part de la main virtuelle et traverse tout le monde virtuel. Le premier objet en intersection avec le rayon laser peut être sélectionné. Bien que très pratique, le ray-casting pose problème lorsqu'il s'agit d'objets petits ou lointains.

Pour y remédier Liang et associés (Liang & Green 1994) proposent d'utiliser un cône à la place du rayon pour pouvoir sélectionner les objets distants ou petits. Forsberg et associés

(Forsberg et al. 1996) proposent que l'angle d'ouverture du cône soit variable en fonction de la distance des objets à sélectionner.

Olwal et Feiner (Olwal & Feiner 2003) proposent un pointeur flexible qui répond au problème d'obstacles entre l'objet à sélectionner et la main virtuelle. Cette technique permet de pointer plus facilement des objets occultés par d'autres objets dans la scène. En effet, le rayon peut être dirigé avec une certaine courbure et une longueur contrôlées à l'aide des deux mains.

Oda & Feiner ont proposé la technique GARDEN (pour Gesturing in an Augmented Reality Depth-mapped ENvironment) (Oda & Feiner 2012). Cette dernière appliquée à la réalité augmentée, se base sur la métaphore du pointeur virtuel (figure 27). La technique GARDEN permet de sélectionner des objets virtuels distants en les pointant par le doigt, ou encore d'indiquer un objet virtuel pour un autre utilisateur. Cependant, cette technique n'assure pas une sélection précise quand les objets sont très loin, ou très petits.

Figure 27. Exemple de la technique GARDEN (Oda & Feiner 2012).

2.5.1.2.4 Les techniques plan-image

La sélection et la manipulation d'un objet 3D s'effectuent à travers sa projection sur un plan image 2D. Pierce et associés (Pierce et al. 1997) présentent un ensemble de techniques plan image, à savoir, la technique paume à plat (Lifting Palm), ou l'utilisateur met la paume de sa main devant lui, en dessous de l'objet à sélectionner afin d'indiquer les objets qu'il souhaite sélectionner. La technique entre deux doigts (Head crusher), où la sélection s'effectue en encadrant l'objet virtuel grâce à un geste du pouce et de l'index de la main réelle. La technique dirigée du doigt (Sticky Finger) se base sur un rayon virtuel qui part de la tête de l'utilisateur passant par l'index de sa main pour sélectionner l'objet à manipuler. Pour finir, la technique de la main encadrante (Framing Hands), où la sélection s'effectue en encadrant par les deux mains réelles le ou les objets.

2.5.1.3 Les techniques hybrides

Les techniques hybrides combinent au moins deux techniques egocentriques et/ou exocentriques, en voici quelques-unes.

2.5.1.3.1 La technique HOMER

La technique HOMER (pour Hand-centered Object Manipulation Extending Ray-casting) (Bowman 1999) combine le Ray-Casting pour la sélection d'un objet et la technique de la main virtuelle simple pour la manipulation. Un objet sélectionné est attaché à la main et hérite de toutes les transformations que la main subit. La main virtuelle reprend sa position initiale dès que l'objet est relâché (figure 28).

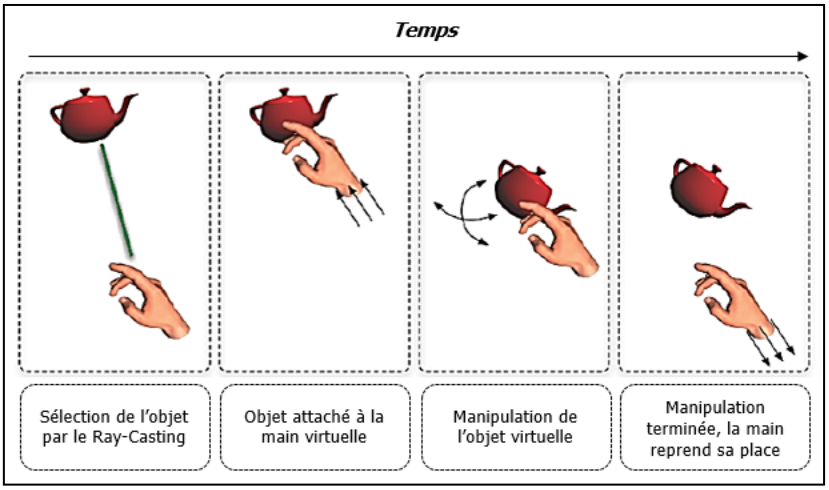


Figure 28. Technique HOMER (Bowman 1999), figure reproduite.

Mossel et associés ont proposé la technique HOMER-S (Mossel et al. 2013), une amélioration de la technique HOMER. La technique HOMER-S permet à l'utilisateur d'interagir avec des objets virtuels en réalité augmentée à travers l'écran de son smartphone (figure 29). Cependant, l'utilisation d'une interface tactile (interaction en 2D) réduit la capacité d'interaction avec des objets virtuels en 3D.

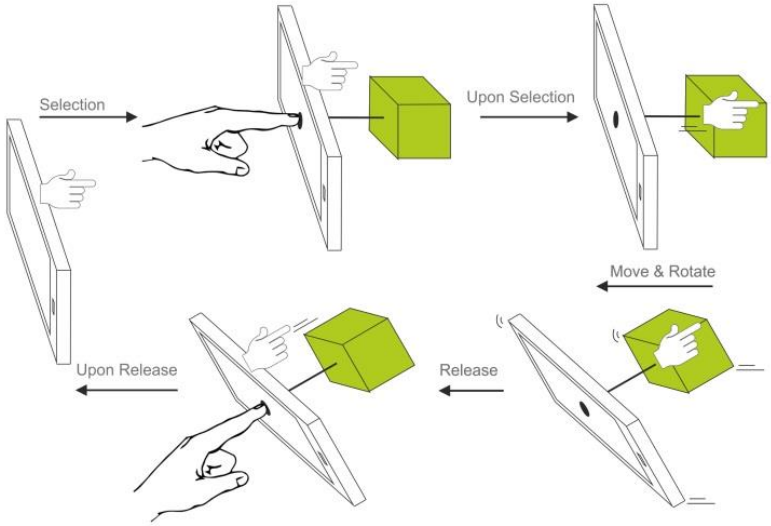


Figure 29. Le principe de la technique HOMER-S, (Mossel et al. 2013).

2.5.1.3.2 La technique Voodoo Dolls

C'est une technique hybride qui combine la technique WIM et une des techniques plans-images. La technique des « Poupées Vaudou » a été proposée par Pierce et associés (Pierce et al. 1999). Celle-ci permet à l'utilisateur de créer des poupées qui sont une représentation miniature des objets virtuels existants dans l'environnement 3D. La sélection des objets à manipuler se fait par la technique « head crusher », à partir de là, on crée une maquette miniature de l'objet sélectionné et de son environnement proche, la manipulation peut alors se faire en utilisant la main virtuelle simple.

2.5.1.4 Discussion

Nous présentons ci-dessous une classification des techniques d'interaction dédiées à la sélection et à la manipulation (Tableau 3). Ainsi, nous avons classifié ces techniques selon leur précision et leur charge cognitive. Nous avons attribué une note pour chaque critère en utilisant une échelle de 1 à 5 (de + à +++++) comme suit :

- Précision :
  - + : Moins précise.
  - +++++ : Très précise.
- Charge cognitive :
  - + : Charge cognitive forte.
  - +++++ : Charge cognitive faible.

Tableau 3. Classification des techniques de sélection et de manipulation.

Classes	**Métaphore**	**Techniques**	**Littératures**	**Précision**	**Charge cognitive**
Techniques Exocentriques		Monde en miniature	(Stoakley et al. 1995)	+ + + + +	+
		Scaled world grab	(Mine et al. 1995)	+ + +	+ +
	Main Virtuelle	Main virtuelle simple	(Jacoby et al. 1994)	+ + + +	+ + + + +
		Go-Go	(Poupyrev et Billinghurst 1996)	+	+ + +
		Stretch Go-Go	(Poupyrev et Billinghurst 1996)	+	+ + +
		Fast Go-Go	(Bowman 1999)	+	+ + +
Techniques Egocentriques	Pointeur Virtuel	Ray-Casting	(Bolt 1980)	+ +	+ + + +
		Cône	(Liang & Green 1994)	+ + +	+ + + +
		Cône-variable	(Forsberg et al. 1996)	+ + +	+ + + +
		Pointeur flexible	(Olwal & Feiner 2003)	+	+ +
		Direction du regard	(Tanriverdi & Jacob 2000)	+ +	+ + + +
		GARDEN	(Oda & Feiner 2012)	+ +	+ + + +
	Plan Image	Paume à plat	(Pierce et al. 1997)	+	+ +
		Sticky Finger	(Pierce et al. 1997)	+	+ + +
		Head crusher	(Pierce et al. 1997)	+	+ + +
		Framing hands	(Pierce et al. 1997)	+	+ + +
Techniques Hybrides		Homer	(Bowman 1999)	+	+ + +
		Homer-S	(Mossel et al. 2013)	+	+ + +
		Voodoo Dolls	(Pierce et al. 1999)	+ + +	+ +

2.5.2 Les techniques de contrôle d'application

Le contrôle d'application regroupe toutes les techniques de manipulation indirectes sur l'environnement 3D. Le contrôle d'application se situe à un niveau conceptuel diffèrent des autres tâches d'interaction, dans le sens où l'utilisateur agit sur l'application en utilisant les services offerts par l'application elle-même.

Plusieurs techniques de contrôle d'application ont été conçues, nous pouvons regrouper ces techniques comme suit.

2.5.2.1 Menu graphique 3D

C'est l'équivalent 3D des menus 2D. Ils peuvent prendre la forme d'un menu 2D transformé qui sera positionné dans le monde virtuel (on utilisera alors une technique de sélection/manipulation 3D pour manipuler les boutons du menu) (Bellarbi et al. 2014a) (figure 30). Ils peuvent également prendre la forme d'un menu 3D.

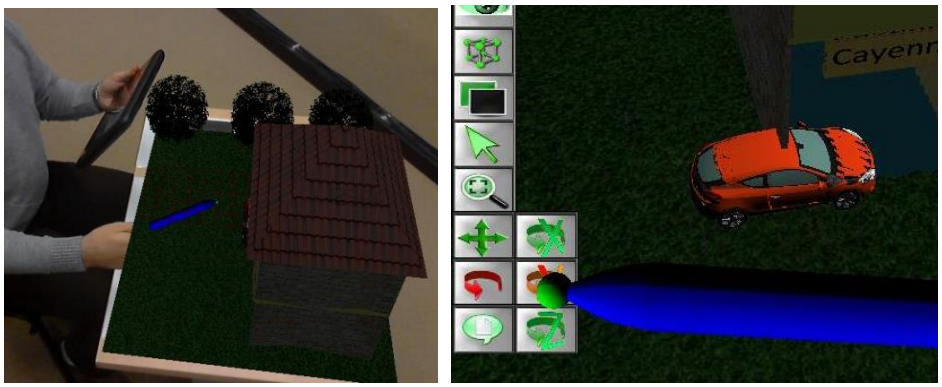


Figure 30. Exemple d'un menu 3D dans une application de RA (Bellarbi et al. 2014a).

2.5.2.2 Commande vocale

Des mots et/ou des phrases sont associés à des commandes de sélection et de manipulation. Ainsi grâce à la reconnaissance vocale on peut contrôler une application par la voix.

2.5.2.3 Commande gestuelle

Des commandes de contrôle sont associées à des gestes bien précis, on peut donc exécuter des commandes à travers des gestes de la main.

2.5.2.4 Autres outils de contrôle d'application

Le contrôle d'application peut se faire par d'autres outils virtuels ou réels avec lesquels l'utilisateur peut agir sur l'environnement par exemple l'utilisation d'un stylet ou d'un joystick.

Nous avons abordé dans ce second chapitre les différentes techniques et modalités d'interaction. Il est à noter que la modalité d'interaction la plus utilisée se base sur les gestes de la main. La plupart des techniques d'interaction existantes portent sur l'utilisation des gestes de la main comme outils d'interaction. De ce fait, nous abordons dans ce qui suit les différentes techniques de reconnaissance de gestes et des technologies existantes facilitant la reconnaissance de gestes.

2.6 Techniques et technologies de reconnaissance de gestes de la main pour l'interaction en RA

Les mouvements de nos mains jouent un rôle important lors d'une communication entre les gens. On utilise la main pour pointer une personne ou un objet, donner une information sur l'espace ou la forme. On utilise aussi la main pour interagir avec un objet : le déplacer, le transformer ou le modifier. D'où l'idée de l'utilisation de la main pour interagir avec la machine. La reconnaissance des gestes en général, et des gestes de la main en particulier, peut aider les gens à communiquer avec les ordinateurs d'une façon plus facile et naturelle (Rautaray & Agrawal 2012).

Le but ultime de la communauté des chercheurs du domaine des interfaces naturelles de l'utilisateur (NUI) est d'arriver à une communication naturelle et intuitive entre l'Homme et la Machine. La conception d'une telle interface requiert un système de reconnaissance de geste très précis pour faciliter l'interaction Homme-Machine .

Pour étudier les systèmes de reconnaissance de gestes, nous allons commencer par définir et classer les gestes. Nous poursuivons avec les différentes techniques de reconnaissance de gestes, ainsi que les technologies existantes permettant l'acquisition de geste (O'hara et al. 2013).

2.6.1 Définition et représentation de geste

Un geste est un signe manuel ou corporel qui permet d'illustrer les mots d'un langage, de les compléter ou de les appuyer. Ainsi, les gestes contrôlés ou incontrôlés, que nous faisons en parlant font partie du message : ils ponctuent la parole, la soulignent ou la renforcent. Ils peuvent aussi se substituer à la parole ou exprimer des idées, des sentiments ou des intentions. Ils transmettent l'information, parfois mieux que les mots, et sont généralement dotés d'une sémantique (Rautaray & Agrawal 2012).

Selon le dictionnaire Larousse[1], un geste est « un mouvement du corps, principalement de la main, des bras, de la tête, porteur ou non de signification ».

Martin et Durand ont considéré que le geste englobe tous les mouvements de la main permettant de communiquer des informations significatives et pertinentes. (Martin & Durand 2000).

La définition effective du geste est relative au domaine d'application, ou encore à son utilisation. Nous nous intéressons à l'interaction en réalité virtuelle et augmentée, de ce fait, nous définissons un geste comme étant un moyen à exploiter afin d'interagir avec les objets dans un environnement virtuel/augmenté.

2.6.2 Techniques d'acquisition et de reconnaissance de gestes

Plusieurs études ont été menées sur la reconnaissance des gestes et plusieurs classifications y ont été proposées selon le type de la technologie d'acquisition utilisée. Nous nous intéressons à la classification de La Viola (LaViola 1999) qui divise la reconnaissance des gestes en deux approches : les approches basées capteurs et les approches basées vision.

2.6.2.1 Approches basées capteurs

Ces approches s'appuient sur l'utilisation de certains dispositifs, et regroupent les capteurs magnétiques, acoustiques, inertiels et haptiques (mécaniques), ainsi que les surfaces tactiles.

2.6.2.1.1 Capteurs magnétiques

Les capteurs magnétiques utilisent le champ magnétique à basse fréquence émis par un émetteur pour un récepteur afin de déterminer sa position et son orientation par rapport à la source magnétique. Le principal inconvénient est la distorsion du champ magnétique que les métaux produisent.

2.6.2.1.2 Capteurs acoustiques

Les capteurs acoustiques convertissent les ondes sonores en signal électrique. Ces capteurs s'avèrent relativement peu coûteux, légers et sans interférence avec les métaux. Cependant, le majeur problème est la sensibilité au bruit.

[1] Larousse, http://www.larousse.fr/dictionnaires/francais, 2016.

2.6.2.1.3 Les capteurs Inertiels

Cette technologie permet de calculer la rotation sur les trois axes, en se basant sur la gravité terrestre et les mouvements de l'utilisateur. Récemment, ce type de capteurs est intégré dans la plupart des smartphones, tablettes et autres consoles et dispositifs. Dans (Katzakis et al. 2015), Katzakis et associés, ont utilisé un smartphone doté d'un capteur inertiel pour manipuler des objets 3D dans une plateforme de réalité virtuelle (figure 31).

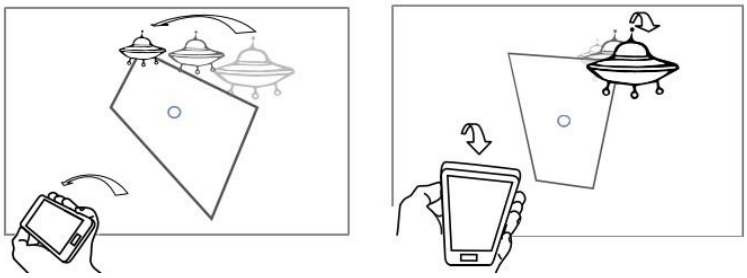


Figure 31. Manipulation d'un objet 3D sur un écran en utilisant le capteur inertiel d'un smartphone (Katzakis et al. 2015).

2.6.2.1.4 Dispositifs haptiques

Cette technologie permet d'interagir physiquement avec un objet virtuel. Ce système restitue à l'utilisateur la perception du toucher, dans le cas d'un écran tactile, et la sensation de déplacement 3D dans l'espace en utilisant un dispositif à retour de force.

Apple[1] a proposé récemment un écran tactile qui détecte le degré de pression des doigts. Cette technologie appelée « 3DTouch » présente sur « iPhone 6S » permet d'effectuer une interaction 3D sur un écran 2D (figure 32).

Figure 32. Le principe de la technologie 3DTouch d'Apple[1]

[1] Apple 3D Touch. http://www.apple.com/fr/iphone-6s/3d-touch/. 2016.

2.6.2.1.5 Nouveaux types de capteurs

Récemment, l'équipe de recherche ATAP[1] de Google ont proposé un nouveau capteur pour la détection des gestes de la main, appelé « Soli » (Lien et al. 2016). Il s'agit d'une puce radar miniature de 9 mm de large, qui fonctionne dans le spectre radio à 60 GHz et qui envoie des ondes radio. Lorsque celles-ci rebondissent sur la main et les doigts, Le radar peut en déduire leur position dans l'espace à l'échelle du millimètre, avec une fréquence équivalente à 10 000 images par seconde (figure 33).

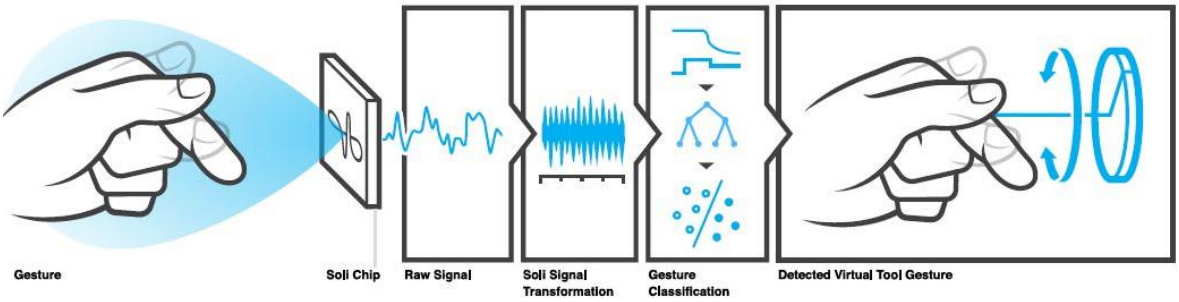


Figure 33. Le principe de fonctionnement du capteur Soli, (Lien et al. 2016).

Ce projet Soli vise en effet à permettre aux utilisateurs d'interagir naturellement et intuitivement avec différents objets connectés comme les smartwatchs et les smartphones en utilisant des mouvements de doigts ou de mains, et sans toucher l'appareil (Wang et al. 2016), (figure 34).

Figure 34. Interaction avec des appareils intelligents par le capteur Soli, (Lien et al. 2016).

Dans le même contexte, Du et associés, ont proposé dans (Du et al. 2016) un prototype d'un écran sans contact appelé « Airtouch » en se basant sur des capteurs capacitifs (figure 35) afin de déterminer la position du doigt en voisinage de l'écran, ce qui permet à l'utilisateur d'interagir en 3D avec les objets virtuels. Cependant, le mouvement des doigts en profondeur est très limité.

[1] Google ATAP, https://atap.google.com/, 2016.

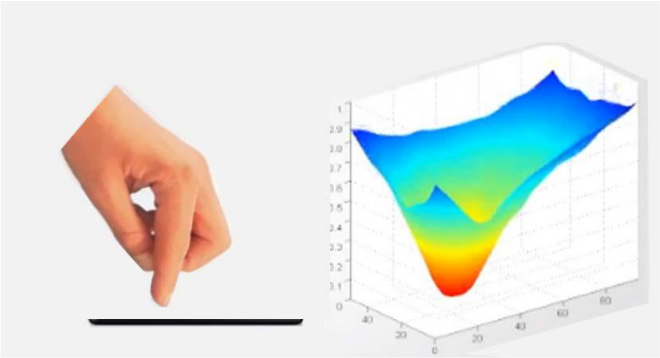


Figure 35. Le concept de Airtouch montrant l'espace de détection de la main (Du et al. 2016).

2.6.2.2 Approches basées vision

Les approches basées vision utilisent le flux vidéo capturé par des caméras pour reconnaitre les gestes. Ces approches se basent sur la constitution des apparences possibles d'un geste sous différents points de vue et différentes conditions (Martin & Durand 2000).

Ainsi, plusieurs représentations ont été proposées afin de reconnaitre et modéliser un geste de la main. Selon (Rautaray & Agrawal 2012), deux grandes catégories de représentation des gestes de la main existent : les méthodes basées sur des modèles 3D et les méthodes basées sur l'apparence, comme illustré dans la figure (36) ci-dessous.

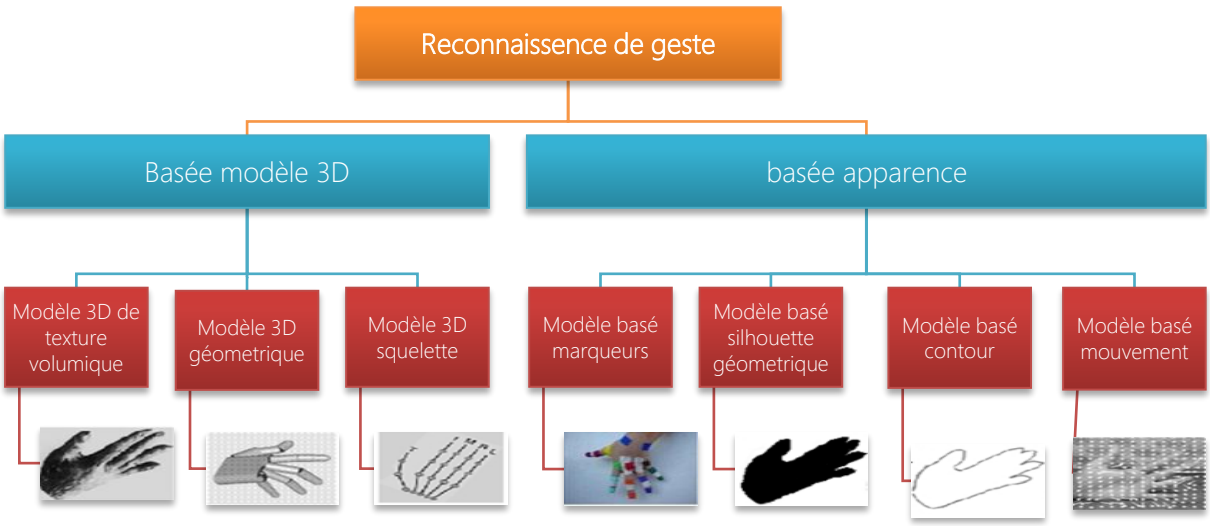


Figure 36. Représentation des gestes de la main, adaptée de (Rautaray & Agrawal 2012).

2.6.2.2.1 Techniques basées apparence

Certaines approches ont été proposées, dont le principe est de porter sur les mains des marqueurs faciles à reconnaitre (gants, couleurs, cibles codées…) afin d'interagir avec le système

(Benbelkacem et al. 2012) . Dans (Wang & Popović 2009), une technique de reconnaissance de geste de la main en 3D en portant des gants colorés a été proposée. Les gants colorés sont modélisés et construits de sorte qu'ils facilitent la reconnaissance des gestes dynamiques de la main par une caméra (RGB) afin d'offrir à l'utilisateur une interaction naturelle avec l'environnement virtuel (figure 37). Bien que les résultats montrés par les auteurs soient convaincants, cependant, ce système est conçu pour reconnaitre que cette forme de gants.

Figure 37. La technique de gants colorés proposée dans (Wang and Popović 2009).

Nous avons proposé dans (Bellarbi et al. 2011) de porter des petits marqueurs de couleur sur les doigts des mains de l'utilisateur (figure 38) afin de manipuler des documents numériques projetés sur une table. Les marqueurs sont fabriqués avec des morceaux de papiers colorés pour ne pas encombrer l'utilisateur. Cependant, cette technique permet de reconnaitre un nombre limité de gestes statiques pour une interaction en 2D, et présente certaines anomalies lors de la confusion des couleurs portées avec celles de la scène.

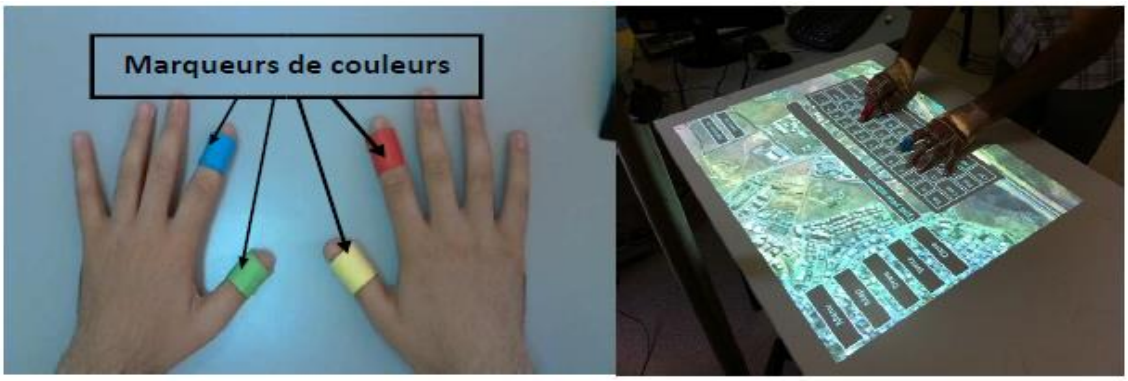


Figure 38. Technique d'interaction basée marqueurs couleurs (Bellarbi et al. 2011).

En revanche, de nombreuses approches ont été proposées afin d'interagir naturellement avec la machine en utilisant la main sans avoir besoin de porter un marqueur ou un capteur sur la main de l'utilisateur et ceci en traitant des images capturées par la caméra.

Certaines approches ont considéré le problème de la reconnaissance de gestes comme un problème de classification en faisant appel à des techniques de classification.

Dans ce sens, les réseaux de neurones ont été largement utilisés pour la reconnaissance des gestes (Murakami & Taguchi 1991), (Stergiopoulou & Papamarkos 2009) et (Hasan & Abdul-Kareem 2014). De leur côté, les chaines de Markov cachées (HMM) ont été également exploitées pour la classification des gestes. Nous pouvons citer (Lee & Kim 1999), (Jinjun et al. 2009) et (Fahn & Chu 2011). Cependant, ces approches ne garantissent pas des applications en temps réel.

De même, certaines approches ont été proposées pour la reconnaissance de gestes, utilisant les moments invariants (Liu et al. 2012), les moments de Krawtchouk (Priyal & Bora 2013), Support Vector Machine SVM (Chen & Tseng 2007), Gabor Filter (Huang et al. 2011)(figure 39), Dynamic Time Warpping DTW (Keskin et al. 2011), Template Matching (Yun et al. 2012) ou encore Randomized Decision Forests (Keskin et al. 2012), (Zhao et al. 2012).

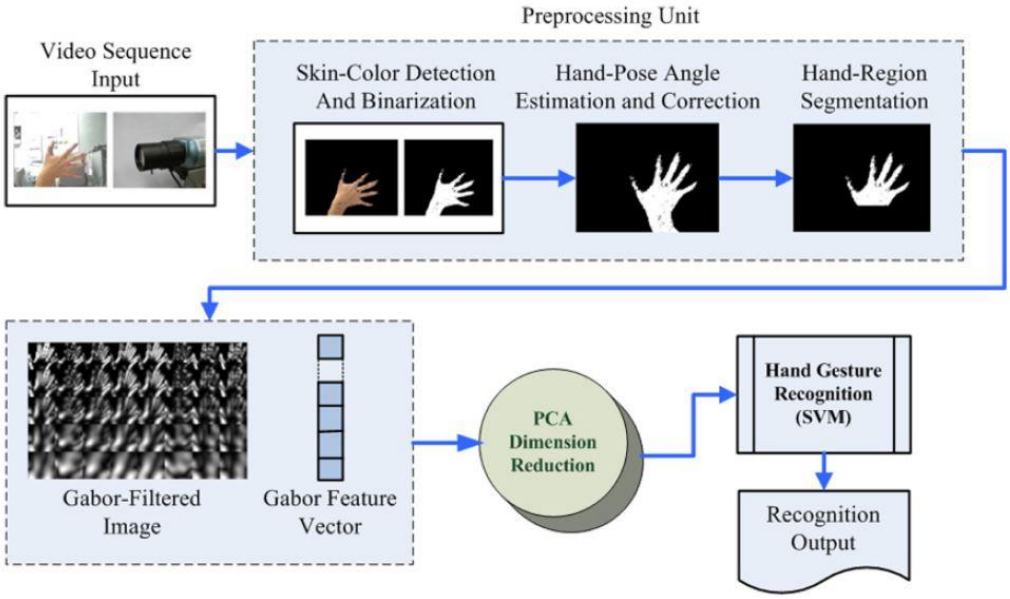


Figure 39. Diagramme du flux de la technique de reconnaissance de geste par filtre de Gabor proposée dans (Huang et al. 2011).

D'autres approches ont appliqué des descripteurs des points d'intérêt sur les images capturées, pour reconnaitre les gestes de la main, tels que SIFT (Lin et al. 2013), SURF (Yao & Li 2013), LBP (Ding et al. 2011).

Nous avons proposé dans (Bellarbi et al. 2013b) une technique de détection et de reconnaissance de gestes 2D de la main en utilisant une simple webcam. La technique proposée, se base d'une part sur la binarisation adaptative et le calcul de l'histogramme de l'image pour la détection et l'extraction de la main, et d'autre part sur l'algorithme de Chain Code avec une version modifiée de la technique ASM (Approximate String Matching) pour la reconnaissance de gestes. Cette technique a été utilisée afin d'interagir en 2D avec une table interactive (voir figure 40).

Figure 40. La table interactive proposée dans (Bellarbi et al. 2013b).

Ce système a été utilisé par la suite comme une interface d'expert lors du projet de réalité augmentée « Remote Gesture » qui a pour objectif de guider un technicien distant par les gestes d'un expert local (figure 41) (Zenati-Henda et al. 2014) (Benbelkacem et al. 2015). (Ce travail est détaillé en Annexe, voir Annexe B.2.).

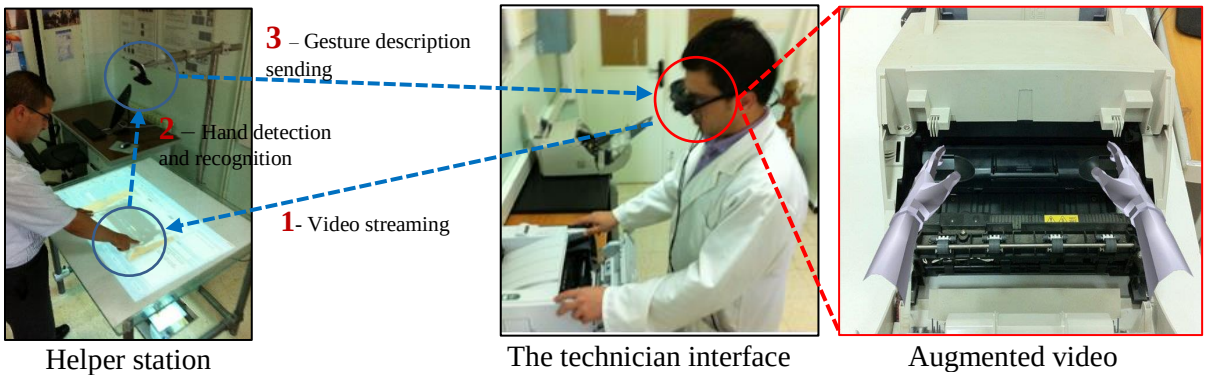


Figure 41. Principe du projet « Remote Gestures ». (Zenati-Henda et al. 2014).

Ces techniques proposées ont été utilisées dans de nombreuses applications telles que le langage des signes, ou encore l'interaction avec des machines (robots, tables interactives …). Cependant, elles ne sont pas adaptées à l'interaction 3D en réalité virtuelle/augmentée.

En revanche, l'émergence des caméras de profondeur a donné naissance à de nouvelles approches plus robustes, offrant une reconnaissance de gestes en 3D.

2.6.2.2.2 Techniques basées modèle 3D

En exploitant les avantages qu'offrent les caméras de profondeur, telles que Kinect de Microsoft, et Xtion de Asus (voir figure 42), plusieurs approches de reconnaissance de gestes en 3D ont été proposées.

Figure 42. Quelques exemples de caméras de profondeurs. *a*) Creative Caméra[1]. b) Xtion Pro[2]. c) Zed[3] caméra. d) Kinect. e) Kinect 2 [4].

Wang et associés (Wang et al. 2011) ont proposé une technique appelée 3Gear. L'idée de cette technique se base sur la recherche dans une base de données des gestes afin de trouver le geste le plus similaire au geste capturé par la Kinect (figure 43).

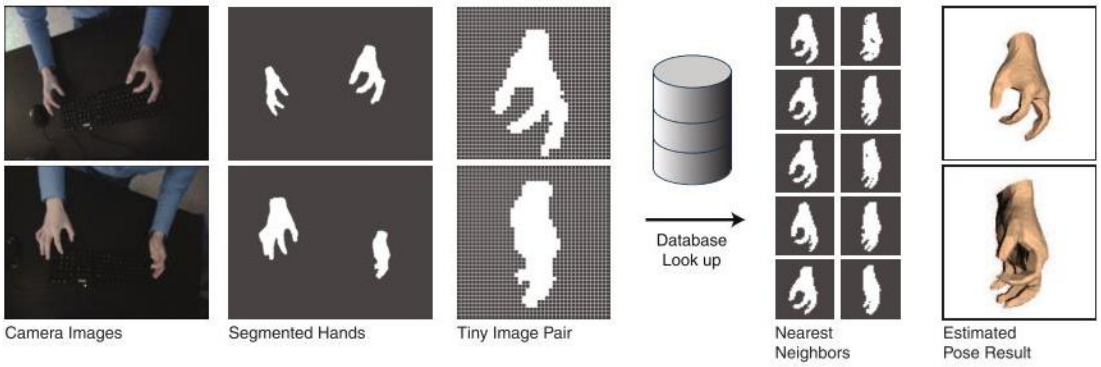


Figure 43. Le principe de la technique 3Gear (Wang et al. 2011).

La base de données de gestes générée hors-ligne contient des images issues de la projection de différents gestes possibles à partir d'un modèle 3D de la main. Cette technique présente une reconnaissance robuste des gestes 3D de la main.

[1] http://fr.creative.com/p/web-caméras/creative-senz3d
[2] https://www.asus.com/3D-Sensor/Xtion_PRO/
[3] https://www.stereolabs.com/
[4] https://developer.microsoft.com/fr-fr/windows/kinect/

Messaci et associés (Messaci et al. 2015) ont utilisé cette technique afin d'interagir dans un environnement virtuel. L'objectif étant de former des techniciens dans le domaine de la maintenance (figure 44).

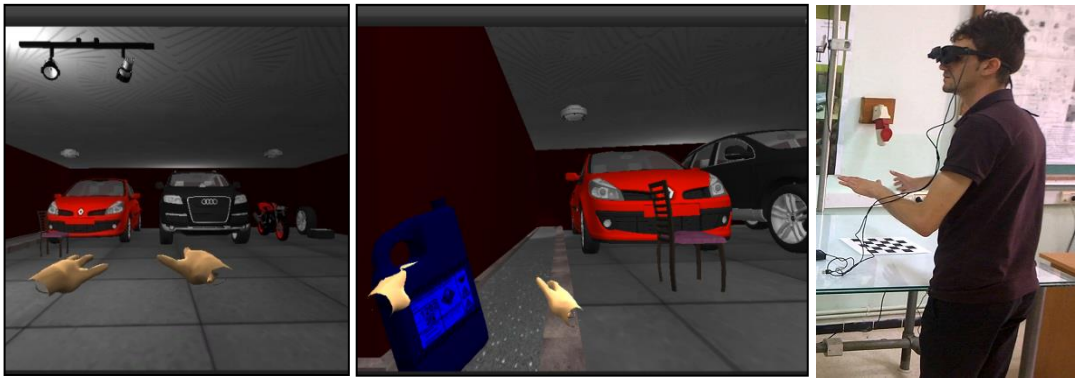


Figure 44. Exemple d'un utilisateur qui évolue dans un environnement virtuel, utilisant le système proposé dans (Messaci et al. 2015).

Avec une idée presque similaire, Khamis et associés (Khamis et al. 2015) ont proposé un algorithme d'apprentissage automatique pour une recherche efficace dans une base de données des images de profondeur des gestes. Cette technique a présenté des résultats meilleurs. Cependant, elle est plus couteuse en temps de calcul.

Certains chercheurs ont modélisé la main selon des points d'articulation. Ils ont appliqué par la suite une fonction de coût, pour minimiser l'erreur entre le modèle 3D et la forme de la main réelle, afin d'estimer la position et le geste 3D de la main (figure 45). Nous pouvons citer les travaux (Taylor et al. 2016), (Qian et al. 2014) et (Melax et al. 2013).

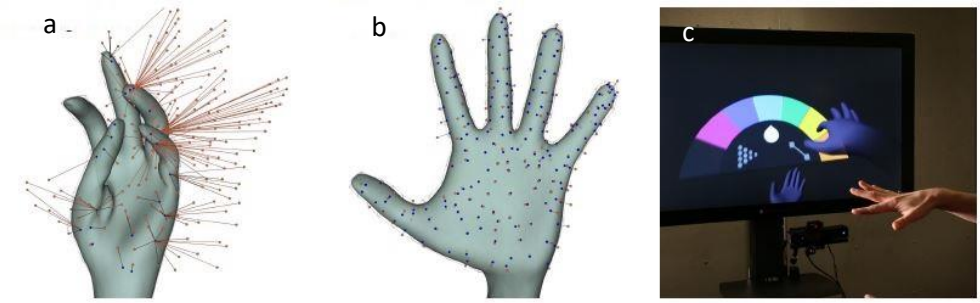


Figure 45. Technique de reconnaissance et de suivi de geste de la main proposée dans (Taylor et al. 2016). a) Initialisation : modèle estimé de geste. b) Convergence vers le geste réel en appliquant une fonction d'optimisation de l'énergie. c) Exemple d'application en RV.

Récemment, les techniques de reconnaissance de gestes 3D ont pris une autre dimension en termes de robustesse et de précision avec l'évolution des techniques d'apprentissage et l'apparition de l'apprentissage profond (Deep Learning). Nous pouvons citer dans ce sens les

travaux (Wu et al. 2016), (Tompson et al. 2014), (Oberweger et al. 2015), (Tang et al. 2015) et (Farabet et al. 2013).

Ces techniques montrent une efficacité et une robustesse pour la reconnaissance de gestes 3D de la main. Néanmoins, les dispositifs de capture utilisés actuellement nécessitent une position fixe de l'utilisateur, et ne lui offrent pas une mobilité totale. Des études comparatives récentes sur les différentes techniques de reconnaissance de gestes 3D de la main peuvent être trouvées dans (Cheng et al. 2016) et (Supancic III et al. 2015).

En Octobre 2012, Holz et associés (Holz et al. 2012) ont développé le « Leap Motion », un nouveau dispositif de reconnaissance et de suivi de gestes des mains, composé d'une partie soft et d'un contrôleur (matériel). Ce dernier qui est de taille 13 mm x 30 mm x 76 mm, est doté de deux caméras infrarouge (figure 46) et peut détecter et reconnaitre les mains dans un espace qui peut aller jusqu'à 50 cm.

Figure 46. Le dispositif Leap Motion.

Ce nouveau dispositif, Leap Motion, est devenu très répandu en RA et/ou RV. Son efficacité, sa mobilité, et son prix abordable le rendent accessible au grand public. Ainsi, beaucoup de travaux ont été menés en utilisant le Leap Motion, nous citons (Bacim et al. 2014), (Kim & Lee 2016), (Caggianese et al. 2016) et (Davis et al. 2016).

Depuis juillet 2016, l'entreprise Intel a dévoilé sa nouvelle technologie RealSense[1] qui consiste en un dispositif compact composé de trois types de capteurs (une caméra RGB de résolution 1080p, une caméra infrarouge et un projecteur laser infrarouge) (Figure 47).

[1] RealSense Intel, http://click.intel.com/realsense.html 2016.

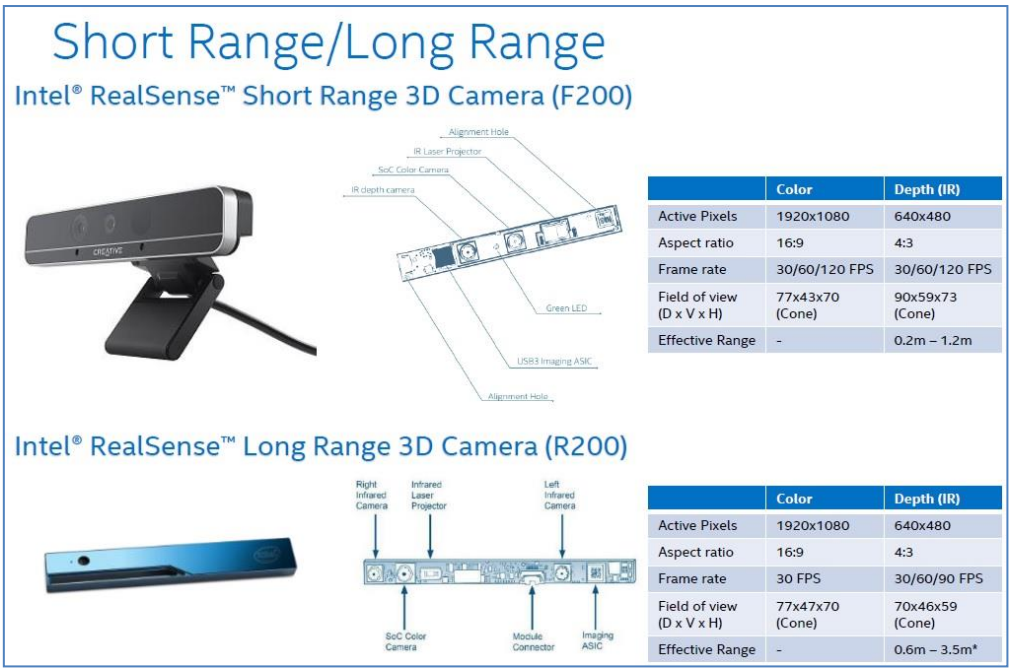


Figure 47. Les nouvelles caméras RealSense de Intel[1].

Cette hybridation de capteurs permet de reconnaitre et suivre les mouvements et les gestes de l'utilisateur d'une façon robuste (selon le constructeur Intel). Ainsi, le SDK fourni avec le système permet de reconnaitre un ensemble de gestes de la main (figure 48).

Figure 48. Ensemble de gestes prédéfinis par le système RealSense[1].

[1] RealSense Intel, http://click.intel.com/realsense.html 2016.

2.6.2.3 Discussion

Dans cette partie du chapitre, nous avons abordé les différentes approches et techniques permettant la reconnaissance des gestes de la main de l'utilisateur ainsi que les différents types de capteurs et de technologies utilisés. Ainsi, nous présentons une classification de quelques techniques de reconnaissance de gestes proposées dans la littérature. Nous avons classifié ces techniques selon le type de capteur utilisé, la forme du geste reconnu ainsi que le domaine d'application destiné (voir Tableau 4).

Tableau 4. Classification des techniques de reconnaissance de gestes.

Techniques	2D/3D	Type de capteur	Dynamique/Statique	Application
(Murakami & Taguchi 1991)	2D	RGB webcam	Statique	Langue des signes
(Jinjun et al. 2009)	2D	RGB webcam	Statique	Langue des signes
(Stergiopoulou & Papamarkos 2009)	2D	RGB webcam	Statique	Langue des signes
(Wang & Popović 2009)	3D	RGB webcam	Dynamique	Réalité virtuelle
(Bellarbi et al. 2011)	2D	RGB webcam	Statique	Table interactive
(Fahn & Chu 2011)	2D	RGB webcam	Statique	Interaction Robot
(Huang et al. 2011)	2D	RGB webcam	Statique	Langue des signes
(Keskin et al. 2011)	2D	RGB webcam	Statique	Langue des signes
(Várkonyi-Kóczy & Tusor 2011)	2D	RGB webcam	Statique	Langue des signes
(Reale et al. 2011)	3D	Stereo webcam	Dynamique	Réalité virtuelle
(Wang et al. 2011)	3D	Kinect	Dynamique	Réalité virtuelle
(Liu et al. 2012)	2D	RGB webcam	Dynamique	Réalité virtuelle
(Holz et al. 2012)	3D	Leap motion	Dynamique	Réalité virtuelle
(Radkowski & Stritzke 2012)	3D	Kinect	Dynamique	Réalité augmentée
(Bellarbi et al. 2013b)	2D	RGB webcam	Dynamique	Table interactive
(Melax et al. 2013)	3D	Kinect	Dynamique	Réalité virtuelle
(Potter et al. 2013)	3D	Leap motion	Statique	Langue des signes
(Bacim et al. 2014)	3D	Leap motion	Dynamique	Réalité virtuelle
(Qian et al. 2014)	3D	Kinect	Dynamique	Réalité virtuelle
(Khamis et al. 2015)	3D	Kinect V2	Dynamique	Réalité virtuelle
(Oberweger et al. 2015)	3D	Kinect	Statique	-
(Jinwook et al. 2016)	2D	Kinect	Dynamique	Réalité augmentée

(Bikos et al. 2016)	3D	RealSense	Statique	Réalité Augmentée
(Kim & Lee 2016)	3D	Leap motion	Dynamique	Réalité augmentée
(Wu et al. 2016)	3D	Kinect	Dynamique	Langue des signes
(Taylor et al. 2016)	3D	Kinect V2	Dynamique	Réalité virtuelle

Selon le tableau 4, nous constatons qu'avec l'émergence des nouveaux capteurs de profondeur tel que la Kinect, les caméras de RealSense ou encore le Leap Motion, la reconnaissance de gestes en 3D et devenue possible. En outre, l'utilisation de ce type de capteurs offre de meilleurs résultats en termes de stabilité et de précision. Dans la suite de notre travail, nous allons utiliser le Leap Motion avec son SDK, afin de mettre en œuvre notre proposition.

2.7 Conclusion

Tout au long de ce chapitre, nous avons abordé l'interaction 3D en réalité augmentée, en présentant dans un premier temps les concepts fondamentaux et les définitions de base qui régissent cette technologie. Nous avons également abordé les tâches élémentaires de l'interaction qui permettent de réaliser des tâches complexes. Nous avons aussi présenté une classification des techniques d'interaction 3D selon les tâches de l'interaction, ainsi que les modalités d'interaction naturelles notamment le geste de la main. Nous nous sommes penchés par la suite sur les techniques et technologies existantes permettant la reconnaissance des gestes de la (les) main (s).

A partir de cet état de l'art, nous avons constaté que la majorité des techniques d'interaction existantes concerne beaucoup plus le domaine de la réalité virtuelle (RV) et ne sont pas forcément adaptées à la RA. Nous pouvons citer notamment les techniques d'interaction (navigation, sélection, manipulation) avec les objets virtuels distants où la plupart de ces techniques ne garantissent pas le recalage des objets virtuels en RA.

En outre, l'étude menée dans ce chapitre, sur les différentes techniques et technologies de reconnaissance de gestes de la main, nous a permis de voir les possibilités en matières d'équipements et de techniques de reconnaissance de gestes pouvant nous être utiles pour la suite de notre travail.

Nous présentons dans ce qui suit un bilan détaillé de cette première partie du mémoire qui nous permet de dégager nos contributions relatives à la problématique définie dans le cadre de cette thèse.

Bilan

Les recherches dans le domaine de la RA ont plus de 50 ans (Marchand et al. 2016) (Billinghurst et al. 2015) (Zhou et al. 2008). La signification du terme « réalité augmentée » a beaucoup évolué dans le temps. Nous sommes passés d'une définition restreinte qui limite la réalité augmentée à un système qui utilise un dispositif de visualisation tête-haute semi-transparents (See-through HMD) à une définition plus large qui caractérise la RA par l'amélioration de la perception de l'utilisateur.

Ainsi, la réalité augmentée représente tout système qui permet la fusion du réel et du virtuel en offrant à l'utilisateur une perception augmentée de la réalité. Celle-ci peut s'étendre à tous les sens de l'utilisateur (le visuel, le toucher ou encore l'ouïe). Cette fusion peut lier sémantiquement le réel et le virtuel, et peut prendre en compte le recalage des deux mondes réel et virtuel, le choix se fait en fonction de l'application et de son objectif. Aussi avec les technologies émergentes, différents dispositifs peuvent être envisagés pour l'estimation de pose, la visualisation (tablettes, casques,…) ou encore pour l'interaction. Ainsi, tous ces choix potentiels notamment pour la visualisation ont un impact certain sur le degré d'immersion de l'utilisateur en réalité augmentée.

Aujourd'hui, cette technologie commence à se démocratiser auprès du grand public, avec les efforts de plusieurs entreprises, la RA fait son entrée dans le monde de l'industrie et du commerce, et commence à connaitre un grand succès auprès du public notamment des amateurs de jeux avec la sortie du jeu Ingress de Google ou encore le phénomène PokémonGo. Ceci a donné lieu à des questions qui relèvent de l'acceptation sociale (Billinghurst et al. 2015), (Sandor et al. 2015). Toutefois, les axes de recherche sont encore ouverts et attendent d'être d'avantage explorés avant que cette technologie atteigne son plein potentiel. En effet, des obstacles scientifiques et technologiques subsistent encore avec des limitations dans les systèmes d'estimation de pose et d'interaction.

En ce qui concerne la partie estimation de pose en RA, beaucoup de travaux ont été menés dans ce sens. Nous avons présenté durant le premier chapitre de ce manuscrit, un état de l'art sur les différentes techniques et approches qui permettent de répondre aux problèmes d'estimation de pose pour les applications de réalité augmentée (extraction et description des primitives, suivi, estimation de pose). De ce fait, nous avons constaté que la qualité de l'extraction et de la description des caractéristiques visuelles en termes de robustesse et temps de calcul influe énormément sur le maintien de la cohérence spatio-temporelle. Ainsi, les techniques existantes n'ont pas encore atteint cet objectif.

Pour ce qui est de l'interaction 3D, nous avons constaté que la recherche dans ce domaine a pris un autre sens par rapport à ses débuts. En effet, la plupart des techniques et des métaphores d'interaction 3D ont été proposées durant les années 90. C'est pendant cette période cruciale que les fondements de base et les classifications des techniques d'interaction ont été instaurés (Mine 1995), (Bowman 1999), (Poupyrev et al. 1998). L'évolution de la technologie des interfaces naturelles utilisateur (Natural User Interface, NUI) a donné un nouvel élan au domaine de l'interaction 3D avec plus de possibilités et de moyens, d'interagir de manière naturelle (Lien et al. 2016).

Aujourd'hui, l'interaction 3D ne manque pas d'intérêt. Les recherches dans ce domaine sont en pleine effervescence. Ainsi, l'utilisation des techniques d'interaction 3D avec les nouvelles technologies des NUI peut mener à des applications réelles et concrètes.

A partir du deuxième chapitre de ce mémoire, nous avons relevé que les travaux existants sur l'interaction 3D, sont beaucoup plus axés sur le domaine de la réalité virtuelle (RV) comparé à la RA. En effet, la majorité des techniques d'interaction ont été proposées dans le contexte de la RV. Bien que certaines de ces techniques soient applicables en RA, néanmoins, le fait d'être lié à la scène réelle donne certaines particularités aux systèmes de RA, ce qui rend intéressant de se pencher sur des techniques d'interactions destinées à la RA et qui répondent au concept de l'immersion mobile.

Dans cette thèse, nous souhaitons aborder le concept d'immersion mobile en réalité augmentée. Celui-ci désigne tout système de réalité augmentée où l'utilisateur se sent immergé dans un mélange physique/virtuel comme s'il s'agissait d'un seul monde, avec la possibilité de se déplacer et d'interagir en toute liberté dans son environnement augmenté tout en conservant une cohérence sensorielle (visuelle, auditive, haptique).

Dans cette optique, nous présentons dans la seconde partie de ce mémoire nos contributions scientifiques dans ce domaine, qui concerne les deux parties essentielles d'un système de RA à savoir le recalage du virtuel par rapport au réel et l'interaction 3D vers une immersion mobile de l'utilisateur dans un environnement augmenté.

PARTIE II. Contributions

Vers l'immersion mobile en réalité augmentée

Chapitre III.

MOBIL : un nouveau détecteur et descripteur

3.1 Introduction

L'estimation de la pose 3D consiste à déterminer la transformation qui permet d'obtenir à partir d'un objet contenu dans l'image 2D, la position 3D de l'objet correspondant.

Concrètement, cela demande l'extraction et la description de caractéristiques visuelles dans l'image pour procéder à l'estimation de cette transformation. Dans ce sens, les descripteurs binaires sont très souvent sollicités. Ils présentent l'avantage d'être rapides, néanmoins, leur robustesse peut toujours être améliorée.

Nous présentons à travers ce chapitre, nos contributions dans la partie vision pour la RA, à savoir notre nouvelle technique de détection et description appelée MOBIL. En effet, nous commençons par présenter le nouveau détecteur de points d'intérêt (MOBIL_Detector). Ce dernier hybride le détecteur AGAST avec la mesure de coins de Shi-Tomasi. Des tests et des comparaisons du détecteur proposé avec d'autres détecteurs connus sont également présentés.

Nous introduisons par la suite, le descripteur binaire MOBIL (MOments-based BInary differences for Local description) (Bellarbi et al. 2014b). Ce dernier se base sur la comparaison binaire des moments géométriques autour du point d'intérêt (le patch), et présente une robustesse contre les différents types de transformations et changements d'intensité afin d'assurer une augmentation stable et précise des scènes réelles.

Deux améliorations apportées au descripteur MOBIL sont présentées par la suite. A savoir MOBIL_2B (Bellarbi et al. 2015) et POLAR_MOBIL (Bellarbi et al. 2017a), dans le but d'améliorer sa robustesse face aux transformations affines et aux changements de point de vue. Des tests des descripteurs proposés et des comparaisons avec des descripteurs de l'état de l'art sont présentés par la suite.

Ces descripteurs sont utilisés durant la deuxième partie de ce chapitre afin de mettre en œuvre le système de réalité augmentée mobile. En effet, durant la tâche de la navigation, nous avons utilisé la technique PTAM (Klein & Murray 2007) afin d'assurer le recalage réel/virtuel en offrant ainsi à l'utilisateur la possibilité de se déplacer dans son environnement augmenté. Pendant la tâche de sélection et de manipulation, nous utilisons notre nouveau « détecteur-descripteur MOBIL » qui assure une insertion stable des objets virtuels.

3.2 Approche proposée

En général, les descripteurs binaires utilisent une seule propriété de l'image telle que la différence d'intensité entre les sous-régions du patch, ou bien les directions de gradient afin de construire le vecteur descriptif. Par conséquent, La description de l'image est insuffisante ; en outre, construire une description à grande dimension augmente la robustesse du descripteur, cependant cela génère des descriptions redondantes et corrélées, et augmente la complexité de calcul.

Pour éviter de tels inconvénients, il est préférable de créer une description multicritères (i.e. qui se base sur plus d'une propriété), et de trouver un compromis entre la dimension de la description et la robustesse du descripteur.

Nos contributions dans cette partie du travail peuvent être décrites comme suit :

- Nous proposons une technique améliorée pour la détection de points d'intérêt appelée « MOBIL_Detector », basée sur le détecteur AGAST (Adaptive and Generic Accelerated Segment Test) et raffinée par la métrique de Shi-Tomasi pour la mesure de coins.
- Nous introduisons un nouveau descripteur binaire MOBIL (ainsi que ses deux améliorations MOBIL_2B, et POLAR_MOBIL), dans lequel nous calculons la différence entre des propriétés géométriques des sous-régions dans le patch.
- Nous appliquons une technique de sélection de bits pour choisir les meilleurs tests binaires qui conduisent à la fois à une faible corrélation et une forte distinction.

3.2.1 MOBIL_Detector : un nouveau détecteur basé sur AGAST et Shi-Tomasi.

Afin de remédier aux différents problèmes des détecteurs classiques (voir chapitre 1, section 1.5.2), nous avons proposé une technique de détection basée sur le détecteur AGAST, raffiné par la mesure de coins de Shi-Tomasi (Shi and Tomasi 1994).

A cet effet, nous avons construit une pyramide à espace d'échelle de neuf niveaux avec un facteur de racine carrée de deux ($\sqrt{2}$) entre chaque deux niveaux successifs afin d'atteindre l'invariance contre les changements d'échelle. Nous avons ensuite appliqué le détecteur AGAST 5-8 pour chaque niveau afin d'en extraire le maximum des coins à haute vitesse (figure 49).

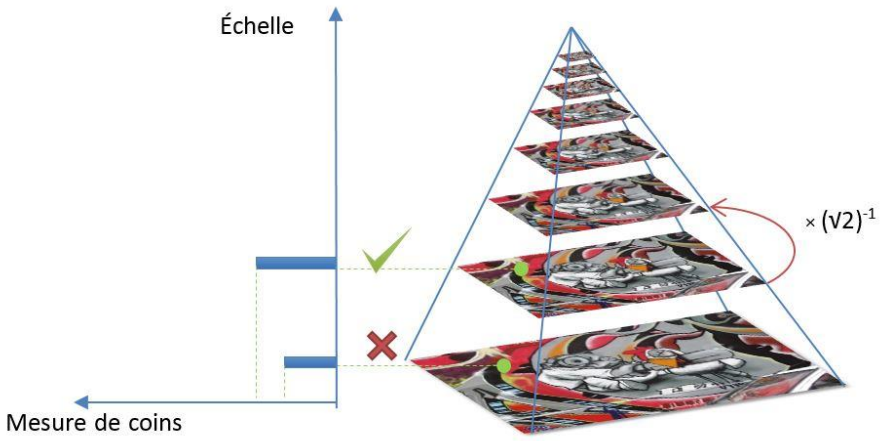


Figure 49. La pyramide à espace d'échelle utilisée dans le détecteur des points

Par la suite, les coins extraits sont raffinés afin de sélectionner ceux qui sont plus stables. Pour chaque point détecté, nous avons calculé la mesure de Shi-Tomasi (Equation. 8) afin de vérifier si le point extrait est un coin, sous un seuil prédéfini.

$$R = min\ (\lambda_1, \lambda_2) \tag{8}$$

λ1 et λ2 sont des valeurs propres définies par le détecteur de coins Harris. Nous avons comparé les points extraits de chaque niveau de la pyramide afin d'éliminer les points redondants. Pour ce faire, nous avons appliqué un algorithme d'optimisation pour comparer les positions des points extraits entre chaque deux niveaux adjacents, et nous avons supprimé les points redondants ayant le score R minimum.

3.2.1.1 Test et évaluation du détecteur MOBIL_Detector

3.2.1.1.1 Plateforme de test et d'évaluation

Afin de tester et d'évaluer nos propositions, nous avons utilisé l'environnement de développement Visual Studio 2013, avec Open CV 2.4.4 sur un processeur i3 3.20GHz Intel® Core™.

Pour l'étape d'évaluation, nous avons utilisé le benchmark de Mikolajczyk et Schmid (Mikolajczyk & Schmid 2005). Ce dernier est devenu une référence standard pour l'évaluation des détecteurs et des descripteurs locaux d'images. Cette base d'images est composée de huit ensembles d'images (voir la figure 50), chacun contenant six images qui représentent un degré croissant d'une transformation relative à une image spécifiée :

- Bark : zoom + rotation.
- Bikes : le flou.
- Boat : zoom + rotation.
- Graffiti : changements de point de vue.

- Leuven : changements d'éclairage.
- Trees : le flou.
- UBC : compression JPEG
- Wall : changements de point de vue.

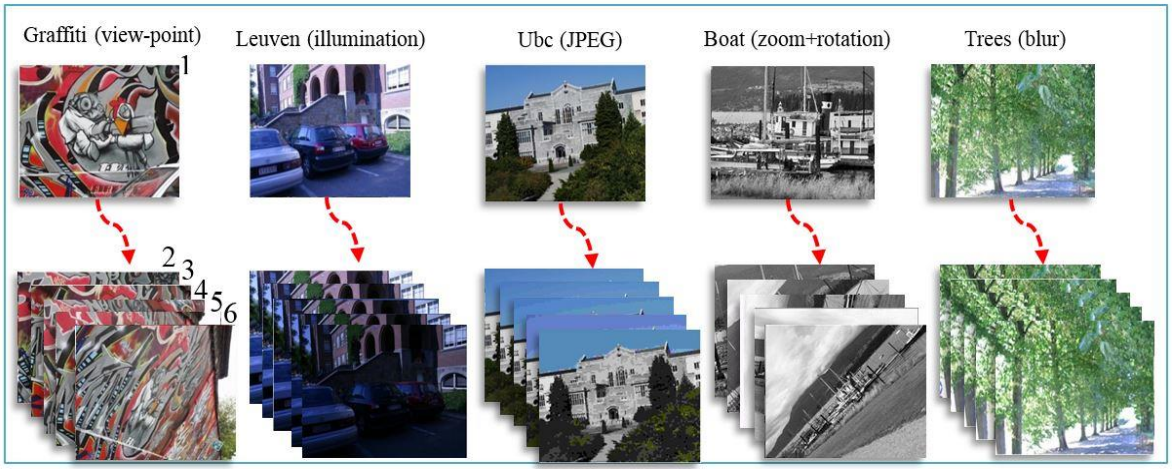


Figure 50. Les différents types de transformations de la base de données de Mikolajczyk (Mikolajczyk & Schmid 2005).

3.2.1.1.2 Test et évaluation du détecteur

Nous avons implémenté notre détecteur via la plateforme décrite dans la section 3.2.1. Afin d'évaluer les performances de notre détecteur, et de le comparer avec les autres détecteurs proposés, nous avons utilisé la métrique de répétabilité (Schmid et al. 2000) qui représente la capacité de détecter le même point de la scène sous différents points de vue et de changements l'éclairage. La répétabilité peut être calculée par l'équation (9) suivante :

$$Répétabilité = \frac{\boldsymbol{nombre\ de\ correspondances\ correctes}}{\boldsymbol{nombre\ total\ de\ points\ détectés}} \quad (9)$$

Nous avons calculé la répétabilité de notre détecteur en variant le point de vue et l'éclairage, et nous l'avons comparé avec d'autres détecteurs récents. La figure 51 montre que, notre détecteur présente des performances nettement meilleures que le détecteur ORB et le détecteur BRISK lors du changement de l'éclairage, et légèrement meilleures pour le changement d'échelle et de rotation.

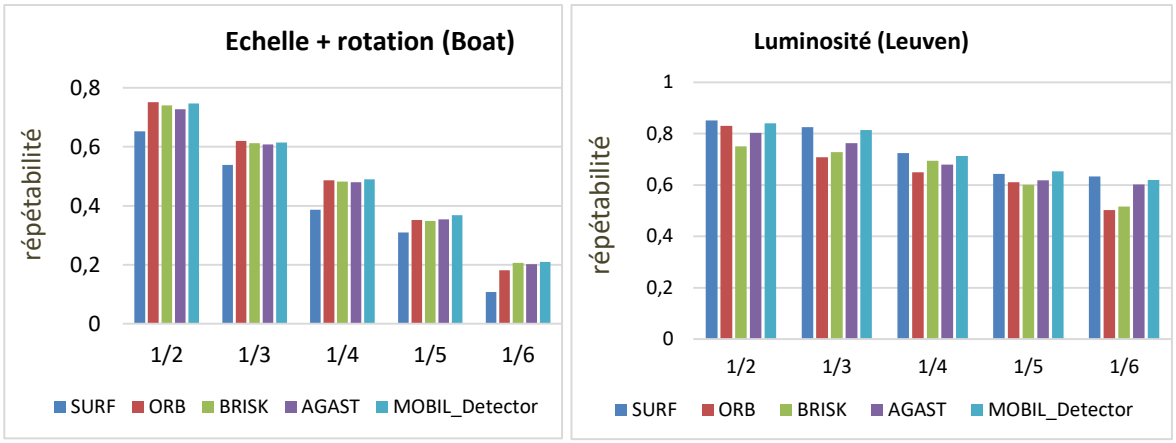


Figure 51. Comparaison de la répétabilité de notre détecteur avec d'autres détecteurs. 1/x signifie la comparaison de l'image 1 avec l'image x dans la famille utilisée.

En outre, nous avons mesuré le temps moyen de la détection pour le détecteur MOBIL_Detector ainsi que pour les détecteurs comparés ci-dessus. Le temps moyen de la détection est le temps nécessaire pour extraire 500 points d'intérêt de différentes images. Tous les détecteurs sont exécutés sur les mêmes images et dans la même configuration que celle décrite dans la section 3.2.1.1.1 ci-dessus. Le tableau 5 montre les temps moyens calculés.

Tableau 5. Le temps moyen pour la détection de 500 points d'intérêt pour les détecteurs MOBIL_Detector SURF, ORB, BRISK et AGAST.

Détecteurs	Temps (ms)
SURF	72.63
ORB	8.37
BRISK	4.12
AGAST	1.87
MOBIL_Detector	3.76

Comme illustré dans le tableau 5, MOBIL_Detector fonctionne légèrement mieux que le détecteur BRISK, et beaucoup mieux qu'ORB et SURF. Cependant, le détecteur AGAST est plus rapide, parce que notre détecteur utilise en plus la mesure de Shi-Tomasi pour le raffinement.

3.2.2 La description des points d'intérêt.

Vu que la plupart des changements photométriques, tels que les changements d'éclairage / contraste, le flou, et les bruits peuvent être éliminés en calculant la différence entre des sous-régions dans un patch (la zone autour du point d'intérêt), les descripteurs binaires sont de plus en plus utilisés dans la vision par ordinateur. Cependant, la plupart des descripteurs binaires utilisent des différences d'intensité, ce qui peut conduire à une insuffisance de description (discrimination limitée).

La figure suivante (figure 52) montre la différence de la précision entre la description basée sur la différence de l'intensité, et la description basée sur la différence des moments géométriques pour deux patchs différents. Nous remarquons ainsi, que malgré que les deux patchs soient visiblement différents, le test binaire de l'intensité rend la même valeur binaire pour les deux patchs. Au contraire, les tests binaires à base de moments montrent une capacité discriminative remarquable.

Avec :

- I^x Intensité moyenne de la région *x*, M_{pq}^x moment d'ordre *pq* pour la sous-région *x*.
- $d(I^{a_1}, I^{a_2})$ et $d(M_{10}^{a_1}, M_{10}^{a_2})$ sont respectivement la différence binaire de l'intensité et les moments entre les deux sous-régions a_1 et a_2.

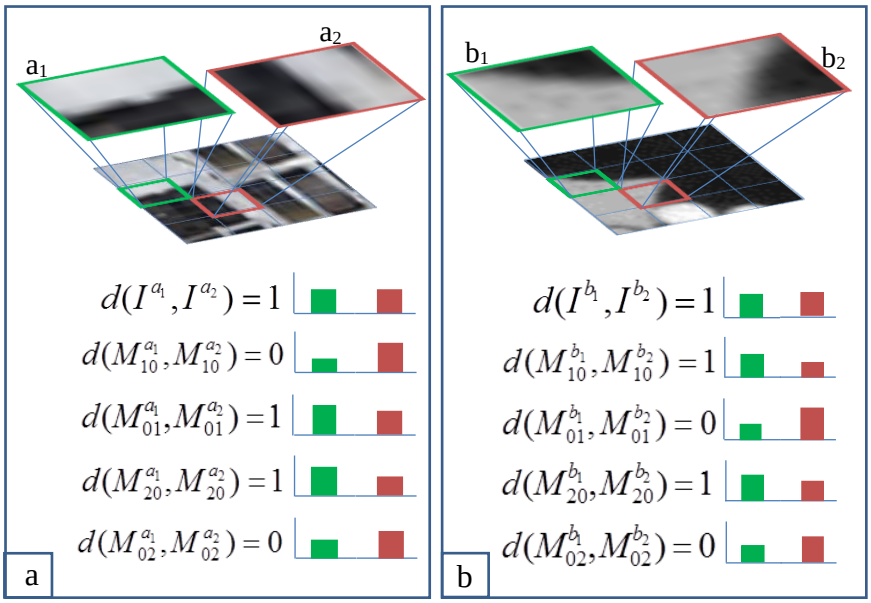


Figure 52. Comparaison du même test binaire entre deux patchs différents (a et b). Nous constatons que les tests basés intensité retournent la même valeur binaire, alors que les patchs soient complètement différents. Cependant les tests binaires basés moments retournent des descriptions exactes de chaque patch.

De ce fait, nous avons proposé un nouveau descripteur binaire appelé « MOBIL : MOments based BInary differences for Local description ». Dans ce descripteur nous effectuons des tests binaires entre des propriétés géométriques et statistiques des sous-régions du patch en utilisant les moments géométriques au lieu des tests binaires classiques basés sur l'intensité, afin d'augmenter la capacité discriminative du descripteur.

3.2.2.1 MOBIL : un nouveau descripteur binaire basé sur les moments.

Les moments fondamentaux ou les moments géométriques ont attiré l'attention des chercheurs depuis plusieurs décennies, et sont considérés comme un outil puissant pour décrire le contenu d'une image. Les moments sont largement utilisés dans les statistiques pour caractériser la distribution des variables aléatoires, ou dans la reconnaissance de formes et la vision par ordinateur (Flusser et al. 2009), (Papakostas et al. 2013), (Doretto & Yao 2010) et (Karakasis et al. 2015), afin de décrire le contenu de l'image. Hu (Hu 1962) a été le premier à introduire des moments en traitement d'images en utilisant les moments géométriques, centrales et normalisées.

Comme défini par Hu (Hu 1962), les moments bidimensionnels pour une image I(x, y) de dimensions N x M, sont définis par l'équation (10).

$$m_{pq} = \sum_{y=0}^{M-1} \sum_{x=0}^{N-1} x^p y^q I(x,y) \tag{10}$$

Avec p, q = 0, 1, 2. Tel que :

- m_{00} (moments d'ordre zéro) : représente la masse totale (ou la somme des valeurs des pixels) pour une région donnée de l'image.
- m_{01}, m_{10} (moments d'ordre un) : sont utilisés pour localiser le barycentre (centre de gravité) d'une région.
- m_{20}, M_{02} (Moments de deuxième ordre) : déterminent les axes principaux de la distribution des pixels dans une image ou une région donnée.

Dans la première version de notre descripteur MOBIL, publiée dans la conférence IEEE ISMAR 2014 (Bellarbi et al. 2014a), nous avons divisé le patch en quatre par quatre sous-régions (4x4). Nous avons calculé pour chaque sous-région les cinq moments : m_{00}, m_{01}, m_{10}, m_{02}, m_{20}. Nous avons ensuite effectué un test binaire τ sur chaque paire de sous-régions adjacentes (*x, y*) en utilisant les moments définis comme suit (équation 11) :

$$\tau(m_{pq}(x), m_{pq}(y)) = \begin{cases} 1 & if\,(m_{pq}(x) > m_{pq}(y)) \\ 0 & otherwise \end{cases} \tag{11}$$

Avec $(p,q) \in \{(0,0),(0,1),(1,0),(0,2),(2,0)\}$.

Nous pouvons généraliser le descripteur MOBIL pour un patch ρ par la formule suivante (équation 12) :

$$MOBIL(\rho) = \sum_{i=0}^{n} \sum_{j=0}^{5} 2^{j+5i} (\tau(m_{pq}(x), m_{pq}(y))$$
$$\text{pour chaque } (p, q) \in \{(0,0),(0,1),(1,0),(0,2),(2,0)\}. \quad (12)$$

Avec n = 42 le nombre de paires possibles des sous-régions adjacentes dans un patch ρ divisé sur 4x4 sous-régions égales. La figure 53 illustre l'architecture du descripteur MOBIL.

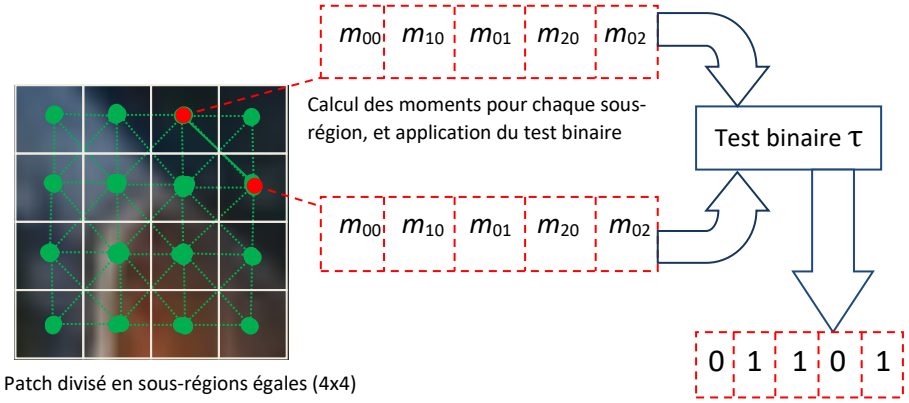


Figure 53. Principe de fonctionnement du descripteur MOBIL.

3.2.2.1.1 Invariance à la rotation.

Puisque nous visons à développer des applications de réalité augmentée en temps réel, pour lesquelles les changements de rotation sont très fréquents, la construction d'un descripteur invariant à la rotation est cruciale. Pour ce faire, nous devons estimer une orientation dominante du patch et aligner ce dernier par rapport à cette orientation avant le calcul de son vecteur descriptif.

Plusieurs approches d'estimation de l'orientation ont été proposées dans la littérature. Nous avons appliqué la méthode du centre de gravité introduite dans (Rosin 1999) qui est connue pour son efficacité et sa robustesse.

Ainsi, nous calculons le barycentre C du patch en utilisant les moments : M_{00}, M_{01} et M_{10} comme suit (Equation 13) :

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (13)$$

Nous avons ensuite construit un vecteur $\overrightarrow{OC}$ du centre O du patch vers le barycentre C. L'orientation θ du patch est alors (Equation 14):

$$\theta = \tan^{-1}(\overrightarrow{OC}_x, \overrightarrow{OC}_y) \quad (14)$$

3.2.2.1.2 Test et évaluation du descripteur MOBIL.

Nous avons implémenté le descripteur MOBIL en C# sous l'environnement Visual Studio de Microsoft. Nous avons effectué des tests sur la base de données d'images Mikolajczyk (voir la section 3.2.1.1.1)

Nous avons tout d'abord calculé le temps moyen pour construire une description d'un patch, puis nous l'avons comparé avec d'autres descripteurs. Comme nous pouvons le constater (voir Tableau 6), MOBIL donne un meilleur résultat en termes de temps de calcul comparé à ORB, et dépasse largement SURF en rapidité de calcul.

Tableau 6. Temps de description pour les descripteurs testés et le descripteur MOBIL

Descripteurs	Temps de description (ms)
SURF	1.488
BRISK	0.062
ORB	0.146
FREAK	0.139
LATCH	0.437
MOBIL	0.127

En outre, nous avons comparé la robustesse du descripteur MOBIL avec d'autres descripteurs, toujours en utilisant la base de données de Mikolajczyk. Les résultats obtenus sont présentés dans la figure 54.

A partir des graphes de la figure 54, nous avons conclu que l'utilisation des moments géométriques pour les tests binaires a augmenté le taux de reconnaissance des objets et ceci en augmentant la distinction entre les descriptions de différents patchs.

Il est à noter que le taux de reconnaissance représente le nombre de correspondances correctes divisé par le nombre total de correspondances.

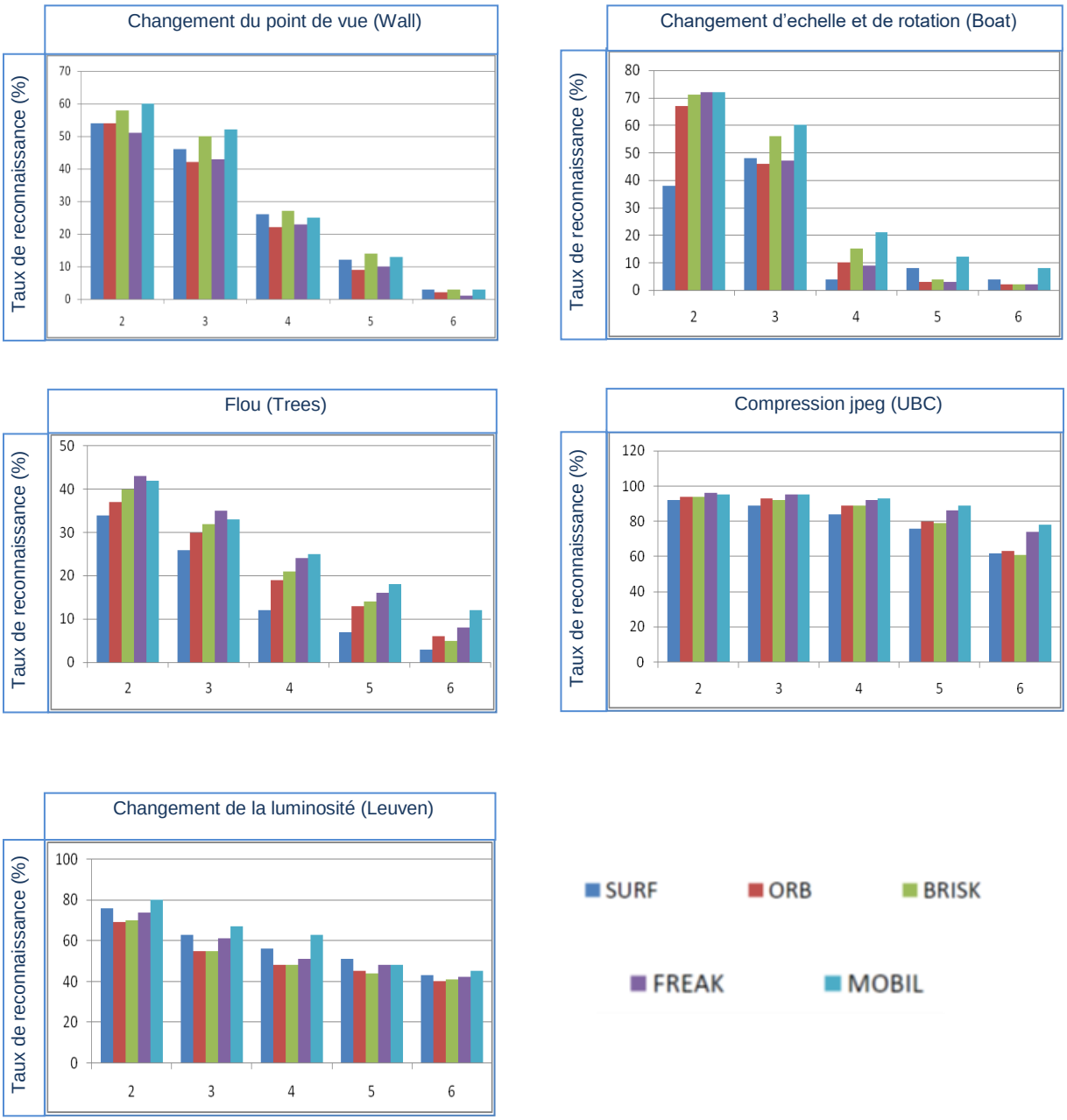
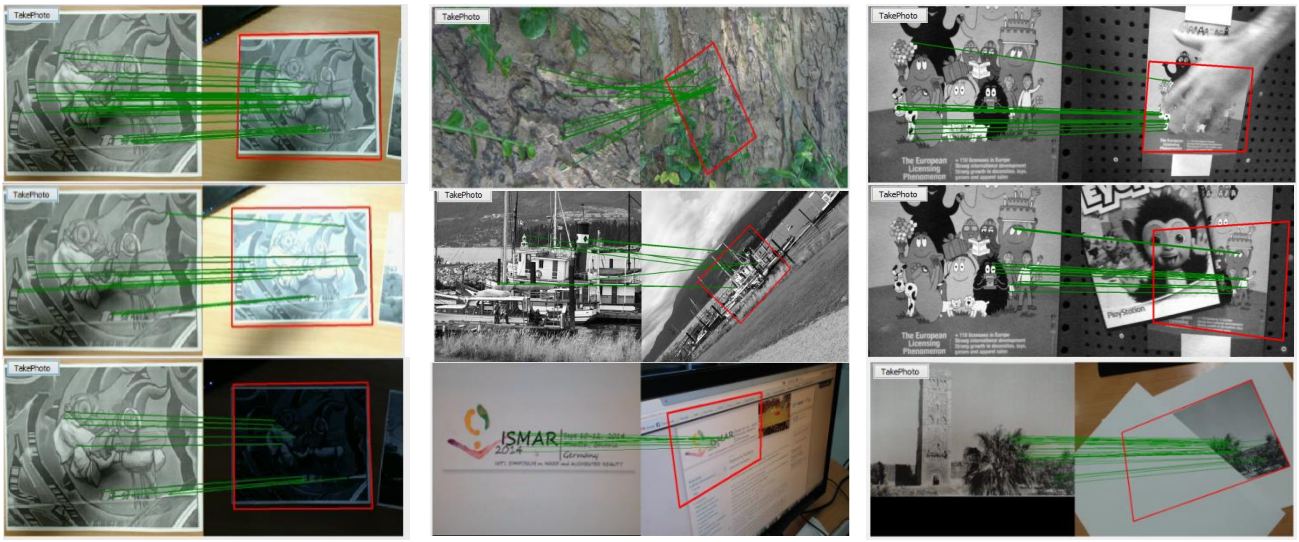


Figure 54 : Comparaison de MOBIL, avec Freak, ORB, BRISK, et SURF en utilisant la base de données (Mikolajczyk & Schmid 2005), 1/x signifie la comparaison de l'image 1 avec l'image x dans la famille utilisée.

La figure 55 montre quelques résultats du descripteur MOBIL avec différents transformations et changements (transformations affines (rotation, translation, et changement d'échelle), changement de luminosité, occultations, et changement de point de vue). Les résultats obtenus ont montré l'invariance du descripteur proposé devant ces différents types de transformations.



Changement de luminosité. Transformations affines Occultations

Figure 55. Exemples de tests de MOBIL avec différents types de transformations.

En outre, des tests, des évaluations et des comparaisons approfondies du descripteur MOBIL avec d'autres descripteurs connus ont été présentés dans (Madeo & Bober 2016). Nous présentons dans la figure ci-dessous (Figure 56) la courbe ROC (Receiver Operating Characteristics) extraite de (Madeo & Bober 2016) et calculée pour le descripteur MOBIL, ainsi que d'autres descripteurs de l'état de l'art, en utilisant l'ensemble des images « Magazines » de la base de données « Surrey Mobile Dataset[1] ».

Notons que la courbe ROC (Fawcett 2004) offre à la fois une vision graphique et une mesure pertinentes de la performance d'un descripteur en termes de taux de reconnaissance. Elle est définie comme le rapport entre le taux du vrai-positif (TPR : True Positive Rate) et le taux du faux-positif (FPR : False Positive Rate) d'un descripteur. Ainsi, un descripteur robuste devrait fournir un taux du vrai-positif élevé pour tout taux du faux-positif donné.

Selon la courbe ROC de la figure 56. Nous constatons que le descripteur MOBIL présente un taux du vrai-positif (TPR) élevé par rapport aux autres descripteurs tout en variant le taux du faux positif.

[1] Surrey Mobile Dataset, http://cvssp.org/data/surreymobile/

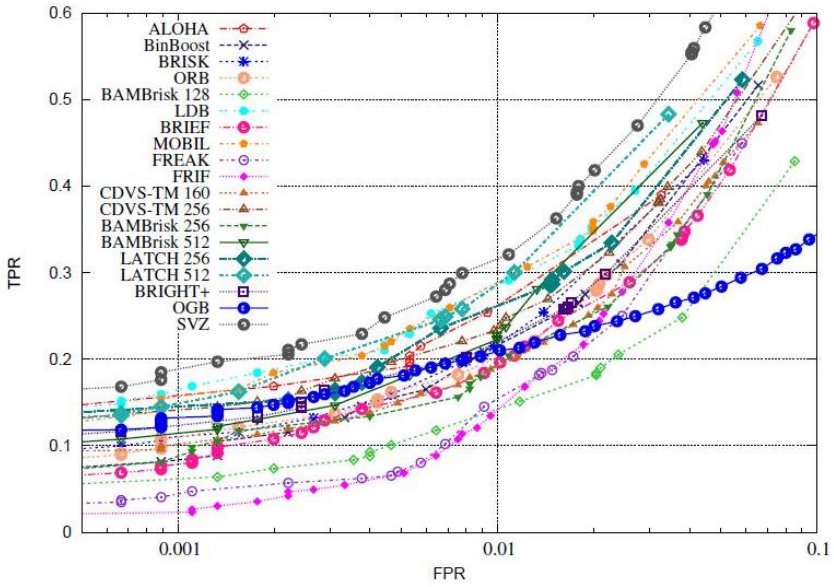


Figure 56. Courbe ROC calculée pour le descripteur MOBIL et d'autres descripteurs de l'état de l'art. Extraite de (Madeo & Bober 2016).

Cette première version du descripteur MOBIL a introduit la comparaison entre des moments géométriques calculés dans les sous-régions du patch, afin de générer une description binaire. Cependant, après une série de tests sur différents types d'images, nous avons constaté que le nombre réduit de tests binaires effectués cause une perte d'informations lors de la description. Dans cette optique, nous avons proposé des améliorations pour notre descripteur MOBIL, afin d'augmenter la robustesse et la précision de la description.

3.2.2.2 MOBIL_2B : MOBIL avec deux bits

Dans cette version améliorée de MOBIL, appelée MOBIL_2B (Bellarbi et al. 2015), tout comme l'indique son nom, nous avons affecté deux (2) bits pour chaque test binaire au lieu d'un seul bit, ceci afin d'améliorer la précision de la description.

Étant donné deux patchs extraits de deux images distinctes, nous voulons calculer les moments géométriques d'ordre 1 (M_{10}) pour les sous-régions des patchs afin de faire la comparaison binaire. Nous pouvons constater dans la figure ci-dessous (figure 57) que malgré la différence visible entre les deux patchs, les résultats issus des tests binaires effectués entre les moments des sous-régions des deux patchs restent sont identiques.

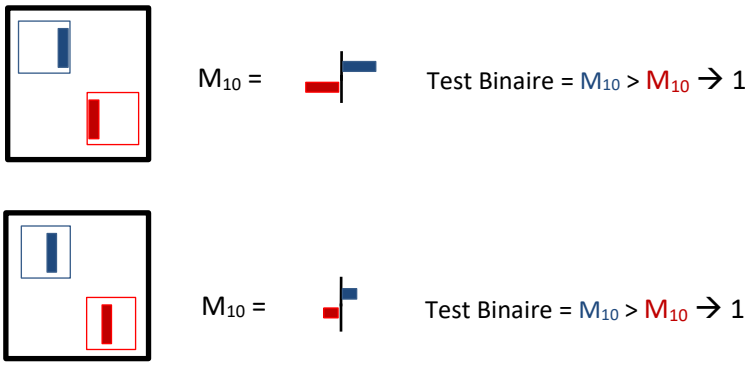


Figure 57. Représentation de deux patches différents qui génèrent la même description binaire.

Afin de remédier à ce problème, nous avons introduit un seuil au niveau du test de comparaison entre les moments ce qui génère quatre possibilités au lieu de deux, et augmente la distinction entre les patchs. La figure 58 montre le schéma global du descripteur MOBIL_2B.

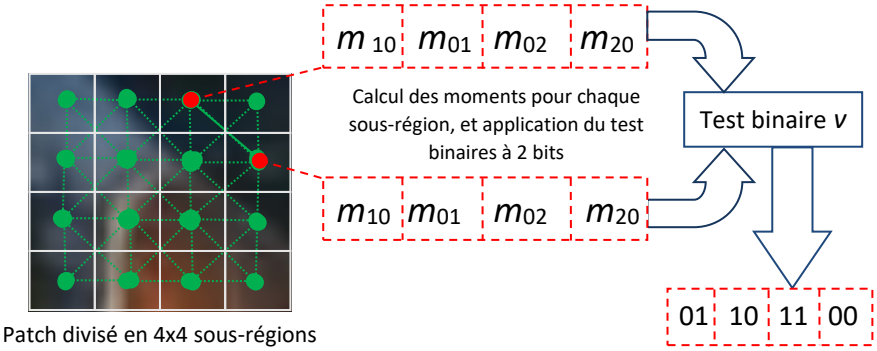


Figure 58. Principe de fonctionnement du descripteur MOBIL_2B

D'un point de vue mathématique, (m_{pq}, m'_{pq}) représente les moments géométriques calculés pour deux sous-régions d'un patch donné, tel que (m_{pq}, m'_{pq}) ∈ {(m_{10}, m'_{10}), (m_{01}, m'_{01}), (m_{20}, m'_{20}), (m_{02}, m'_{02})}.

On définit le test binaire *v* appliqué sur la paire (m_{pq}, m'_{pq}) comme suit (Equation 15) :

$$v = \begin{cases} 00 & if \quad m_{pq} > m'_{pq} + t \\ 01 & if \quad m'_{pq} < m_{pq} < m'_{pq} + t \\ 10 & if \quad m'_{pq} > m_{pq} > m'_{pq} - t \\ 11 & if \quad m_{pq} < m'_{pq} - t \end{cases} \tag{15}$$

Avec t un seuil prédéfini.

Vu la différence de la nature des moments calculés, le choix du seuil *t* a été fait de la manière suivante :

- Pour les moments M_{10} et M_{01} qui signifient les centres de masse respectivement par rapport à l'axe x et y, nous avons choisi le seuil *t* comme étant la moitié de la taille de la sous-région D (Equation 16).

$$t = M_{00} \times D/2 \tag{16}$$

- Pour les moments M_{20} et M_{02} qui signifient la distribution des données respectivement sur les axes x et y, nous avons choisi le seuil comme étant un quart du carré de la taille D de la sous-région. Donc la valeur *t* est calculée comme suit (Equation 17).

$$t = M_{00} \times D^2/4 \tag{17}$$

Quant aux propriétés relatives aux points d'intérêts, nous avons gardé les mêmes que celles utilisées dans la première version de MOBIL. Ainsi, pour l'invariance à la rotation, nous avons appliqué la méthode du centre de gravité tout comme dans la première version.

On note que pour ce qui suit, nous allons appeler la première version de **MOBIL par MOBIL_1B**.

Nous avons implémenté MOBIL_2B en C# sous l'environnement Visual Studio de Microsoft. Nous avons effectué des tests sur la base de données des images Mikolajczyk (voir section 3.2.1.1.1.). En outre, nous avons comparé la robustesse du descripteur MOBIL_2B avec MOBIL_1B et avec d'autres descripteurs. Les résultats obtenus sont décrits par les graphes de la figure 59.

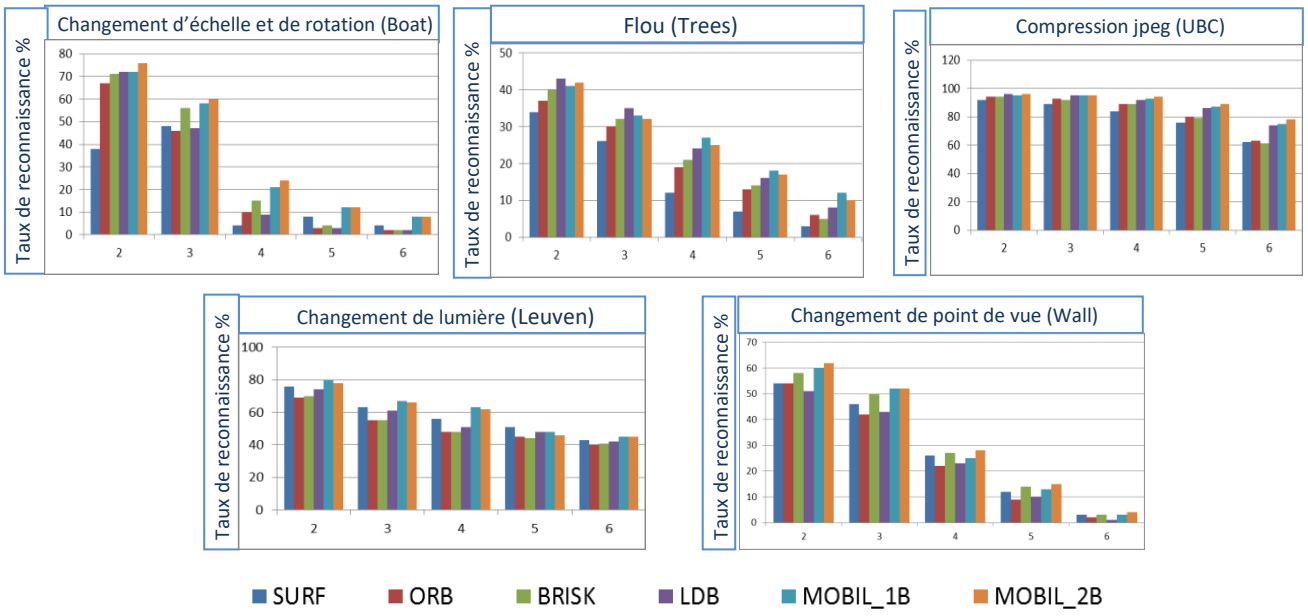


Figure 59. Comparaison de MOBIL_2B avec MOBIL_1B, Freak, ORB, BRISK, et SURF. 1/x signifie la comparaison de l'image 1 avec l'image x dans la famille utilisée.

Les résultats obtenus ont montré que l'introduction du test à 2 bits a amélioré la précision de la description par rapport à la version de MOBIL_1B. La figure ci-dessous (figure 60) montre l'efficacité de MOBIL_2B sur les images « Wall » et « Boat ».

Figure 60. Résultats de test de MOBIL_2B avec les images Boat 1/6 et Wall 1/5.

Bien que MOBIL_2B soit plus efficace que MOBIL_1B en terme de distinction, cependant, cette amélioration consomme plus de temps de calcul par rapport à la première version (136 ms pour MOBIL_2B vs 127 ms pour MOBIL_1B). (Voir Tableau 7).

Tableau 7. Temps moyen par description pour le descripteur MOBIL_2B et les autres les descripteurs.

Descripteurs	Temps moyen par description (ms)
SURF	1.488
BRISK	0.062
ORB	0.146
LDB	0.139
LATCH	0.437
MOBIL_1B	0.127
MOBIL_2B	0.136

A partir des tests effectués, nous avons constaté que MOBIL_2B améliore légèrement la description des patchs comparé à MOBIL_1B. Cependant, cette approche est très sensible aux bruits et consomme plus de temps de calcul par rapport à la version précédente.

En outre, nous avons constaté que la limitation de cette technique de description est due d'une part à la forme du patch et les sous-régions, et d'autre part au choix des tests binaires. Dans cette optique, nous allons présenter dans ce qui suit, une nouvelle version de notre descripteur en introduisant des améliorations dans les 2 parties : description et apprentissage.

3.2.2.3 POLAR_MOBIL : MOBIL avec des images polaires

Lors des tests de nos descripteurs MOBIL_1B et MOBIL_2B, nous avons souvent remarqué dans l'étape d'appariement que de nombreux patchs qui sont visuellement différents sont étiquetés comme identiques et vice-versa, surtout en présence d'une transformation affine. Nous avons constaté que cela est dû aux deux principaux problèmes qui sont :

- La configuration du patch (Sampling Pattern) : dans les deux versions précédentes de MOBIL, nous avons divisé le patch en 4x4 sous-régions égales. Ainsi, lors d'un changement du point de vue, le contenu des sous-régions change, ce qui génère des descriptions différentes bien que le patch soit identique. Une des solutions à ce problème est d'opter pour une configuration polaire.
- La génération d'un vecteur descriptif de grande dimension conduit à une meilleure description, mais augmente la complexité du calcul et génère des descriptions redondantes. Afin de remédier à ce problème, nous avons proposé une stratégie de sélection des tests binaires pour éliminer les tests redondants ou corrélés.

Dans cette optique, nous avons proposé une version améliorée du descripteur MOBIL, dans laquelle nous calculons la description binaire de l'image du patch et l'image polaire du patch, et nous concaténons les deux vecteurs descriptifs résultants.

L'image polaire est une image dans laquelle les coordonnées des pixels sont calculées en fonction de l'angle θ et du rayon ρ. Ainsi, la transformation d'une image polaire vers une image cartésienne (Polar Image Mapping) est la transformation d'une image de l'espace polaire en fonction de l'angle θ et le rayon ρ vers l'espace cartésien en fonction de x, y. (figure 61)

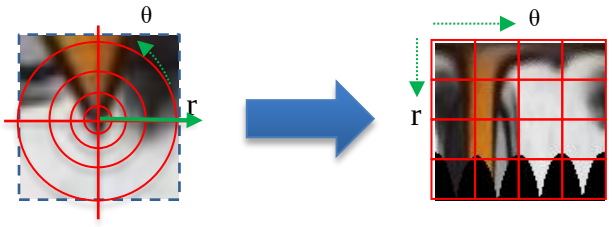


Figure 61. La projection du patch de l'espace cartésien sur l'espace log-polaire

L'image polaire **logarithmique** (Log-Polar) est une image polaire où la variation du rayon suit une échelle logarithmique. Les équations 18 et 19 montrent la conversion d'une image avec des coordonnées log-polaires vers une image cartésienne et vice-versa successivement.

$$\begin{cases} x = e^{\rho} \cos\theta \\ y = e^{\rho} \sin\theta \end{cases} \tag{18}$$

$$\begin{cases} \rho = \log\sqrt{x^2 + y^2} \\ \theta = \arctan y/x \end{cases} \tag{19}$$

Les images polaires sont utilisées généralement pour amplifier ou donner plus de poids pour une région dans une image par rapport à l'autre avec une dégradation uniforme. Dans notre

cas, nous avons utilisé l'image polaire du patch afin d'augmenter le poids du centre du patch par rapport à la partie extérieure. Ce qui diminue l'effet du changement du point de vue. Dans cette optique, notre nouveau descripteur binaire, est construit comme suit :

Nous avons pris un patch carré ρ autour du point d'intérêt détecté, avec une longueur L = 32 pixels. Ensuite, nous l'avons divisé en 4 × 4 cellules (sous-régions) de tailles égales. Nous avons obtenu 16 sous-régions de 8 × 8 pixels chacune.

D'autre part, nous avons pris un patch circulaire autour du même point d'intérêt, avec un rayon r = ½ L × √2 ≈ 23 pixels et nous avons construit son log-polaire image. Nous avons ensuite rogné la région extérieure de l'image Log-polaire pour obtenir un patch carré de longueur L'= 32 pixels. Ce dernier a été également divisé en 4 × 4 cellules de tailles égales, avec 8 × 8 pixels chacune. La figure 62 illustre cette procédure.

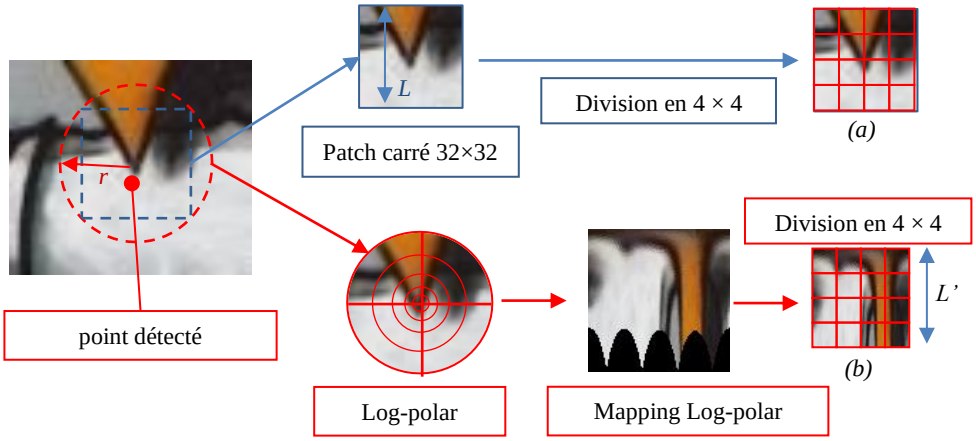


Figure 62. Extraction et échantillonnage du patch cartésien (a) et du patch log-polaire (b).

avec L = L' = 32, et r = ½ L × √2 = 23.

Nous notons que plusieurs essais expérimentaux ont été effectués afin de choisir la taille adéquate des patchs, ainsi que le nombre et les tailles des sous-régions.

Les étapes de cette nouvelle version sont résumées comme suit :

1. Calculer l'image log-polaire du patch.
2. Utiliser les 4 moments : M_{01}, M_{10}, M_{02} et M_{20}, pour la description binaire.
3. Appliquer les tests binaires sur les deux : image cartésienne et log-polaire du patch.
4. Effectuer les tests binaires sur toutes les combinaisons possibles des paires de sous-régions du patch.
5. Concaténer les deux vecteurs binaires descriptifs.

En utilisant le test binaire τ (Equation 11, Section 3.2.2.1) décrit dans la première version du MOBIL, nous pouvons définir notre nouveau descripteur POLAR_MOBIL pour décrire un patch ρ par l'équation suivante (Equation. 20) :

$$POLAR\_MOBIL(\rho) = \sum_{i=0}^{n/2-1}\sum_{j=0}^{3} 2^{j+4i}(\tau(m_{pq}(x), m_{pq}(y)) + \sum_{i=n/2}^{n}\sum_{j=0}^{3} 2^{j+4i}(\tau(m'_{pq}(x), m'_{pq}(y))$$

$$\text{pour chaque } (p,q) \in \{(0,1),(1,0),(0,2),(2,0)\}. \qquad (20)$$

n représente le nombre de tests binaires dans les deux images cartésienne et log-polaire du patch ρ. Le vecteur binaire résultant contient alors n × 4 bits. « *m* » et « *m*' » sont les moments calculés pour les patchs cartésiens et log-polaires respectivement.

Ainsi, la figure 63 montre l'architecture de la nouvelle version appelée « POLAR_MOBIL ».

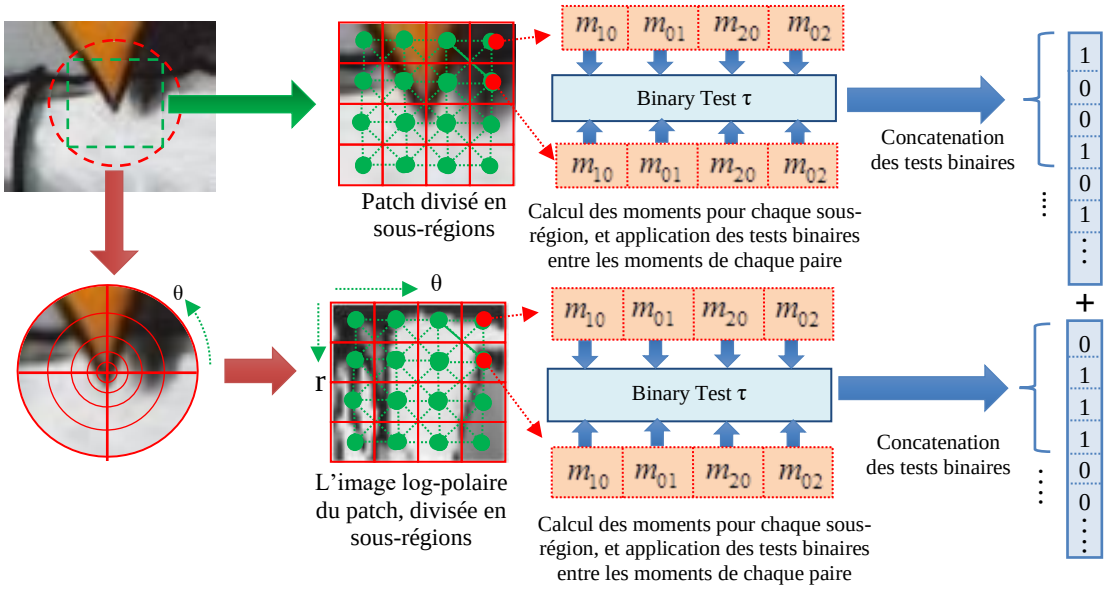


Figure 63. Architecture du descripteur POLAR_MOBIL

Pour l'invariance à la rotation, nous avons appliqué la même technique basée sur le centre de gravité, utilisée dans la version MOBIL_1B (Section 3.2.2.1).

3.2.2.3.1 Sélection des meilleurs tests de POLAR_MOBIL

Ce descripteur génère une description de dimension de 960 bits :

- Tous les tests possibles = $C_2^{16} = \frac{16!}{2!\times(16-2)!} = 120$
- Moments utilisés = 4.
- Patch Cartésien + patch log-polaire = 2.

⇨ $120\times4\times2 = 960$ bits

Cette dimension, qui est relativement grande, augmente la consommation de temps de traitement et de la mémoire. En outre, les vecteurs descriptifs de grande taille peuvent contenir des bits hautement corrélés qui réduisent la capacité discriminative du descripteur.

Pour remédier à cela, nous avons appliqué une stratégie de sélection de tests binaires afin d'une part de réduire la forte dimensionnalité de ce descripteur, et d'autre part de permettre de sélectionner des tests binaires qui conduisent à une forte variance et une faible corrélation.

Pour ce faire, nous avons appliqué un apprentissage de notre descripteur sur un grand ensemble de patchs. Nous avons utilisé la base de données Caltech-256 (Griffin et al. 2007) composée de 30.607 images (Figure 64). Nous avons ensuite extrait, 100 points d'intérêt (en moyenne) de chaque image. Nous avons obtenu environ 3 millions patchs sur toute la base.

Figure 64. Base de données Caltech-256[1].

Pour chaque point extrait, nous avons calculé son descripteur binaire par POLAR_MOBIL. Ensuite, dans chaque vecteur binaire, nous avons regroupé tous les quatre (4) bits successifs (les quatre bits générés par les quatre moments à chaque test binaire) et nous les avons convertis en équivalent décimal (c'est-à-dire, $(1001)_2 = (9)_{10}$). Par conséquent, nous avons obtenu des vecteurs décimaux de dimension 960/4 = 240 avec des valeurs comprises entre 0 et 15, comme illustré à la Figure 65.

[1] Caltech-256, 2007. http://www.vision.caltech.edu/Image_Datasets/Caltech256/

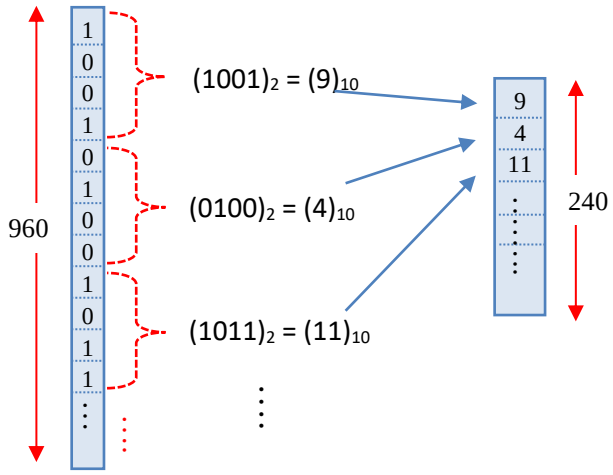


Figure 65. Conversion du vecteur binaire en vecteur de valeurs décimales.

Ces vecteurs de valeurs décimales ont formé une matrice de k × n. Avec, n = 2 * 120 = 240 est le nombre total possible de tests binaires pour les deux représentations du patch (cartésien et log-polaires), et *k* est le nombre total de patchs extraits de la base de données (~ 3 Millions). Nous avons ensuite calculé la moyenne et la variance pour chaque rangée. La figure 66 montre les valeurs moyennes tracées.

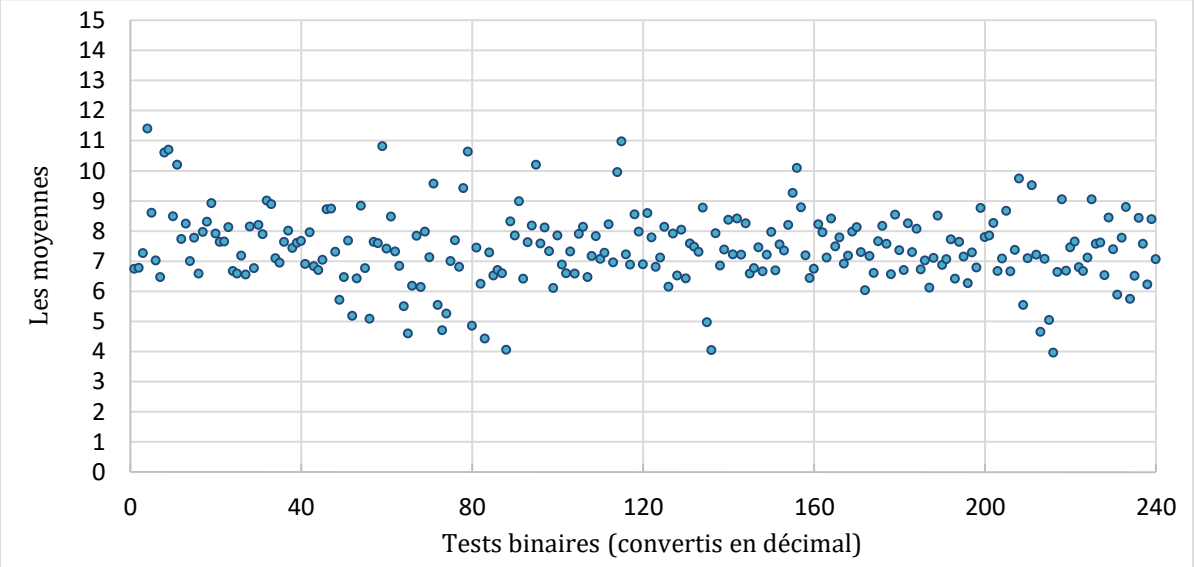


Figure 66. Graphe des valeurs moyennes de 240 lignes de la matrice. Nous pouvons constater que les tests binaires corrélés sont ceux qui sont loin du centre (7.5).

Sélectionner des tests non corrélés revient à sélectionner des lignes qui suivent une distribution uniforme (c.à.d. moyenne proche de 7,5 (voir la figure 66.) et un écart type au voisinage de 21,25), selon l'équation de la variance de la distribution discrète uniforme (Equation 21) (Johnson et al. 2005)).

$$V = (k^2 - 1)/12 \qquad (21)$$

L'algorithme se résume comme suit : (Algorithme 1.).

Algorithme 1. Sélection de Tests Binaires

Entrée:

- M : Matrice $k \times n$ des patchs appris, avec les valeurs dans [0, 15].
- $\varepsilon_1, \varepsilon_2$ *:* deux seuils pour la sélection respectivement de la moyenne et l'écart type.
- $k' = 32 \times 2 = 64$: nombre de lignes à sélectionner.

Sortie:

- I: ensemble des tests binaires sélectionnés.

Début:

1. Pour chaque ligne i de la matrice M, calculer la moyenne m_i et la variance v_i.
2. Trier par ordre croissant les lignes i en fonction de $|m_i - 7.5|$.
3. $i = 1$*;* Initialiser I*;*
4. *tant que* $|m_i - 7.5| < \varepsilon_1$*, faire* :
   a. *Si* $(|v_i - 21.25| < \varepsilon_2)$ *alors* mettre i dans I.
   b. *Si* la taille de I égal à k*', alors aller à étape 5*
   c. *Si* $(i = k)$*, alors* réajuster ε_1 et ε_2*, aller à étape 3*
   d. i++;
5. *Fin tant que*.
6. Terminer la procédure d'apprentissage et sortir I.

Fin.

Une fois l'apprentissage terminé, nous obtenons un ensemble de tests égal à 64 (pour les deux patchs cartésien et polaire), ce qui génère un vecteur binaire de taille 265 bits.

Nous avons remarqué que la plupart des tests horizontaux espacés (avec longue distance) appliqués sur le patch cartésien sont éliminés. Ceci s'explique par le fait que le patch est tourné vers son orientation dominante. Ainsi, il apparait homogène horizontalement. Pour la même raison, les tests verticaux de longue distance dans le patch log-polaire sont également supprimés.

3.2.2.3.2 Test et évaluation du descripteur POLAR_MOBIL

Nous avons implémenté, testé et évalué notre descripteur proposé sous le même environnement détaillé dans la section 3.2.1.1.1.

Nous avons tout d'abord mesuré le temps moyen d'une description d'un seul patch, puis nous l'avons comparé avec d'autres descripteurs. Le tableau 8 montre que le temps de description de POLAR_MOBIL est inférieur à celui d'ORB et LDB et meilleur que celui de SURF.

Le temps de la description moyenne est calculé sur plus de 5000 images de différents types. Tous les descripteurs comparés sont exécutés sur les mêmes images et avec la même configuration et conditions que celles décrites dans la section 3.2.1.1.1.

Tableau 8. Temps de description moyen pour les descripteurs testés et le descripteur POLAR_MOBIL

Descripteurs	Temps moyen par description (ms)
SIFT	3.121
SURF	1.488
BRISK	0.072
A-KAZE	0.094
ORB	0.146
LDB	0.139
LATCH	0.437
MOBIL_1B	0.127
MOBIL_2B	0.136
POLAR_MOBIL	0.107

Nous avons par la suite comparé POLAR_MOBIL avec d'autres descripteurs de l'état de l'art en utilisant la base de données Mikolajczyk. Les premiers résultats obtenus (Figure. 67) montrent que POLAR_MOBIL présente un taux de reconnaissance plus élevé que les autres descripteurs à la fois pour le changement d'échelle et la rotation, ainsi que le changement point de vue.

Notons que le taux de reconnaissance est le nombre de correspondances correctes divisé par le nombre total de correspondances.

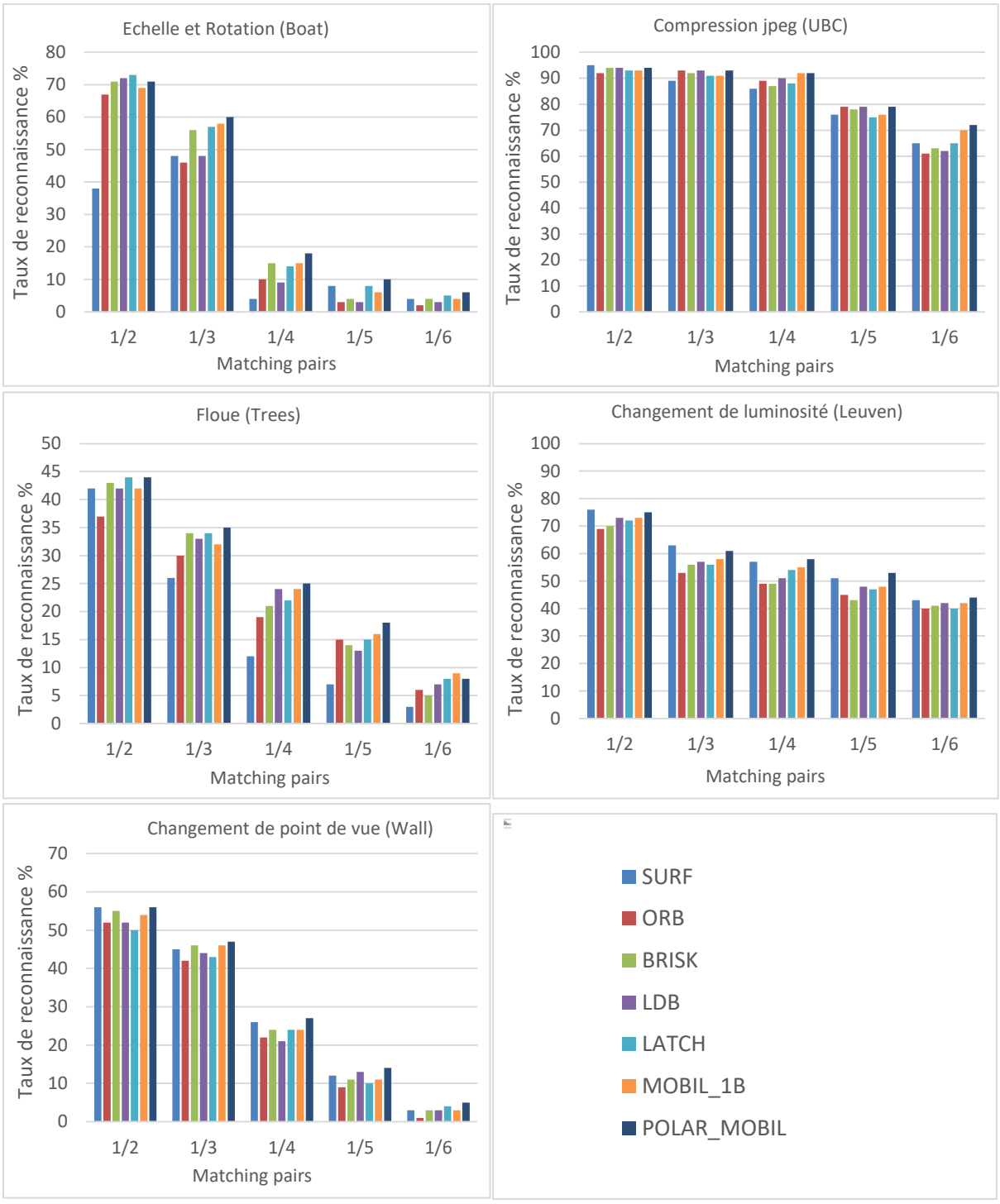


Figure 67. Comparaison du descripteur POLAR_MOBIL, avec MOBIL, LDB, ORB, BRISK, SURF et LATCH. 1/x signifie la comparaison de l'image 1 avec l'image x dans la famille utilisée.

Afin de mieux étudier les performances de notre descripteur, nous avons calculé les courbes de Rappel vs 1-précision (Recall vs 1-Precision) pour les paires 1/3 des images des six séquences d'images de la base Mikolajczyk.

Notons que le Rappel (Recall) est défini comme le nombre de correspondances correctes trouvées divisé par le nombre total de correspondances correctes. La Précision est le nombre de

correspondances correctes trouvées sur le nombre total de correspondances trouvées (Kent et al. 1955).

Une description distinctive devrait fournir une précision élevée pour tout rappel donné. Les résultats présentés sur la Figure 68 montrent que POLAR_MOBIL offre de meilleures performances que SURF et ORB, en particulier pour les changements de luminosité et les transformations affines (changement d'échelle, rotation et changement de point de vue).

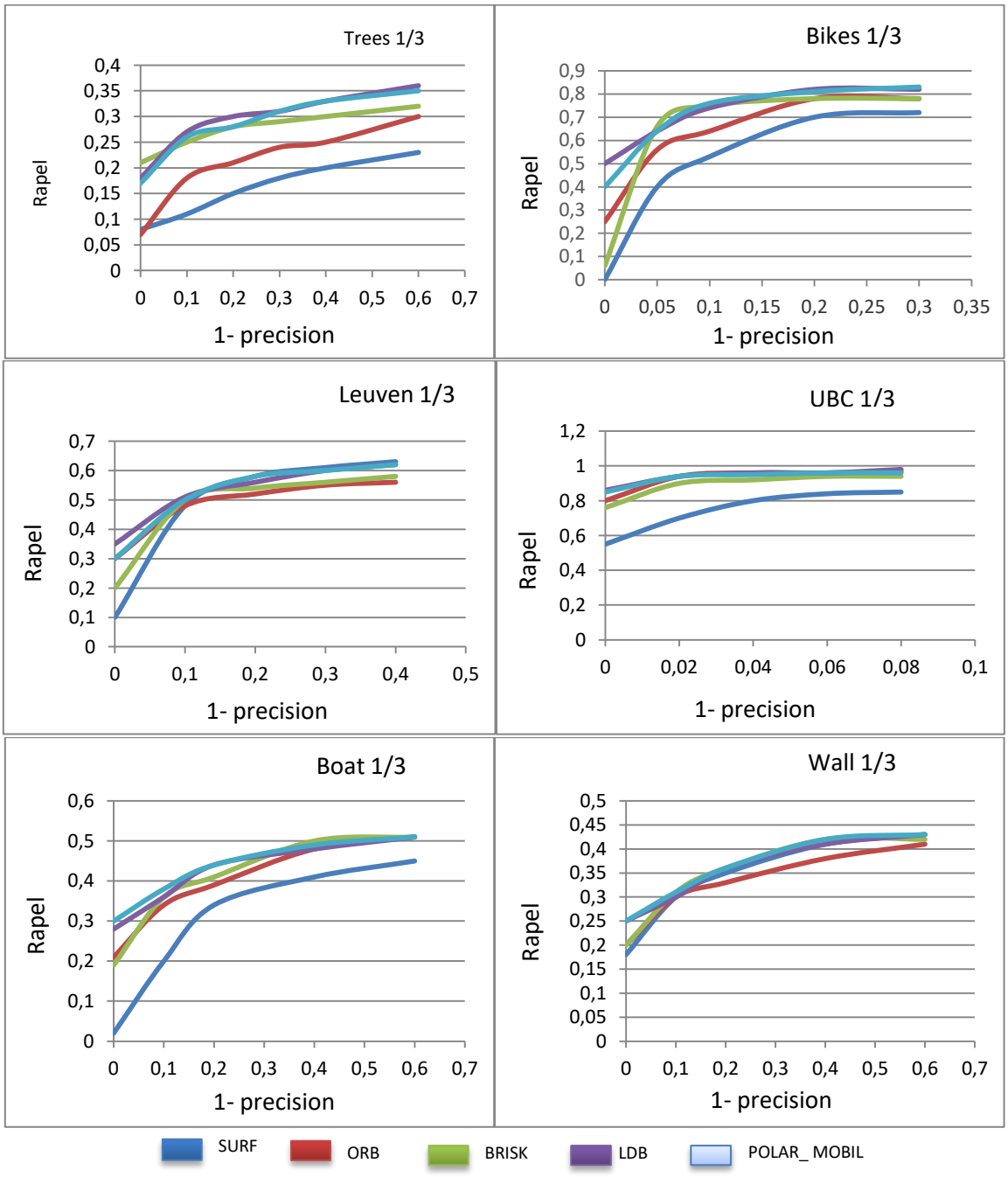


Figure 68. Rappel vs. 1-precision pour POLAR_MOBIL, LDB, ORB, BRISK, et SURF.

Nous avons également analysé la répartition de la distance de Hamming entre les descriptions de POLAR_MOBIL pour des différentes paires d'images. Nous avons pris les paires 1/4 pour chacune des familles d'images : Wall, Leuven, Trees et Boat de la base de données Mikolajczyk. Nous avons calculé les histogrammes normalisés de distances de Hamming entre les correspondances correctes (vert) et les fausses correspondances (rouge). La figure 69 montre la répartition obtenue des distances de Hamming.

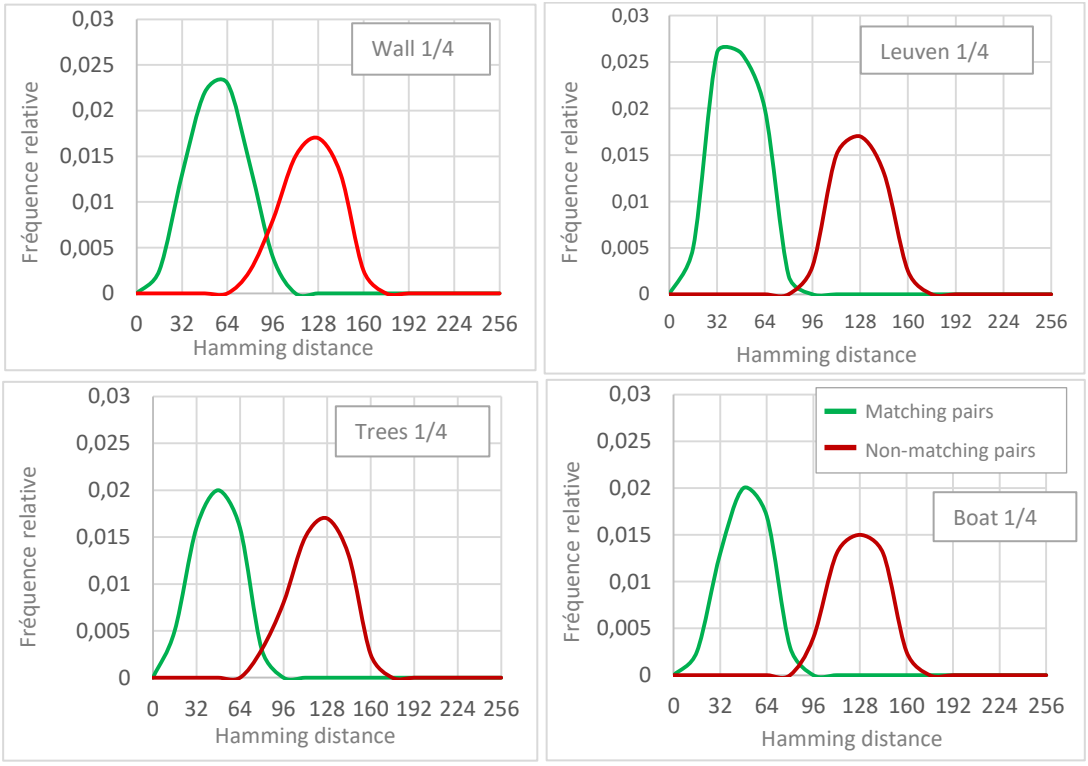


Figure 69. Graphes illustrant les distributions de distances de Hamming. Les corrects matches (vert) et les non matches (rouge). On voit que pour la plupart des paires, les courbes sont clairement séparées, à l'exception des paires de Wall, où les courbes se chevauchent partiellement.

3.3 Vers l'immersion mobile en réalité augmentée

Jusqu'à présent, nous avons abordé la problématique liée à l'extraction et la description des informations à partir d'images capturées par la caméra. Celles-ci seront utilisées lors de la phase de la détermination du point de vue de l'utilisateur dans son environnement.

A cet effet, afin d'assurer une estimation précise de la pose de l'utilisateur pendant tout le processus de navigation/sélection/manipulation, nous avons opté pour deux approches qui seront utilisées respectivement lors de la navigation de l'utilisateur, et pendant sélection/manipulation des objets virtuels. Ces approches seront détaillées ci-après.

3.3.1 Maintien de la cohérence visuelle lors de la tâche de la navigation

Dans le but d'assurer le recalage des objets virtuels lors de la navigation de l'utilisateur dans son environnement augmenté, nous avons opté pour une approche d'estimation de pose qui garantit un suivi stable et précis même si l'objet virtuel et/ou l'utilisateur sont en mouvement.

Pour ce faire, nous nous sommes basés sur l'approche PTAM (Parallel Tracking And Mapping) (Klein & Murray 2007) (voir Chapitre 1, Section 1.5.1) qui assure l'estimation de pose. Cette approche peut être décrite comme suit :

a) Initialisation du système.

Dans un premier temps, l'algorithme PTAM requière une étape d'initialisation afin de créer une carte « map » initiale. Pour ce faire, on capture deux images avec un léger déplacement. Le système détecte les points d'intérêt des deux images (PTAM utilise le détecteur FAST). A partir des informations extraites des deux images, nous pouvons par triangulation calculer notre map de base, c.-à-d. les positions 3D des points détectés.

Par la suite, au fur et à mesure que le système capture des images, la carte est enrichie par de nouveaux points. Toutefois, afin d'éviter d'alourdir le système, la mise à jour de la carte n'est pas effectuée pour chaque image capturé.

b) Suivi de l'utilisateur.

Pour le suivi, lors de l'acquisition d'une nouvelle image, une estimation apriori de la pose est faite par le biais d'un modèle de mouvement. Cette estimation est utilisée pour projeter les points 3D du map sur l'image capturée. A partir de là, on recherche ces points sur l'image à proximité du résultat de la projection, en prenant en considération les transformations que subit le patch sur l'image capturé. Les résultats permettent de mettre à jour la pose estimée. Par la suite, une re-projection puis la recherche d'un ensemble de points est effectuée dans l'image ceci permet de calculer la pose finale de la caméra.

L'estimation de pose est donc assurée sans être liée à un objet de référence. Ainsi l'utilisateur peut bouger, tout en assurant l'augmentation de son environnement. La figure 70 montre une séquence d'images d'un objet virtuel en mouvement.

Figure 70. Voiture virtuelle en mouvement.

Bien que cette approche permette d'assurer la mobilité de l'utilisateur, néanmoins le problème de l'échelle de la scène visée subsiste vu que la quantité d'information à gérer augmente avec le temps. Ainsi, les recherches dans ce domaine restent toujours ouvertes.

3.3.2 Maintien de la cohérence visuelle lors de la tâche de sélection/manipulation

Pour la deuxième approche, nous avons opté pour une technique d'estimation de pose à base d'informations coplanaires à partir de marqueurs naturels en utilisant le principe de suivi par détection. Cette approche sera utilisée pendant les tâches de sélection et de manipulation.

En effet, quand l'utilisateur souhaite interagir avec un objet virtuel proche ou distant, le système bascule vers la deuxième approche d'estimation de pose. Ainsi, le frame actuel est sauvegardé, et il est considéré comme image de référence (cible naturelle) pendant toute la tâche de sélection et de manipulation.

Pour ce faire, nous avons utilisé notre descripteur POLAR_MOBIL avec la technique Coplanar POSIT pour l'estimation de pose à partir des objets planaires. Cette technique est une version coplanaire de l'algorithme POSIT, proposée par Denis Oberkampf et associés (Oberkampf et al. 1996).

D'une manière générale, cette technique utilise au minimum quatre (4) points coplanaires et se base sur le principe de projection orthogonale afin de remédier à la non-linéarité du problème d'estimation de pose. Ensuite, itérativement on revient au système de base dit projection perspective. L'aspect itératif permet donc de converger vers des résultats précis. Et vu

que la convergence se fait en un nombre réduit d'itérations, l'algorithme reste rapide et utilisable dans des applications à temps réel.

Le calcul de la pose des objets virtuels insérés (proches et/ou distants) ainsi que la technique d'interaction proposée seront détaillés dans le chapitre suivant (chapitre 4).

Après avoir implémenté cette approche (POLAR_MOBIL + Coplanar POSIT), nous l'avons testé sur différents types d'images (cibles naturelles), en utilisant la plateforme de test décrite dans le chapitre 3, section 3.2.1.1.1. Nous avons calculé le temps moyen de l'exécution de chaque algorithme par frame, ainsi que le nombre de frames par seconde (voir tableau 9). La figure 71 ci-dessous présente quelques images d'une scène augmentée.

Tableau 9. Temps moyen de l'exécution de POLAR_MOBIL et Coplanar POSIT.

Technique	Temps moyen d'exécution d'un frame (ms)
POLAR_MOBIL	31.85
Coplanar POSIT	11.62
Temps global	43.47 **(23 frames par sec.)**

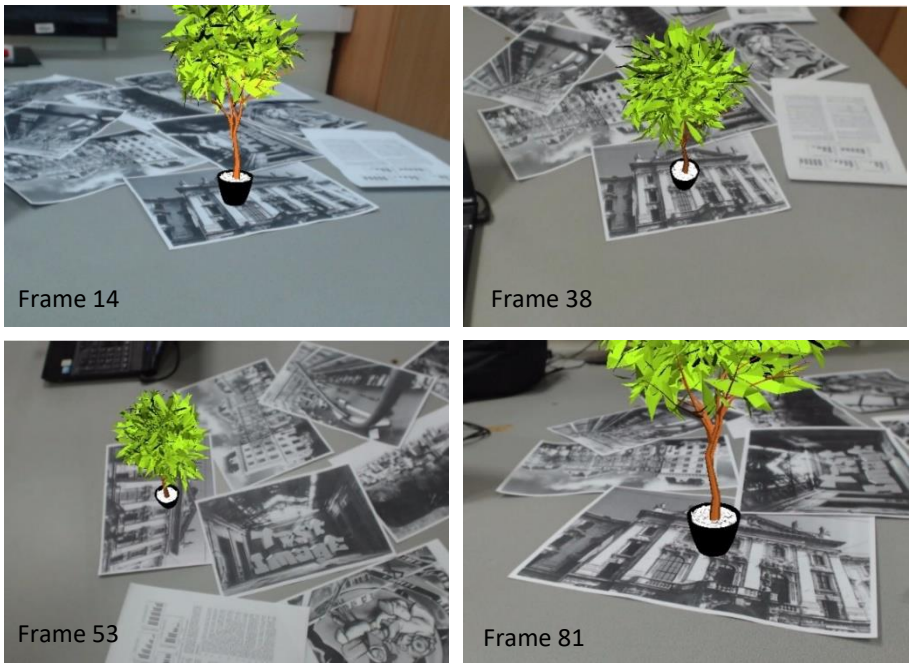


Figure 71. Exemples d'application de POLAR_MOBIL + Coplanaire POSIT en RA.

3.4 Conclusion

Nous avons abordé dans ce chapitre l'estimation de pose pour la réalité augmentée avec le concept d'immersion mobile. En vision par ordinateur, l'immersion mobile en RA consiste à : rendre possible le recalage des objets virtuels dans le monde réel quand l'utilisateur et/ou l'objet virtuel sont en mouvement.

De ce fait, nous avons présenté dans ce chapitre, la première partie de notre travail qui concerne l'estimation de pose pour le recalage des objets virtuels. Nous nous sommes penchés sur la partie détection et description des points d'intérêts dans l'image. Nous avons donc proposé un détecteur de points d'intérêt (MOBIL_Detector) qui hybride le détecteur AGAST avec la mesure de coins de Shi-Tomasi. Nous avons également proposé un descripteur binaire basé sur les moments géométriques MOBIL, ainsi que des améliorations de ce dernier, à savoir MOBIL_2B et POLAR_MOBIL. Chacune de ces propositions a montré une certaine compétitivité par rapport aux travaux existants. Les résultats obtenus présentent une bonne robustesse aux différentes variations notamment celles du point de vue, et de changement d'échelle. Et offrent aussi, une certaine rapidité en temps de calcul, qui convient aux applications temps réel de réalité augmentée.

Nous avons par la suite utilisé ce nouveau détecteur-descripteur MOBIL avec la technique PTAM afin de mettre en œuvre un système de réalité augmentée mobile. Ceci nous a permis d'assurer le recalage réel/virtuel pendant les différentes tâches d'interaction de l'utilisateur.

Après avoir traité le problème de recalage des objets virtuels, nous allons aborder dans le prochain chapitre, la partie interaction qui contribue au concept d'immersion mobile. A savoir comment interagir avec ces objets virtuels, notamment les objets distants en RA tout en conservant la cohérence visuelle ?

Chapitre IV.

Zoom-In : Contribution à l'interaction 3D en RA

4.1 Introduction

Nous avons présenté dans le précédent chapitre nos contributions relatives à l'estimation de pose en RA. A présent, nous pouvons assurer le recalage spatiotemporel des objets virtuels dans une scène réelle.

Dans ce présent chapitre, nous présentons nos travaux qui concernent l'interaction 3D en RA. A l'issue de l'état de l'art présenté dans la première partie de ce manuscrit, nous avons constaté que cette question est abordée beaucoup plus en RV qu'on RA. Nous avons également conclu que certaines contraintes d'interaction sont présentes uniquement en RA.

Dans cette optique, nous allons proposer dans ce chapitre une technique d'interaction pour la RA que nous appelons « Zoom-In » (Bellarbi et al. 2017b). Cette technique facilite et simplifie la sélection et la manipulation des objets virtuels distants en évitant les déplacements inutiles de l'utilisateur tout en respectant les contraintes de recalage de ces objets.

A cet effet, nous présentons dans ce qui suit le principe de la technique Zoom-In. Une étude expérimentale est menée à la fin de ce chapitre, afin d'évaluer cette technique d'interaction.

4.2 Zoom-In : une technique d'interaction 3D en RA basée sur le zoom de l'image.

Zoom-In (Bellarbi et al. 2017b) est une technique d'interaction hybride qui combine la métaphore de la main virtuelle (Virtual Hand) et le zoom de la caméra. Cette technique a pour objectif de faciliter la sélection et la manipulation des objets virtuels notamment les objets distants, tout en restant lié au monde réel en réalité augmentée.

Dans ce sens, cette technique repose sur l'idée que le zoom des images capturées permet de rapprocher les deux objets éloignés réels et virtuels, tout en gardant le recalage spatial entre les objets virtuels et la scène réelle.

4.2.1 Dispositif matériel

Pour ce faire, nous avons construit un prototype, qui est composé d'un contrôleur « Leap Motion » monté sur un casque de réalité augmentée vidéo see-through « Vizux 1200AR » doté d'une caméra. Pour éviter l'occultation de la main de l'utilisateur par les objets virtuels, nous avons tourné le Leap Motion de 45° vers le bas par rapport à la caméra RGB, puis nous avons aligné le contrôleur virtuel du Leap Motion avec la caméra virtuelle, afin que l'utilisateur puisse voir ses mains virtualisées sur l'affichage du casque (Bellarbi et al. 2017b) (voir figure 72).

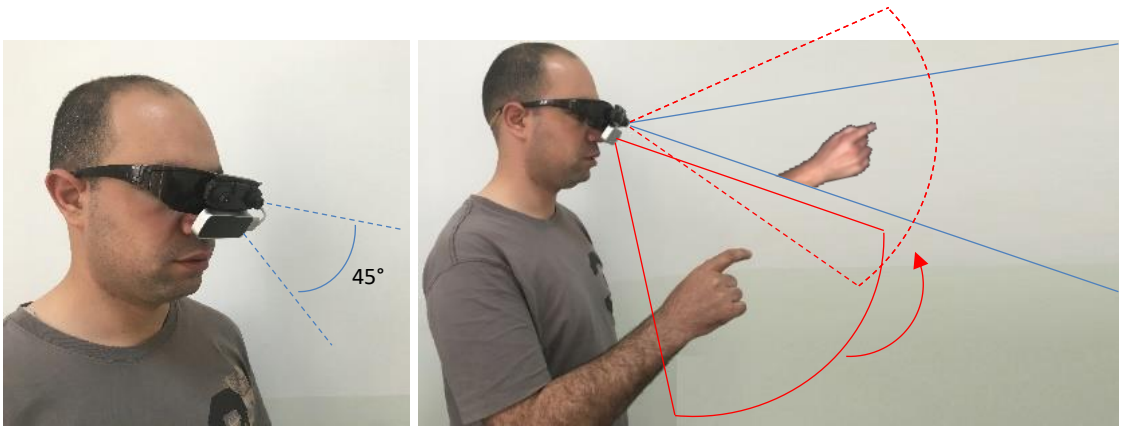


Figure 72. L'architecture du prototype de la technique Zoom-In.

4.2.2 Principe de fonctionnement

Pour sélectionner un objet distant, l'utilisateur pointe vers l'objet désiré. Un zoom de la caméra est activé (nous avons appliqué un zoom numérique sur l'image capturée), jusqu'à ce que l'objet virtuel désiré soit suffisamment proche pour être à la portée de la main de l'utilisateur. L'utilisateur peut utiliser la métaphore standard main virtuelle pour saisir et manipuler l'objet. (Figure 73). A cet effet, deux types de zoom sont proposés : zoom automatique et zoom manuel.

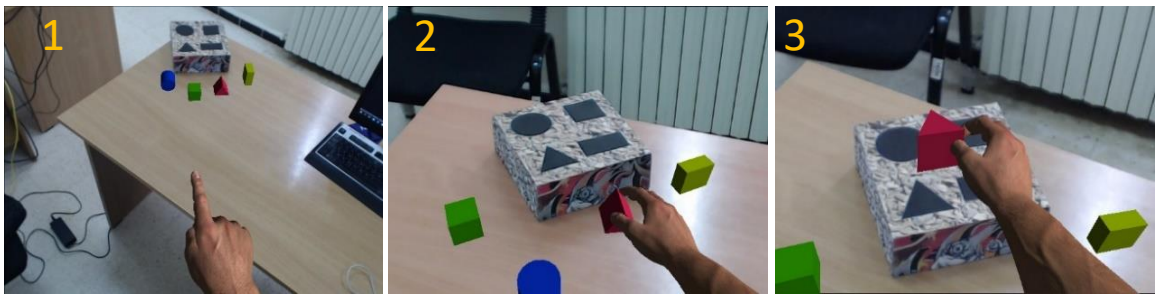


Figure 73. Déroulement de la technique d'interaction Zoom-In : 1) pointage vers l'objet désiré. 2) un zoom est appliqué sur l'image. 3) l'utilisateur utilise la main virtuelle simple pour saisir et manipuler l'objet.

4.2.2.1 Zoom-In avec un zoom automatique.

Dans le cas du zoom automatique, lorsque l'utilisateur pointe vers un objet distant en utilisant la métaphore ray-casting, le système calcule la distance entre l'objet virtuel et la main virtuelle de l'utilisateur afin d'estimer le facteur zoom « F » à appliquer sur l'image capturée et la partie de l'image de centre $Q(u, v, 1)^t$ qui doit être zoomée pour que l'objet soit à la portée de la main de l'utilisateur (Bellarbi et al. 2017c). Une fois le facteur zoom calculé, une animation zoom est effectuée. Ce principe est illustré sur la figure 74.

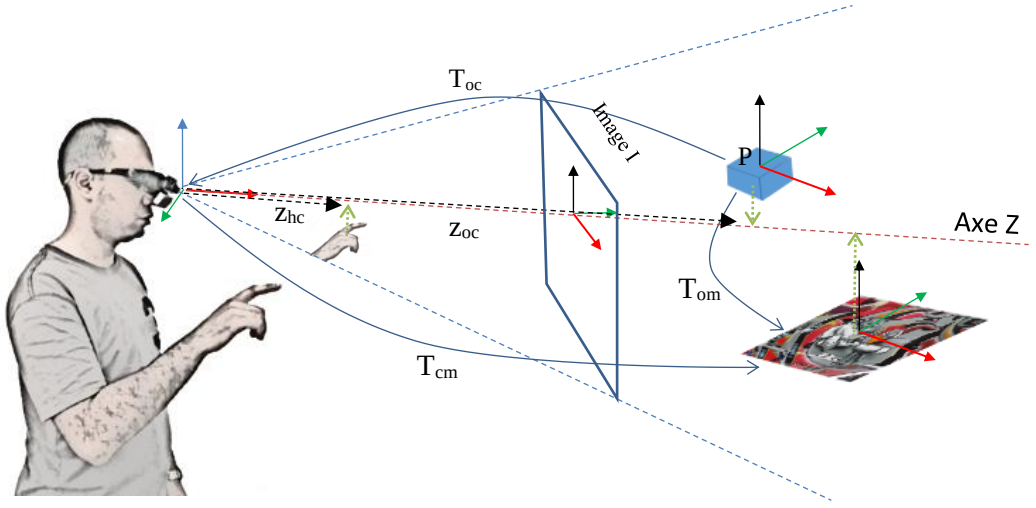


Figure 74. Principe du zoom automatique.

4.2.2.1.1 Formulation du problème

1. Calcul du facteur zoom F.

Soit T_{oc} la matrice de transformation d'un objet O par rapport à la caméra C. Cette matrice est obtenue par le produit des deux matrices T_{om} et T_{cm} qui sont les matrices de transformation Objet-Map et Map-Caméra respectivement. Soit Z_{oc} la translation de l'objet sur l'axe Z extraite de la matrice T_{oc}.

T_{hc} est la matrice de transformation de la main virtuelle par rapport à la caméra, et Z_{hc} sa translation sur l'axe Z extraite de la matrice T_{hc}.

Soit I l'image capturée par la caméra avec les dimensions L (largeur) et H (hauteur). I'(H', L') est une partie de l'image I tel que H'/L'= H/L qui représente la partie de l'image où l'objet virtuel sélectionné est projeté.

Notre objectif est de rendre l'objet sélectionné à la portée de la main de l'utilisateur c.à.d. réduire sa distance de Z_{oc} vers Z_{hc}. Ce qui se traduit par zoomer l'image I' avec le facteur $F = \frac{Z_{oc}}{Z_{hc}}$.

Une fois que le facteur zoom F est calculé, nous pouvons calculer les dimensions H' et L' respectivement par $H' = \frac{H}{F}$ et $L' = \frac{L}{F}$.

2. Calcul de la position du centre $Q(u, v)^t$ de l'image I'.

Afin de déterminer la portion de l'image à zoomer, nous projetons la position 3D $P(x, y, z)^t$ de l'objet virtuel sélectionné sur le plan image I. Ceci se traduit par l'équation suivante (équation 20).

$$sQ = s\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = A\, T_{oc}\, P\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{20}$$

Avec s facteur d'échelle et A la matrice des paramètres intrinsèques (voir chapitre 1, section 2).

Une fois que nous avons la position 2D $(u, v)^t$ de l'objet sur l'image I, nous pouvons extraire la portion de l'image I' à partir de l'image I comme suit (équations 21 et 22) et illustrée dans la figure 75.

$$p_x = u - \frac{L'}{2} \tag{21}$$

$$p_y = v - \frac{H'}{2} \tag{22}$$

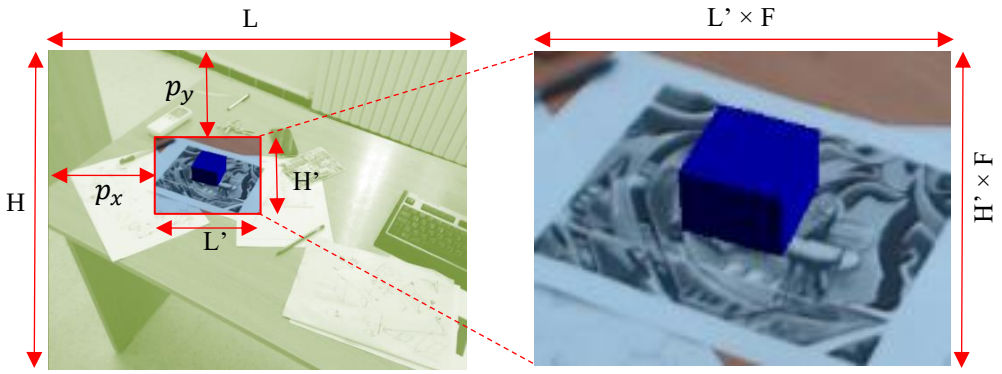


Figure 75. Extraction de la partie de l'image à zoomer.

Une fois que l'objet est à la portée de l'utilisateur, il peut utiliser la métaphore Main Virtuelle simple (VirtualHand) afin de saisir et manipuler cet objet. Cette technique permet également à l'utilisateur de sélectionner et manipuler les objets virtuels voisins, sans avoir besoin de répéter complètement le processus du zoom dès le début. Ainsi, ceci permet de sélectionner et de manipuler les objets occultés par rapport au rayon-laser.

Lorsque toutes les manipulations sont terminées, l'utilisateur effectue un geste par sa main (dans ce cas, nous avons choisi le geste Main-Ouverte ou Open Palm) afin de terminer l'opération, une animation de zoom arrière (dé-zoom) est activée pour restituer la taille réelle de l'image capturée.

4.2.2.1.2 Zoom-In avec un zoom manuel.

L'utilisateur peut également choisir l'option zoom manuel de notre technique d'interaction 3D. Dans ce cas, quand l'utilisateur effectue le geste de pointage par son index, un facteur zoom de 1.2 (ce facteur a été choisi en premier lieu arbitrairement, puis raffiné selon les réponses des participants lors de l'évaluation) est appliqué sur l'image. Le zoom est appliqué sur le centre de l'image.

Cette opération de zoom est répétée tant que le geste de pointage est toujours maintenu. Cette option de la technique proposée offre la possibilité de sélectionner et manipuler tous les objets virtuels qui sont devenus à la portée de l'utilisateur, notamment les objets occultés. En outre, l'utilisateur peut également reculer en arrière, en dé-zoomant (zoom arrière) l'image par le facteur 1/1.2 en effectuant le geste de main ouverte (Open Palm).

De même que le cas du zoom automatique, la sélection et la manipulation des objets proches et/ou objets devenus à la portée de l'utilisateur, sont faites par la métaphore de la main virtuelle simple.

4.2.2.2 Recalage des objets virtuels

Nous avons détaillé dans le chapitre précédent l'approche proposée pour assurer le recalage des objets virtuels. En effet, le descripteur POLAR_MOBIL présente une robustesse au changement d'échelle, ce qui le rend adéquat pour notre technique d'interaction.

Lors de la tâche de sélection/manipulation, l'estimation de pose est assurée par le suivi par détection (voir chapitre 3, section 3.3.) afin de prendre en compte le changement brusque de l'échelle.

A cet effet, nous considérons notre problème comme étant un problème de changement de repère, du repère de map défini par la technique PTAM lors de la tâche de navigation, au nouveau repère marqueur qui représente l'image I.

Pour ce faire, nous calculons la transformation Objet-Marqueur (soit T_{o_mq}) à partir du produit des deux transformations T_{om} et $T_{mq\_m.}$ Tel que, T_{mq_m} la transformation marqueur-map est égale à T_{cm} de frame 0 (début du zoom).

A partir de là, l'image I est considérée comme image de référence pendant toute la procédure du zoom (i.e. pendant la tâche de sélection/manipulation) sur laquelle on applique le suivi par détection. Ainsi, nous calculons l'homographie entre les deux images I et I' via POLAR_MOBIL. La pose est estimée par Coplanaire POSIT.

Enfin, la position de l'objet virtuel par rapport au nouveau repère peut être calculée en multipliant T_{o_mq} par la pose estimée.

4.3 Test et evaluation de la technique Zoom-In

Nous avons implémenté notre technique proposée sous Unity3D, version 5.3, sous les mêmes conditions décrites dans le chapitre 3, Section 3.2.1.1.1.

4.3.1 Le protocole expérimental

Ainsi, afin de tester et d'évaluer notre technique Zoom-In, nous avons développé une application de jeu éducatif dont l'objectif consiste à remettre des formes géométriques virtuelles dans leurs emplacements adéquats dans une boite réelle aussi rapidement et aussi précisément que possible. (Voir figure 76). Les objets virtuels sont quatre, avec différentes formes géométriques : Cube, Prisme, Parallélogramme et Cylindre. Les dimensions des objets varient entre 3 à 8 cm. La boite réelle est de dimensions 20x20x10 cm. Avec 4 trous de différentes formes sur la face supérieure, qui correspondent aux formes virtuelles.

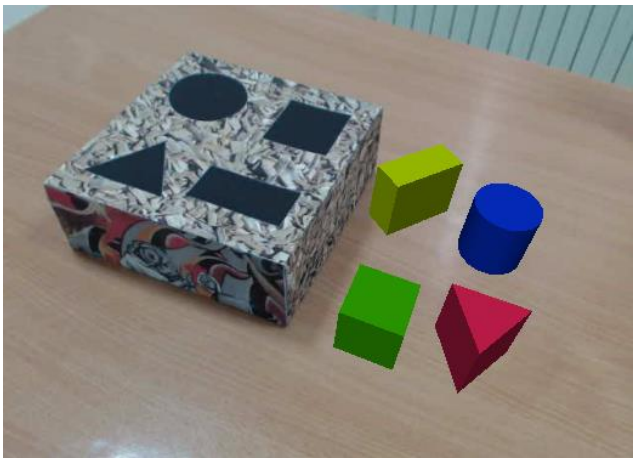


Figure 76. La boite et les formes virtuelles utilisées dans le jeu éducatif.

Le but principal de cette évaluation est de comparer objectivement la précision et le temps pour accomplir une tâche par la technique Zoom-In (avec l'option du zoom automatique) avec une technique de sélection-manipulation similaire à savoir la technique HOMER que nous avons implémentée sous Unity3D, avec les mêmes conditions que celles de la technique Zoom-In.

4.3.2 Les tâches

Nous avons défini une tâche comme la sélection et la manipulation (déplacement/rotation) d'un seul objet pour le mettre dans son emplacement adéquat. Ainsi, nous avons considéré deux types de tâches de manipulation : simples et complexes et nous avons regroupé les quatre formes géométriques en deux groupes (deux formes dans chaque groupe), selon les deux types de tâches de manipulation comme suit :

1. Une tâche manipulation simple : celle-ci concerne les deux formes : le cube et le cylindre, et ne nécessite qu'un déplacement simple de l'objet, sans rotation.
2. Une tâche manipulation complexe : celle-ci concerne les deux formes : le prisme et le parallélogramme, et nécessite deux manipulations (déplacement + rotation) pour pouvoir mettre l'objet dans sa place.

4.3.3 Les Hypothèses

Nous avons défini les trois hypothèses suivantes : Hypothèses

1. H0 : Le temps de réalisation d'une tâche donnée par la technique Zoom-In, est inférieur ou égal à celui de la technique HOMER.
2. H1 : La précision de la technique Zoom-In lors de la réalisation d'une tâche donnée est meilleure que celle de la technique HOMER.

3. H2 : L'influence de la complexité de la tâche sur la technique Zoom-In, en termes de temps de calcul et de précision, est négligeable par rapport à celle de HOMER.

4.3.4 Les participants

Pour répondre à nos hypothèses, nous avons pris un échantillon de 18 participants de différents âges (entre 22 à 38, avec une moyenne de 32.1) et genres (5 femmes, 13 hommes), et de différentes années d'expérience dans le domaine de la RA, de l'interaction 3D et des jeux. Toutes ces personnes étaient droitières et aucune d'entre elles ne souffrait de problème de vue identifié.

4.3.5 La procédure

Après avoir expliqué le principe de fonctionnement des deux techniques d'interaction Zoom-In et HOMER, ainsi que le principe du jeu, les 18 participants ont fait des tests de familiarisation de 5 à 10 mn avec chaque technique en utilisant le jeu développé.

Une fois les tests terminés, les participants ont commencé leurs expériences pour l'évaluation en utilisant le jeu développé (figure 77). Afin d'éviter un transfert d'apprentissage entre les différentes étapes, les participants n'ont pas réalisé les expériences dans le même ordre. De ce fait, nous avons divisé les participants en deux groupes égaux nommés A et B. Les expériences ont été faites par la suite comme suit :

1. Les participants du groupe A ont commencé leurs expériences d'abord par la technique Zoom-In, puis la technique HOMER.
2. Les participants du groupe B ont commencé leurs expériences d'abord par la technique HOMER puis la technique Zoom-In.
3. La boite et les formes géométriques virtuelles sont mises à une distance de 2 mètres du participant.
4. Chaque participant doit répéter chaque expérience 4 fois, pour chaque objet et avec chaque technique d'interaction.

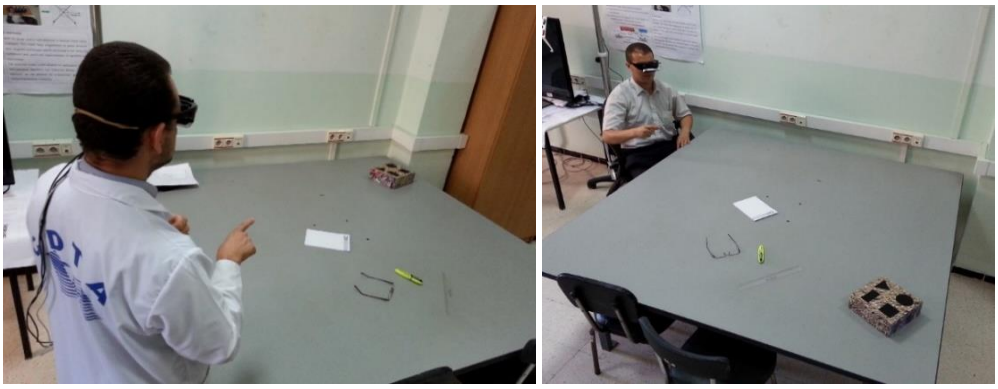


Figure 77. Des participants pendant les expériences.

Lors de chaque expérience, le système calcule le temps d'accomplissement d'une tâche (temps de sélection + temps de manipulation d'une seule forme) en secondes. Ainsi que la précision i.e. l'erreur lors du positionnement de l'objet dans son emplacement pour les deux types de tâches simples et complexes définies précédemment (section 4.3.2).

Pour les tâches simples, l'erreur de positionnement d'un objet est calculée uniquement en fonction de son déplacement selon les trois axes (x, y z), car les objets dans ce cas ont la bonne orientation. L'erreur alors, est la distance euclidienne entre le centre de gravité de l'objet et le centre de gravité de son emplacement en millimètres (Figure 78.A).

Pour les tâches complexes, nous avons calculé l'erreur de déplacement de la même manière que celle utilisée pour les tâches simples. En outre, nous avons calculé l'erreur de la rotation de l'objet selon l'axe Y (voir Figure 78.B). Le calcul de cette erreur s'est fait de la manière suivante :

1. Estimer l'angle minimal pour la rotation adéquate de l'objet.
2. Calculer la longueur de l'arc de l'angle estimé par rapport à la taille de l'objet 3D.

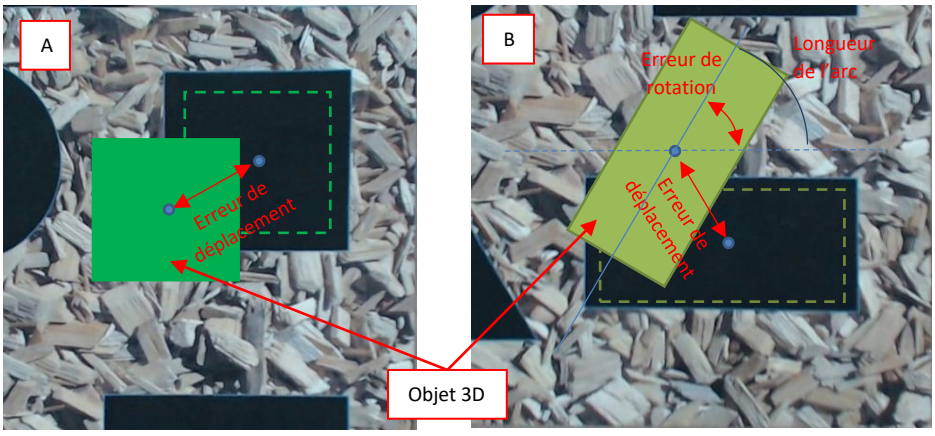


Figure 78. Calcul de l'erreur de positionnement et de rotation. A) Calcul de l'erreur dans une tâche simple. B) Calcul de l'erreur dans une tâche complexe.

A la fin des expériences, les participants ont rempli des questionnaires afin d'évaluer subjectivement les techniques d'interaction (section 4.3.7).

4.3.6 Evaluation objective

Après avoir récolté toutes les données (temps, erreurs de déplacement, et erreurs de rotation), nous avons réalisé une ANOVA à deux facteurs avec mesures répétées (Two-way ANOVA with repeated mesures) sur les données recueillies afin d'étudier l'effet de la technique d'interaction utilisée sur le temps et la précision. Ainsi que l'influence de la complexité de la tâche sur le temps et la précision de la technique d'interaction.

Les résultats obtenus sont de ($F = 18{,}321$, $p < 0{,}0012$) pour le temps d'accomplissement de la tâche et de ($f=13{,}83$, $p < 0{,}002$) pour l'erreur. Ceci révèle un effet significatif de la complexité de la tâche sur ces deux indicateurs (temps et précision).

Cependant, en analysant le graphe des moyennes de l'erreur (Figure 79), nous constatons que la technique Zoom-In est moins influencée par la complexité de la tâche par rapport à la technique HOMER. Ainsi, nous avons trouvé une erreur moyenne de 16,94 mm avec un écart-type de 4,72 pour la réalisation d'une tâche simple par la technique Zoom-In, contre une erreur moyenne de 22,66 mm avec un écart-type de 5,01 pour la technique HOMER.

En outre, nous avons révélé une erreur moyenne de 22,44 mm avec un écart-type de 4,27 pour la réalisation d'une tâche complexe par la technique Zoom-In, par rapport à une erreur moyenne de 32,88 mm avec un écart-type de 5,51 pour la technique HOMER. Ce qui signifie un effet important de la complexité de la tâche sur la technique HOMER par rapport à la technique Zoom-In.

Figure 79. Comparaison de l'influence de la complexité de la tâche sur l'erreur de la manipulation pour les techniques Zoom-In et HOMER.

D'un autre côté, le graphe des moyennes de temps, illustré par la figure 80, montre que la technique Zoom-in nécessite un temps d'interaction (21,51 sec avec un écart-type de 3,93) inférieur à celui de la technique HOMER (24,85 sec avec un écart-type 3,25) pour les tâches simples, ainsi qu'une moyenne de 24,04 sec avec un écart-type de 3,44 pour Zoom-In pour les tâches complexes par rapport à 32,90 sec avec un écart-type de 2,23 pour HOMER.

Figure 80. Comparaison de l'influence de la complexité de la tâche sur le temps de l'interaction pour les techniques Zoom-In et HOMER.

Plus précisément, nous avons analysé le temps de sélection et le temps de manipulation pour les deux techniques avec les deux types de tâches (Figure 81). Nous avons observé que pour la sélection des objets, HOMER est légèrement plus rapide que Zoom-in dans les deux cas (tâches

simple et complexes). Ceci est dû au fait que la technique HOMER permet de sélectionner l'objet directement en pointant vers lui. Tandis qu'avec la technique Zoom-In, pointer vers l'objet permet seulement de zoomer la scène. La sélection se fait après le zoom via la technique de la main virtuelle simple.

Par contre, le temps de manipulation d'un objet est plus petit avec la technique Zoom-In, surtout dans le cas des tâches complexes, du fait que la scène concernée par la tâche d'interaction est mise à la portée de l'utilisateur ce qui lui permet de manipuler l'objet plus facilement.

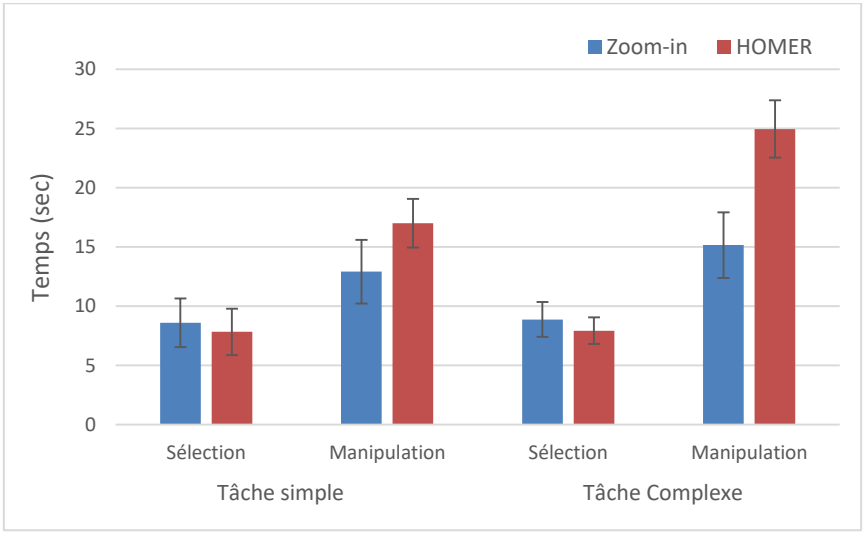


Figure 81. Comparaison du temps de sélection et de manipulation entre les deux techniques Zoom-In et HOMER.

4.3.7 Evaluation subjective

Une fois que toutes les expériences terminées, Nous avons demandé aux participants de remplir deux types de questionnaires, avec deux exemplaires chacun afin d'évaluer notre technique et de la comparer avec la technique HOMER.

4.3.7.1 Questionnaire USE

Le premier questionnaire concerne l'utilisabilité du système. Pour cela, nous avons utilisé le questionnaire USE (Usefulness, Satisfaction and Ease) (Lund 2001). Ce dernier est largement utilisé pour l'évaluation des systèmes de réalité augmentée/virtuelle et de l'interaction 3D. Il se compose de 30 questions regroupées en 4 catégories : l'utilité (Usefulness), la satisfaction (Satisfaction), la facilité de l'utilisation (Ease of Use) et la facilité de l'apprentissage (Ease of Learning). Le participant peut répondre aux questions à travers une échelle de 7 points, allant de « 1 : fortement en désaccord » à « 7 : fortement d'accord ». Une version française de ce questionnaire peut être trouvée en Annexe (Annexe A, section A.1.).

Une fois les questionnaires remplis par les participants, nous avons calculé la moyenne et l'écart type de chaque catégorie du questionnaire USE. Les résultats obtenus sont illustrés par les graphes de la figure suivante (Figure 82).

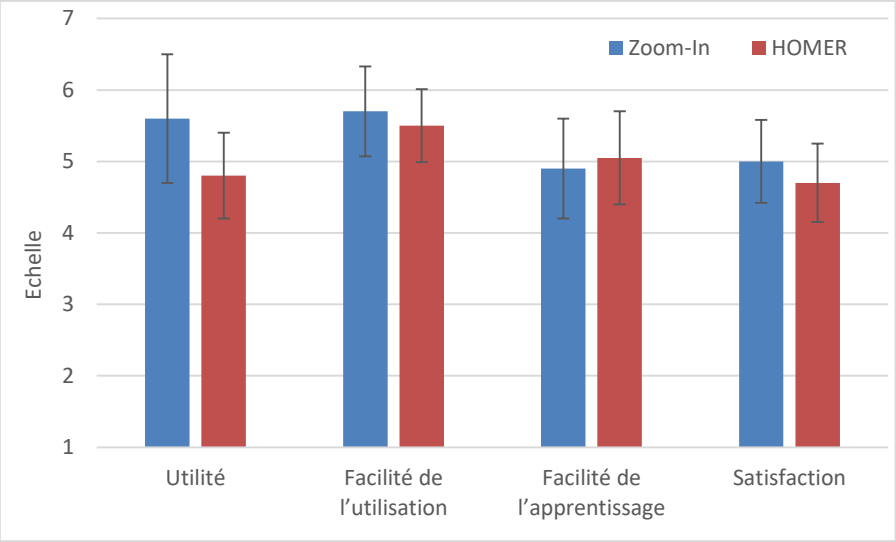


Figure 82. Résultats du questionnaire d'évaluation USE.

Nous avons remarqué que la plupart des participants ont apprécié l'utilité de la technique Zoom-In (5.6 avec un écart-type de 0.9) par rapport à la technique HOMER (4.8 avec un écart-type de 0.6). Cela est due à la précision qu'offre la technique Zoom-In lors de la manipulation des objets distants par rapport à la technique HOMER. Pour les questions concernant la facilité de l'utilisation, nous avons constaté que les réponses des participants étaient pratiquement identiques. Avec un léger dépassement de la technique Zoom-In (5.7 avec un écart-type de 0.63) par rapport à la technique HOMER (5.5 avec un écart-type de 0.5).

En revanche, les participants ont estimé que la technique HOMER est légèrement plus facile à apprendre (5.05 et un écart-type de 0.65) par rapport la technique Zoom-In (4.9 avec un écart-type de 0.7). Quant au critère de la satisfaction, la technique Zoom-In présente une moyenne de 5.0 avec un écart-type de 0.58. La technique HOMER présente une moyenne de 4.7 avec un écart-type de 0.55.

4.3.7.2 Questionnaire NASA TLX

Le NASA TLX (NASA Task Load Index) est une méthode qui permet l'évaluation subjective de la charge de travail globale. Développé au NASA AMES RESEARCH CENTER par Hart et Staveland (Hart & Staveland 1988), cet outil a été testé dans diverses conditions expérimentales : lors de simulations de vol, de simulations de contrôle de processus, ainsi que dans différentes tâches en laboratoire (tâches de calcul mental, d'imagerie mentale, d'acquisition de cible, de raisonnement grammatical, etc.).

Le questionnaire prend en compte plusieurs critères indépendants que les participants doivent évaluer en fonction de leur ressenti. Les trois premiers critères représentent les astreintes imposées au participant par la tâche (exigences physique, mentale et temporelle) et les trois autres rendent compte des interactions du participant avec la tâche (performance, effort et frustration). Ainsi, les participants peuvent répondre aux six questions à travers une échelle de 20 points, allant du très faible au très élevé. Une version française du questionnaire peut être trouvée en Annexe (Annexe A. Section A.2.).

La figure ci-dessous (figure 83) illustre les résultats (moyennes et écarts-type) des réponses des participants au questionnaire NASA-TLX.

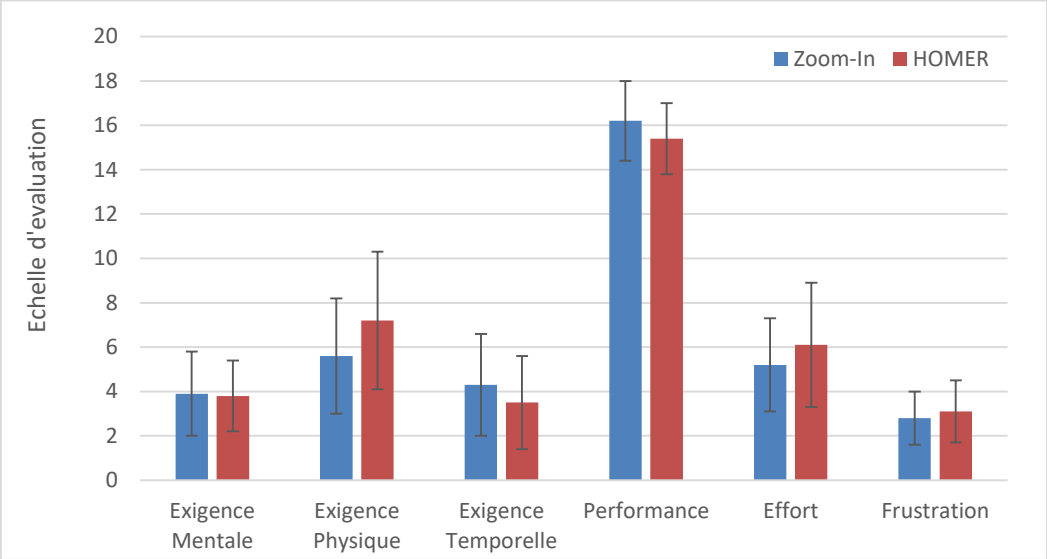


Figure 83. Résultat de l'évaluation subjective pour les techniques d'interaction Zoom-In et HOMER, en utilisant le questionnaire NASA-TLX.

Selon les résultats obtenus, nous avons remarqué que les participants ont considéré que les deux techniques Zoom-In et HOMER présentent des degrés d'exigences mentales et temporelles faibles. Avec un dépassement léger de la technique HOMER (3.8 avec un écart-type de 1.6 pour l'exigence mentale et 3.5 avec un écart-type de 2.1 pour l'exigence temporelle) par rapport à la technique Zoom-in (3.9 avec un écart-type de 1.9 pour l'exigence mentale et 4.3 avec un écart-type de 2.3 pour l'exigence temporelle).

En revanche, selon les participants, la technique Zoom-In nécessite une exigence physique (5.6 avec un écart-type de 2.6) faible par rapport à celle de HOMER (7.2 avec un écart-type de 3.1). De même pour le degré de frustration, Zoom-In présente un degré légèrement plus faible que celui de la technique HOMER. Par contre en ce qui concerne les performances, Zoom-In (16.2 avec un écart-type de 1.8) a été jugée nettement meilleure que la technique HOMER (15.2 avec un écart-type de 1.6).

4.4 Conclusion

Nous avons présenté dans ce chapitre nos contributions dans la partie interaction pour la réalité augmentée. En effet, nous avons proposé une nouvelle technique d'interaction 3D appelée « Zoom-In » basée sur le zoom de l'image avec maintien du recalage des objets virtuels. Cette technique permet de faciliter la sélection et la manipulation des objets virtuels sans que l'utilisateur n'ait à se déplacer et ce en conservant la cohérence visuelle. Elle répond également au problème d'objet occulté par d'autres objets qui empêche le rayon virtuel d'atteindre l'objet désiré.

La technique Zoom-In dépend en grande partie de l'approche utilisée pour l'estimation de pose. Dans ce sens, nos choix en matière de description de caractéristiques ou de stratégie de suivi présentés dans le chapitre 3 ont été faits relativement à la technique d'interaction 3D proposée.

Les tests effectués sur cette technique ont abouti à des résultats satisfaisants et compétitifs par rapport à la technique HOMER, aussi bien en précision qu'en temps d'accomplissement d'une tâche d'interaction.

La technique Zoom-In permet d'interagir avec des objets éloignés sans se déplacer tout en respectant le recalage spatio-temporel des objets virtuels dans la scène. Elle répond donc au problème de mobilité de l'utilisateur en lui permettant de naviguer virtuellement dans son environnement augmenté tout en conservant la cohérence visuelle.

Conclusion Générale et Perspectives

Nous avons présenté dans ce mémoire l'ensemble de nos travaux de thèse. Notre objectif était de proposer une approche basée sur l'estimation de pose et l'interaction 3D pour permettre l'immersion mobile de l'utilisateur en réalité augmentée.

En effet, nous nous sommes intéressés d'une part à :

- *Comment immerger un utilisateur dans un environnement de réalité augmentée tout en garantissant sa mobilité et d'autre part,*
- *Comment maintenir la cohérence visuelle dans l'espace et dans le temps entre l'action de l'utilisateur vers le système (en entrée) et la réaction du système vers l'utilisateur (en sortie) dans un environnement augmenté.*

Afin de répondre à cela, nous avons réalisé un état de l'art sur les travaux existants dans ce contexte. Ceci nous a permis d'orienter le reste de notre travail, en déterminant les parties ou nous devons contribuer, les techniques que nous devons utiliser et les adaptations à envisager.

Dans ce sens, nous avons dégagé deux axes principaux à savoir : l'estimation de pose et l'interaction 3D.

Dans un premier temps nous avons travaillé sur l'estimation de pose qui représente la base d'un système de réalité augmentée. Nous nous sommes donc penchés sur la partie détection et description des caractéristiques. A cet effet, nous avons donc proposé un détecteur de points d'intérêt (MOBIL_Detector) (Bellarbi et al. 2017a) qui hybride le détecteur AGAST avec la mesure de coins de Shi-Tomasi. Nous avons également proposé un descripteur binaire MOBIL (Bellarbi et al. 2014b) qui se base sur les moments géométriques, l'idée est de faire des tests binaires en les moments afin d'avoir une description robuste. Par la suite, nous avons proposé deux améliorations de ce descripteur à savoir, MOBIL_2B (Bellarbi et al. 2015) et POLAR_MOBIL (Bellarbi et al. 2017a). La première, introduit un bit supplémentaire pour la description ce qui augmente le degré de distinction du descripteur. Quant à la seconde amélioration, elle se base sur des patchs polaires qui permettent de gagner en robustesse face au changement du point de vue.

Le descripteur proposé est utilisé par la suite avec la technique PTAM afin d'assurer la cohérence spatio-temporelle des objets virtuels avec la scène réelle respectivement lors de la tâche de la sélection/manipulation et la tâche de la navigation.

D'un autre côté, nous avons abordé l'interaction 3D avec la proposition d'une nouvelle technique d'interaction pour la RA nommée « Zoom-In » (Bellarbi et al. 2017b) et qui se base sur le zoom de l'image. Cette technique permet à l'utilisateur de sélectionner et manipuler des objets virtuels sans avoir à se déplacer quand la nécessité n'y est pas. Elle utilise également la première partie de notre travail afin de garder un recalage cohérent des objets virtuels. Notons que, la technique Zoom-In ne peut être utilisée qu'avec un dispositif d'affichage vidéo see-through.

Nous avons tenté à travers nos travaux durant cette thèse, de traiter le problème d'immersion mobile en RA en proposant un détecteur-descripteur robuste qui permet d'assurer un recalage stable et précis des objets virtuels. De son côté, la technique d'interaction 3D « Zoom-In », que nous avons proposée, permet de contribuer autrement au concept d'immersion mobile. Ceci, en offrant à l'utilisateur la possibilité de naviguer virtuellement dans son environnement augmenté, en simulant son déplacement dans la scène tout en conservant la cohérence visuelle.

Ces travaux ont mené à des résultats satisfaisants lors des tests effectués. Néanmoins, des améliorations peuvent être envisagées à court terme :

- En ce qui concerne la navigation, la technique PTAM utilisée présente un manque en termes de détection et description des points d'intérêt. A cet effet, nous envisageons d'introduire dans cet algorithme notre détecteur-descripteur MOBIL.

- Pour la sélection et la manipulation, la technique Zoom-In présente des faiblesses lors d'une distance relativement grande (supérieure à quatre mètre). Ce problème est lié d'une part à la qualité de la technologie utilisée (caméra), et d'autre part à la technique de vision (MOBIL). Une solution possible est d'utiliser une caméra de meilleure résolution et d'augmenter le nombre de niveaux dans la pyramide à espace d'échelle utilisée dans notre détecteur MOBIL_Detector.

- Aussi, comme nous utilisons un HMD video-see through, il est préférable d'introduire la stéréo vision pour offrir à l'utilisateur la sensation de profondeur. De plus, utiliser une seconde caméra permet d'améliorer le système de suivi.

Par ailleurs, ce travail peut évoluer vers des applications multi-utilisateurs. L'idée est de connecter plusieurs utilisateurs co-localisés pour partager un ensemble de données extraites de l'environnement. Ainsi, la construction du map est faite en collaboration entre ces utilisateurs (Belghit et al. 2015).

L'immersion mobile en réalité augmentée est un concept très intéressant à exploiter. Nous nous sommes intéressés dans notre travail à l'aspect visuel de la RA. Toutefois, l'immersion mobile en RA ne se limite pas qu'à l'aspect visuel. Bien au contraire, elle concerne tous les sens de l'utilisateur : visuel, touché et l'ouïe. Ces derniers contribuent considérablement au degré d'immersion de l'utilisateur.

Beaucoup de verrous subsistent encore, autant sur le plan technologique que scientifique. En effet, bien que les caméras, les casques de visualisation, les dispositifs d'interactions ont évolué ces dernières années (casques Hololens, Meta2, dispositifs de reconnaissance de geste Leap motion, Realsense …). Ces technologies n'ont pas atteint leur maximum. Aussi, les recherches en termes de techniques de vision ou d'interaction doivent être approfondies en bénéficiant des technologies d'aujourd'hui qui ouvrent un autre regard sur le monde de demain qui sera peut-être un monde augmenté.

Bibliographie

Agrawal & Konolige 2008 — M. Agrawal et K. Konolige, « FrameSLAM: From Bundle Adjustment to Real-Time Visual Mapping », IEEE Trans. Robot., vol. 24, no 5, p. 1066-1077, 2008.

Alahi et al. 2012 — A. Alahi, R. Ortiz, et P. Vandergheynst, « FREAK: Fast Retina Keypoint », Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., p. 510-517, juin 2012.

Alcantarilla et al. 2012 — P. F. Alcantarilla, A. Bartoli, et A. Davison, « KAZE Features », in European Conference of Computer Vision ECCV, 2012, vol. 7577 LNCS, no PART 6, p. 214-227.

Alcantarilla et al. 2013 — P. F. Alcantarilla, J. J. Nuevo, et A. Bartoli, « Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces », in Procedings of the British Machine Vision Conference 2013, 2013, p. 13.1-13.11.

Amicis et al. 2001 — D. Amicis, M. Fiorentino, et A. Stork, « Parametric Interaction for CAD application in Virtual Reality Environment », in 12th International ADM Conference, 2001.

Andoni & Indyk 2006 — A. Andoni et P. Indyk, « Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions », in Proceedings - Annual IEEE Symposium on Foundations of Computer Science, FOCS, 2006, p. 459-468.

Argelaguet et al. 2013 — F. Argelaguet et C. Andujar, « A survey of 3D object selection techniques for virtual environments », Comput. Graph., vol. 37, no 3, p. 121-136, 2013.

Atcheson et al. 2010 — B. Atcheson, F. Heide, et W. Heidrich, « CALTag: High Precision Fiducial Markers for Caméra Calibration », in Vision, Modeling, and Visualization, 2010.

Azuma 1997 — R. Azuma, « A survey of augmented reality », Presence Teleoperators Virtual Environ., vol. 6, no 4, p. 355-385, 1997.

Azuma et al. 2001 — R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, et B. MacIntyre, « Recent advances in augmented reality », IEEE Comput. Graph. Appl., vol. 21, no 6, p. 34-47, 2001.

Bacim et al. 2014 — F. Bacim, M. Nabiyouni, et D. A. Bowman, « Slice-n-Swipe: A free-hand gesture user interface for 3D point cloud annotation », in IEEE Symposium on 3D User Interfaces 2014, 3DUI 2014 - Proceedings, 2014, vol. 185, p. 185-186.

Balntas et al. 2015 — V. Balntas, L. Tang, et K. Mikolajczyk, « BOLD - Binary online learned descriptor for efficient image matching », Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 07-12-June, p. 2367-2375, 2015.

Balntas et al. 2016 — V. Balntas, E. Johns, L. Tang, et K. Mikolajczyk, « PN-Net: Conjoined Triple Deep Network for Learning Local Image Descriptors », CoRR, vol. abs/1601.0, 2016.

Baroffio et al. 2014 — L. Baroffio, M. Cesana, A. Redondi, et M. Tagliasacchi, « Bamboo: A fast descriptor based on AsymMetric pairwise BOOsting », in 2014 IEEE International Conference on Image Processing, ICIP 2014, p. 5686-5690, 2014.

Bawa et al. 2005 — M. Bawa, T. Condie, et P. Ganesan, « LSH forest: self-tuning indexes for similarity search », Proc. 14th Int. Conf. World Wide Web - WWW '05, p. 651, 2005.

Bay et al. 2006 H. Bay, T. Tuytelaars, et L. Van Gool, « SURF: Speeded up robust features », in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 3951 LNCS, p. 404-417, 2006.

Belghit et al. 2015 H. Belghit, A. Bellarbi, N. Zenati, S. Benbelkacem, and S. Otmane. "Vision-based collaborative & mobile augmented reality." In Proceedings of the 2015 Virtual Reality International Conference, p. 23. ACM, 2015.

Belghit et al. 2012 Belghit, H., Zenati-Henda, N., Bellabi, A., Benbelkacem, S. and Belhocine, M., 2012, May. Tracking Color marker using projective transformation for augmented reality application. In Multimedia Computing and Systems (ICMCS), 2012 International Conference on (pp. 372-377). IEEE.

Bellarbi et al. 2011 A. Bellarbi, S. Benbelkacem, N. Zenati-Henda, et M. Belhocine, « Hand gesture interaction using color-based method for tabletop interfaces », in WISP 2011 - IEEE International Symposium on Intelligent Signal Processing, Proceedings, 2011, p. 180-185.

Bellarbi et al. 2012 A. Bellarbi, C. Domingues, S. Otmane, S. Benbelkacem, et A. Dinis, « Underwater augmented reality game using the DOLPHYN », in Proceedings of the 18th ACM symposium on Virtual reality software and technology - VRST '12, 2012, p. 187.

Bellarbi et al. 2013a A. Bellarbi, C. Domingues, S. Otmane, S. Benbelkacem, et A. Dinis, « Augmented reality for underwater activities with the use of the DOLPHYN », in 2013 10th IEEE International Conference on Networking, Sensing and Control, ICNSC 2013, 2013, p. 409-412.

Bellarbi et al. 2013b A. Bellarbi, H. Belghit, S. Benbelkacem, N. Zenati, et M. Belhocine, « Hand gesture recognition using contour based method for tabletop surfaces », in 10th IEEE International Conference on Networking, Sensing and Control, p. 832-836, 2013,

Bellarbi et al. 2014a A. Bellarbi, S. Benbelkacem, H. Belghit, S. Otmane, et N. Zenati, « Design and evaluation of a low-cost 3D interaction technique for wearable and handled AR devices », in 2014 4th International Conference on Image Processing Theory, Tools and Applications, IPTA 2014, 2014, p. 1-5.

Bellarbi et al. 2014b A. Bellarbi, S. Otmane, N. Zenati, et S. Benbelkacem, « MOBIL: A moments based local binary descriptor », in IEEE International Symposium on Mixed and Augmented Reality, ISMAR, 2014, p. 251-252.

Bellarbi et al. 2015 A. Bellarbi, N. Zenati-Henda, H. Belghit, M. Hamidia, S. Benbelkacem, et S. Otmane, « An Improved MOBIL Descriptor for Markerless Augmented Reality », in 3rd International Conference on Control, Engineering and Information Technology, CEIT 2015, 2015.

Bellarbi et al. 2017a A. Bellarbi, N. Zenati, S. Otmane et H. Belghit; Learning moment-based fast local binary descriptor. J. Electron. Imaging. 0001; 26(2):023006. doi:10.1117/1.JEI.26.2.023006.

Bellarbi et al. 2017b A. Bellarbi, N. Zenati, S. Otmane, et H. Belghit, "Design and Evaluation of Zoom-based 3D Interaction Technique for Augmented Reality", In Proceedings of the 2017 ACM Virtual Reality International Conference VRIC 2017.

Benbelkacem et al. 2011 S. Benbelkacem et al., « Augmented reality platform for collaborative E-maintenance systems », Andrew Yeh Ching Nee--InTech, Augmented reality--some Emerg. Appl. areas, p. 211-226, déc. 2011.

Benbelkacem et al. 2013 — S. Benbelkacem, M. Belhocine, A. Bellarbi, N. Zenati-Henda, et M. Tadjine, « Augmented reality for photovoltaic pumping systems maintenance tasks », Renew. Energy, vol. 55, p. 428-437, 2013.

Benbelkacem et al. 2012 — Benbelkacem, S., Zenati-Henda, N., Belhocine, M., Bellarbi, A. and Tadjine, M., 2012, May. Interactive space for management of documents in a maintenance context. In Multimedia Computing and Systems (ICMCS), 2012 International Conference on (pp. 378-383). IEEE.

Benbelkacem et al. 2015 — Benbelkacem, S., Zenati-Henda, N., Belghit, H., Bellarbi, A. and Otmane, S., 2015, May. Extended web services for remote collaborative manipulation in distributed augmented reality. In Control, Engineering & Information Technology (CEIT), 2015 3rd International Conference on (pp. 1-5). IEEE.

Bencina & Kaltenbrunner 2005 — R. Bencina et M. Kaltenbrunner, « The Design and Evolution of Fiducials for the reacTIVision System », Interfaces (Providence)., 2005.

Benhimane & Malis 2004 — S. Benhimane et E. Malis, « Real-time image-based suivi of planes using efficient second-order minimization », 2004 IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IEEE Cat. No.04CH37566), vol. 1, p. 943-948, 2004.

Bikos et al. 2016 — M. Bikos, Y. Itoh, G. Klinker, et K. Moustakas, « An Interactive Augmented Reality Chess Game Using Bare-Hand Pinch Gestures », Proc. - 2015 Int. Conf. Cyberworlds, CW 2015, p. 355-358, 2016.

Billinghurst et al. 2015 — M. Billinghurst, A. Clark, et G. Lee, « A Survey of Augmented Reality », Found. Trends Human-Computer Interact., vol. 8, no 2-3, p. 73-272, 2015.

Bolt 1980 — R. A. Bolt, « "Put-that-there": Voice and Gesture at the Graphics Interface », Proc. 7th Annu. Conf. Comput. Graph. Interact. Tech. - SIGGRAPH '80, vol. 14, no 3, p. 262-270, 1980.

Boulanger 2004 — P. Boulanger, « Application of augmented reality to industrial Tele-training », in Proceedings - 1st Canadian Conference on Computer and Robot Vision, 2004, p. 320-328.

Bowman 1999 — D. A. Bowman, « Interaction Techniques for Common Tasks in Immersive Virtual Environments », These de doctorat, Georgia Institute of Technology, 1999.

Bowman et al. 2002 — D. A. Bowman, J. L. Gabbard, et D. Hix, « A Survey of Usability Evaluation in Virtual Environments: Classification and Comparison of Methods », Presence Teleoperators Virtual Environ., vol. 11, no 4, p. 404-424, août 2002.

Buddemeier & Hartmut 2008 — U. Buddemeier et N. Hartmut, « Systems and methods for descriptor vector computation », 2008.

Caggianese et al. 2016 — G. Caggianese, L. Gallo, et P. Neroni, « An Investigation of Leap Motion Based 3D Manipulation Techniques for Use in Egocentric Viewpoint », Springer International Publishing, 2016, p. 318-330.

Calonder et al. 2010 — M. Calonder, V. Lepetit, C. Strecha, et P. Fua, « BRIEF: Binary robust independent elementary features », Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 6314 LNCS, no PART 4, p. 778-792, 2010.

Caudell & Mizell 1992 — T. P. Caudell et D. W. Mizell, « Augmented reality: an application of heads-up display technology to manual manufacturing processes », in Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences, 1992, vol. ii, p. 659-669 vol.2.

Chen & Hsieh 2015 — C. C. Chen et S. L. Hsieh, « Using binarization and hashing for efficient SIFT matching », J. Vis. Commun. Image Represent., vol. 30, p. 86-93, 2015.

Chen & Tseng 2007 — Y. T. Chen et K. T. Tseng, « Multiple-angle hand gesture recognition by fusing SVM classifiers », in Proceedings of the 3rd IEEE International Conference on Automation Science and Engineering, IEEE CASE 2007, 2007, p. 527-530.

Cheng et al. 2016 — H. Cheng, L. Yang, et Z. Liu, « Survey on 3D Hand Gesture Recognition », IEEE Trans. Circuits Syst. Video Technol., vol. 26, no 9, p. 1659-1673, sept. 2016.

Comport et al. 2006 — A. I. Comport, E. Marchand, M. Pressigout, et F. Chaumette, « Real-time markerless suivi for augmented reality: The virtual visual servoing framework », in IEEE Transactions on Visualization and Computer Graphics, 2006, vol. 12, no 4, p. 615-628.

Davis et al. 2016 — M. M. Davis, D. Gracanin, V. Tech, J. L. Gabbard, D. A. Bowman, et D. Gracanin, « Depth-based 3D Gesture Multi-Level Radial Menu for Virtual Object Manipulation », in 2016 IEEE Virtual Reality (VR), 2016, p. 169-170.

Davison 2003 — A. J. Davison, « Real-time Simultaneous Localisation and Mapping with a Single Caméra », ICCV, vol. 2, p. 1403-1410, 2003.

Dementhon & Davis 1995 — D. F. Dementhon et L. S. Davis, « Model-based object pose in 25 lines of code », Int. J. Comput. Vis., vol. 15, no 1-2, p. 123-141, juin 1995.

Desai et al. 2014 — A. Desai, D. J. Lee, et C. Wilson, « Using affine features for an efficient binary feature descriptor », in Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation, 2014, p. 49-52.

Desai et al. 2016 — A. Desai, D. J. Lee, et D. Ventura, « An efficient feature descriptor based on synthetic basis functions and uniqueness matching strategy », Comput. Vis. Image Underst., vol. 142, no C, p. 37-49, janv. 2016.

DeTone et al. 2016 — D. DeTone, T. Malisiewicz, et A. Rabinovich, « Deep Image Homography Estimation », in RSS Workshop on Limits and Potentials of Deep Learning in Robotics, 2016.

Didier 2005 — J.-Y. Didier, « Contributions to the dexterity of an augmented reality system applied to industrial maintenance », These de doctorat, Université d'Evry Val d'Essonne, 2005.

Didier et al. 2005 — J.-Y. Didier et al., « AMRA : Augmented Reality assistance in train maintenance tasks », in 4th ACM/IEEE International Symposium on Mixed and Augmented Reality (ISMAR) - Workshop Industrial Augmented Reality, 2005, p. 1-10.

Ding et al. 2011 — Y. Ding, H. Pang, X. Wu, et J. Lan, « Recognition of hand-gestures using improved local binary pattern », in 2011 International Conference on Multimedia Technology, ICMT 2011, 2011, p. 3171-3174.

Djelil et al. 2013 — F. Djelil, S. Otmane, et S. Wu, « Apport des NUIs pour les applications de réalité virtuelle et augmentée: état de l'art », in 8emes journées de l'AFRV, Laval, 2013.

Doretto & Yao 2010 — G. Doretto et Y. Yao, « Region moments: Fast invariant descriptors for detecting small image structures », in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, p. 3019-3026.

Du et al. 2016 — L. Du et al., « Airtouch », in Proceedings of the 53rd Annual Design Automation Conference on - DAC '16, 2016, p. 1-6.

Dubois 2009 — E. Dubois, « Conception, Implémentation et Evaluation de Systèmes Interactifs Mixtes: une Approche basée Modèles et centrée sur l'Interaction », Thèse HDR. Univ. Toulouse, 2009.

Eade & Drummond 2006 — E. Eade et T. Drummond, « Scalable monocular SLAM », in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, vol. 1, p. 469-476.

Engel et al. 2014 — J. Engel, T. Schops, et D. Cremers, « LSD-SLAM: Large-Scale Direct monocular SLAM », in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2014, vol. 8690 LNCS, no PART 2, p. 834-849.

Fahn & Chu 2011 — C.-S. Fahn et K.-Y. Chu, « Hidden-Markov-model-based hand gesture recognition techniques used for a human-robot interaction system », in 14th International Conference on Human-Computer Interaction, HCI International 2011, 2011, vol. 6762 LNCS, no PART 2, p. 248-258.

Farabet et al. 2013 — C. Farabet, C. Couprie, L. Najman, et Y. LeCun, « Learning hierarchical features for scene labeling », IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no 8, p. 1915-1929, août 2013.

Faugeras & Lustman 1988 — O. D. Faugeras et F. Lustman, « Motion and structure from motion in a piecewise planar environment », Int. J. Pattern Recognit. Artif. Intell., vol. 2, no 3, p. 485-508, 1988.

Fawcett 2004 — T. Fawcett, "ROC graphs: Notes and practical considerations for researchers". Machine learning, 31(1), 2004. pp.1-38.

Fiala 2004 — M. Fiala, « Artag revision 1, a fiducial marker system using digital techniques », 2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., no November, p. 590-596, 2004.

Flusser et al. 2009 — J. Flusser, T. Suk, et B. Zitová, « Moments and Moment Invariants in Pattern Recognition ». John Wiley & Sons, 2009.

Forsberg et al. 1996 — A. Forsberg, K. Herndon, et R. Zeleznik, « Aperture based selection for immersive virtual environments », in Proceedings of the 9th annual ACM symposium on user interface software and technology, 1996, p. 95-96.

Fuchs & Moreau 2003 — P. Fuchs et G. Moreau, "Le traité de la réalité virtuelle". Volume 1, Fondements et interfaces comportementales. Les Presses de l'École des Mines, 2003.

Fuchs & Moreau 2006 — P. Fuchs et G. Moreau, Le traité de la réalité virtuelle. Volume 2, Les Presses de l'École des Mines, 2006.

Fuchs et al. 2003 — P. Fuchs, B. Arnaldi, et J. Tisseau, « La réalité virtuelle et ses applications », in Le traité de la Réalité Virtuelle Volume 1, 2003, p. 3-52.

Gionis et al. 1999 — A. Gionis, P. Indyk, et R. Motwani, « Similarity Search in High Dimensions via Hashing », VLDB '99 Proc. 25th Int. Conf. Very Large Data Bases, vol. 99, no 1, p. 518-529, 1999.

Gong et al. 2013 — Y. Gong, S. Kumar, H. A. Rowley, et S. Lazebnik, « Learning binary codes for high-dimensional data using bilinear projections », in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2013, p. 484-491.

Griffin et al. 2007 — G. Griffin, a Holub, et P. Perona, « Caltech-256 object category dataset », Caltech mimeo, vol. 11, no 1, p. 20, 2007.

Ha et al. 2014 — T. Ha, S. Feiner, et W. Woo, « WeARHand: Head-Worn, RGB-D Camera-Based, Bare-Hand; User Interface with Visually Enhanced Depth Perception », in IEEE International Symposium on Mixed and Augmented Reality (ISMAR), 2014, pp. 219-228.

Hachet 2003 — M. Hachet, « Interaction avec des environnements virtuels affichés au moyen d'interfaces de visualisation collective » Thèse de doctorat, Bordeaux 1, 2003.

Hamidia et al. 2014 — M. Hamidia, N. Zenati-Henda, H. Belghit, et M. Belhocine, « Markerless tracking using interest window for augmented reality applications », in International Conference on Multimedia Computing and Systems -Proceedings, 2014, p. 20-25.

Han & Zhao 2016 — P. Han et G. Zhao, « L-split marker for augmented reality in aircraft assembly », Opt. Eng., vol. 55, no 4, p. 43110, 2016.

Haouchine et al. 2013a — N. Haouchine, J. Dequidt, I. Peterlik, E. Kerrien, M. O. Berger, et S. Cotin, « Image-guided simulation of heterogeneous tissue deformation for augmented reality during hepatic surgery », in 2013 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2013, 2013, p. 199-208.

Haouchine et al. 2013b — N. Haouchine, J. Dequidt, M. O. Berger, et S. Cotin, « Deformation-based augmented reality for hepatic surgery », in Studies in Health Technology and Informatics, 2013, vol. 184, p. 182-188.

Harris & Stephens 1988 — C. Harris et M. Stephens, « A Combined Corner and Edge Detector », in Procedings of the Alvey Vision Conference 1988, 1988, p. 147-151.

Hart & Staveland 1988 — S.G. Hart, and L.E. Staveland, "Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research". Advances in psychology, 52, pp.139-183. 1988.

Hartley & Zisserman 2005 — R. Hartley et A. Zisserman, "Multiple View Geometry In Computer Vision", vol. 23, no 2. Cambridge University Press, 2005.

Hasan & Abdul-Kareem 2014 — H. Hasan et S. Abdul-Kareem, « Static hand gesture recognition using neural networks », Artif. Intell. Rev., vol. 41, no 2, p. 147-181, févr. 2014.

He et al. 2013 — K. He, F. Wen, et J. Sun, « K-means hashing: An affinity-preserving quantization method for learning binary compact codes », in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2013, p. 2938-2945.

Heise et al. 2015 — P. Heise, B. Jensen, S. Klose, et A. Knoll, « Fast dense stereo correspondences by binary locality sensitive hashing », in Proceedings - IEEE International Conference on Robotics and Automation, 2015, vol. 2015-June, no June, p. 105-110.

Hirokazu & Billinghurst 1999 — K. Hirokazu et M. Billinghurst, « Marker suivi and HMD calibration for a video-based augmented reality conferencing system », Proc. 2nd IEEE ACM Int. Work. Augment. Real., p. 85-94, 1999.

Hix & Hartson 1993 — D. Hix et H. R. Hartson, Developing user interfaces : ensuring usability through product & process. John Wiley & Sons, Inc. New York, NY, USA, 1993.

Holz et al. 2012 — D. Holz, K. Hay, et M. Buckwald, « Electronic sensor », US Patent., 2012.

Hu 1962 — M.-K. Hu, « Visual pattern recognition by moment invariants », IRE Trans. Inf. Theory, vol. 8, p. 179-187, 1962.

Hua et al. 2007 G. Hua, M. Brown, et S. Winder, « Discriminant embedding for local image descriptors », Proc. IEEE Int. Conf. Comput. Vis., 2007.

Huang et al. 2011 D. Y. Huang, W. C. Hu, et S. H. Chang, « Gabor filter-based hand-pose angle estimation for hand gesture recognition under varying illumination », Expert Syst. Appl., vol. 38, no 5, p. 6031-6042, 2011.

Hugues 2011 O. Hugues, « Réalité augmentée pour l'aide à la navigation. SIGMA : Système d'information Géographique Maritime Augmentée », Thèse de doctorat, Université Sciences et Technologies - Bordeaux I, 2011.

Iwamoto et al. 2013 K. Iwamoto, R. Mase, et T. Nomura, « BRIGHT: A scalable and compact binary descriptor for low-latency and high accuracy object identification », in 2013 IEEE International Conference on Image Processing, 2013, p. 2915-2919.

Izadi et al. 2011 S. Izadi et al., « KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Caméra », Proc. 24th Annu. ACM Symp. User Interface Softw. Technol., p. 559–568, 2011.

Jacoby et al. 1994 R. H. Jacoby, M. Ferneau, et J. Humphries, « Gestural interaction in a virtual environment », in Proceedings of SPIE, 1994, vol. 2177, p. 355-364.

Jankowski & Hachet 2013 J. Jankowski et M. Hachet, « A Survey of Interaction Techniques for Interactive 3D Environments », EUROGRAPHICS 2013 State Art Reports, p. 65-93, 2013.

Ji et al. 2012 R. Ji, L. Y. Duan, J. Chen, et W. Gao, « Towards compact topical descriptors », in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2012, p. 2925-2932.

Jinjun et al. 2009 R. Jinjun, G. Tongyue, G. Zhenbang, et J. Zhen, « Low Cost Hand Gesture Learning and Recognition System Based on Hidden Markov Model », in Proceedings of the Second International Symposium on Information Science and Engineering, 2009, p. 433-438.

Jinwook et al. 2016 J. Shim, Y. Yang, N. Kang, Jonghoon Seo, and Tack-Don Han. "Gesture-based interactive augmented reality content authoring system using HMD." Virtual Reality 20, no. 1 (2016): 57-69.

Johnson et al. 2005 N. L. Johnson, A. W. Kemp, et S. Kotz, « Families of Discrete Distributions », in Univariate Discrete Distributions, 2005, p. 74-107.

Kan et al. 2009 T.-W. Kan, C.-H. Teng, et W.-S. Chou, « Applying QR code in augmented reality applications », Proc. 8th Int. Conf. Virtual Real. Contin. its Appl. Ind. VRCAI 09, vol. 1, no 212, p. 253, 2009.

Karakasis et al. 2015 E. G. Karakasis, A. Amanatiadis, A. Gasteratos, et S. A. Chatzichristofis, « Image moment invariants as local features for content based image retrieval using the Bag-of-Visual-Words model », Pattern Recognit. Lett., vol. 55, p. 22-27, 2015.

Katz 2007 V. J. Katz, "The Mathematics of Egypt, Mesopotamia, China, India, and Islam: A Sourcebook". Princeton University Press, 2007.

Katzakis et al. 2015 N. Katzakis, R. J. Teather, K. Kiyokawa, et H. Takemura, « INSPECT: extending plane-casting for 6-DOF control », Human-centric Comput. Inf. Sci., vol. 5, no 1, p. 22, 2015.

Ke & Sukthankar 2004 — Y. K. Y. Ke et R. Sukthankar, « PCA-SIFT: a more distinctive representation for local image descriptors », Proc. 2004 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, 2004. CVPR 2004., vol. 2, p. 2-9, 2004.

Kent et al. 1955 — A. Kent, M. M. Berry, F. U. Luehrs, et J. W. Perry, « Machine literature searching VIII. Operational criteria for designing information retrieval systems », Am. Doc., vol. 6, no 2, p. 93-101, 1955.

Keskin et al. 2011 — C. Keskin, A. T. Cemgil, et L. Akarun, « DTW based clustering to improve hand gesture recognition », in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2011, vol. 7065 LNCS, p. 72-81.

Keskin et al. 2012 — C. Keskin, F. Kiraç, Y. E. Kara, et L. Akarun, « Randomized decision forests for static and dynamic hand shape classification », in IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2012, p. 31-36.

Khamis et al. 2015 — S. Khamis, J. Taylor, J. Shotton, C. Keskin, S. Izadi, et A. Fitzgibbon, « Learning an efficient model of hand shape variation from depth images », in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, vol. 07-12-June, p. 2540-2548.

Kim & Choi 2015 — S. Kim et S. Choi, « Bilinear random projections for locality-sensitive binary codes », in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, vol. 07-12-June, p. 1338-1346.

Kim & Lee 2016 — M. Kim et J. Y. Lee, « Touch and hand gesture-based interactions for directly manipulating 3D virtual objects in mobile augmented reality », Multimed. Tools Appl., p. 1-22, 2016.

Klein & Murray 2007 — G. Klein et D. Murray, « Parallel tracking and mapping for small AR workspaces », in 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR, 2007, p. 1-10.

Klein et al. 2009 — G. Klein et D. Murray, « Parallel suivi and mapping on a caméra phone », in Science and Technology Proceedings - IEEE 2009 International Symposium on Mixed and Augmented Reality, ISMAR 2009, 2009, p. 83-86.

Kneip et al. 2011 — L. Kneip, D. Scaramuzza, et R. Siegwart, « A novel parametrization of the perspective-three-point problem for a direct computation of absolute caméra position and orientation », in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2011, p. 2969-2976.

Kneip et al. 2013 — L. Kneip, P. Furgale, et R. Siegwart, « Using multi-caméra systems in robotics: Efficient solutions to the NPnP problem », in Proceedings - IEEE International Conference on Robotics and Automation, 2013, p. 3770-3776.

Kneip et al. 2014 — L. Kneip, H. Li, et Y. Seo, « UPnP: An optimal O(n) solution to the absolute pose problem with universal applicability », in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2014, vol. 8689 LNCS, no PART 1, p. 127-142.

Kordelas & Daras 2009 — G. Kordelas et P. Daras, « Robust SIFT-based feature matching using Kendall's rank correlation measure », in Proceedings - International Conference on Image Processing, ICIP, 2009, p. 325-328.

LaViola 1999 — Joseph J. LaViola Jr, « A Survey of Hand Posture and Gesture Recognition Techniques and Technology », Brown University, 1999.

Lee et al. 1999 — H.-K. Lee et J. H. Kim, « An HMM-based threshold model approach for gesture recognition », IEEE Trans. Pattern Anal. Mach. Intell., vol. 21, no 10, p. 961-973, 1999.

Lepetit 2001 — V. Lepetit, « Gestion des occultations en réalité augmentée », Thèse de doctorat, Université de Nancy, 2001.

Lepetit et al. 2009 — V. Lepetit, F. Moreno-Noguer, et P. Fua, « EPnP: An accurate O(n) solution to the PnP problem », Int. J. Comput. Vis., vol. 81, no 2, p. 155-166, févr. 2009.

Leutenegger et al. 2011 — S. Leutenegger, M. Chli, et R. Y. Siegwart, « BRISK: Binary Robust invariant scalable keypoints », in Proceedings of the IEEE International Conference on Computer Vision ICCV, 2011, p. 2548-2555.

Levi & Hassner 2016 — G. Levi et T. Hassner, « LATCH: Learned arrangements of three patch codes », in IEEE Winter Conference on Applications of Computer Vision, WACV 2016, 2016.

Li et al. 2013 — X. Li, C. Shen, A. Dick, et A. Van Den Hengel, « Learning compact binary codes for visual Tracking », in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2013, p. 2419-2426.

Li et al. 2015 — Y. Li, S. Wang, Q. Tian, et X. Ding, « A survey of recent advances in visual feature detection », Neurocomputing, 2015.

Liang & Green 1994 — J. Liang et M. Green, « JDCAD: A highly interactive 3D modeling system », Comput. Graph., vol. 18, no 4, p. 499-506, juill. 1994.

Liao et al. 2013 — K. Liao, G. Liu, et Y. Hui, « An improvement to the SIFT descriptor for image representation and matching », Pattern Recognit. Lett., vol. 34, no 11, p. 1211-1220, 2013.

Lien et al. 2016 — J. Lien et al., « Soli : Ubiquitous Gesture Sensing with Millimeter Wave Radar », ACM Trans. Graph., vol. 142, no July, p. 1-19, juill. 2016.

Lim& Kim 2012 — C. J. Lim et D. Kim, « Development of gaze tracking interface for controlling 3D contents », Sensors Actuators, A Phys., vol. 185, p. 151-159, 2012.

Lin et al. 2013 — W.-S. Lin, Y.-L. Wu, W.-C. Hung, et C.-Y. Tang, « A Study of Real-Time Hand Gesture Recognition Using SIFT on Binary Images », Smart Innovation, Systems and Technologies, vol. 21. Springer Berlin Heidelberg, p. 235-246, 2013.

Lin et al. 2016 — K. Lin, J. Lu, C.-S. Chen, et J. Zhou, « Learning Compact Binary Descriptors with Unsupervised Deep Neural Networks », in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

Liong et al. 2015 — V. E. Liong, J. Lu, G. Wang, P. Moulin, et J. Zhou, « Deep hashing for compact binary codes learning », Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 07-12-June, p. 2475-2483, juin 2015.

Liu et al. 2012 — Y. Liu, Y. Yin, et S. Zhang, « Hand Gesture Recognition Based on Hu Moments in Interaction of Virtual Reality », in International Conference on Intelligent Human-Machine Systems and Cybernetics, 2012, p. 145-148.

Liu et al. 2016 — L. Liu, M. Yu, et L. Shao, « Projection bank: From high-dimensional data to medium-length binary codes », in Proceedings of the IEEE International Conference on Computer Vision, 2016, vol. 11-18-Dece, p. 2821-2829.

Lowe 1999 — D. Lowe, « Object recognition from local scale-invariant features », Proc. Seventh IEEE Int. Conf. Comput. Vis., vol. 2, 1999.

Lowe 2004 — D. Lowe, « Distinctive image features from scale-invariant keypoints », Int. J. Comput. Vis., vol. 60, no 2, p. 91-110, nov. 2004.

Lu et al. 2000 — C. P. Lu, G. D. Hager, et E. Mjolsness, « Fast and globally convergent pose estimation from video images », IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no 6, p. 610-622, juin 2000.

Lucas & Kanade 1981 — B. D. Lucas et T. Kanade, « An Iterative Image Registration Technique with an Application to Stereo Vision », in Ijcai, 1981, vol. 130, p. 674-679.

Lund 2001 — A. M. Lund, « Measuring Usability with the USE Questionnaire », Usability User Exp., vol. 8, no 2, 2001.

Mackay 1998 — W. E. Mackay, « Augmented reality: linking real and virtual worlds: a new paradigm for interacting with computers », Proc. Work. Conf. Adv. Vis. interfaces - AVI '98, p. 13, 1998.

Madeo & Bober 2016 — S. Madeo et M. Bober, « Fast, Compact and Discriminative: Evaluation of Binary Descriptors for Mobile Applications », IEEE Trans. Multimed., vol. PP, no 99, p. 1-1, 2016.

Mair et al. 2010 — E. Mair, G. D. Hager, D. Burschka, M. Suppa, et G. Hirzinger, « Adaptive and generic corner detection based on the accelerated segment test », in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2010, vol. 6312 LNCS, no PART 2, p. 183-196.

Malik et al. 2002 — D. Malik, G. Roth, et C. McDonald, « Robust Corner tracking for Real-Time Augmented Reality », in Proceedings of Vision Interface (VI), 2002, p. 399-406.

Marchand et al. 2016 — E. Marchand, H. Uchiyama, F. Spindler, « Pose estimation for augmented reality : a hands-on survey Pose estimation for augmented reality : a hands-on survey », IEEE Trans. Vis. Comput. Graph., 2016.

Martin & Durand 2000 — J. Martin et J.-B. Durand, « Automatic handwriting gestures recognition using hidden Markov models », Autom. Face Gesture Recognit, p. 403-409, 2000.

Melax et al. 2013 — S. Melax, L. Keselman, et S. Orsten, « Dynamics based 3D skeletal hand tracking », Graph. Interface Conf., p. 63-70, 2013.

Messaci et al. 2015 — A. Messaci, N. Zenati, A. Bellarbi, et M. Belhocine, « 3D Interaction techniques using gestures recognition in virtual environment », 2015 4th Int. Conf. Electr. Eng., no ii, p. 2-6, déc. 2015.

Mikolajczyk et al. 2005 — K. Mikolajczyk et C. Schmid, « A performance evaluation of local descriptors », IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no 10, p. 1615-1630, oct. 2005.

Miksik & Mikolajczyk 2012 — O. Miksik et K. Mikolajczyk, « Evaluation of local detectors and descriptors for fast feature matching », Pattern Recognit. (ICPR), 2012 21st, p. 2681-2684, 2012.

Milgram & Kishino 1994 — P. Milgram et F. Kishino, « Taxonomy of mixed reality visual displays », IEICE Trans. Inf. Syst., vol. E77-D, no 12, p. 1321-1329, 1994.

Mine 1995 — M. R. Mine, « Virtual Environment Interaction Techniques », Proc. ACM SIGGRAPH Int. Conf. Virtual Real. Contin. its Appl. Ind., p. 120-126, 1995.

Mine 1997 — M. R. Mine, « Exploiting proprioception in virtual-environment interaction », Rapport technique, University of North Carolina, Chapel Hill, NC, USA 1997.

Mine et al. 1995 — M. R. Mine, F. P. B. Jr, T. Problem, et F. P. B. Jr., « Moving Objects in Space : Exploiting Proprioception In Virtual-Environment Interaction », Computer (Long. Beach. Calif)., p. 19-26, 1995.

More 1978 — J. J. More, « The Levenberg-Marquardt algorithm: Implementation and theory », Lect. Notes Math., vol. 630, p. 105-116, 1978.

Mossel et al. 2013 — A. Mossel, B. Venditti, et H. Kaufmann, « 3DTouch and HOMER-S : Intuitive Manipulation Techniques for One-Handed Handheld Augmented Reality », Proc. Virtual Real. Int. Conf. Laval Virtual, p. 1-10, 2013.

Mouragnon et al. 2006 — E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, P. Sayd, et P. Cnrs, « Monocular Vision Based SLAM for Mobile Robots », 18th Int. Conf. Pattern Recognit., vol. 3, p. 1027-1031, 2006.

Mulloni et al. 2012 — A. Mulloni, H. Seichter, et D. Schmalstieg, « Indoor navigation with mixed reality world-in-miniature views and sparse localization on mobile devices », Proc. Int. Work. Conf. Adv. Vis. Interfaces - AVI ’12, p. 212, 2012.

Murakami & Taguchi 1991 — K. Murakami et H. Taguchi, « Gesture Recognition using Recurrent Neural Networks », in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems Reaching Through Technology CHI 91, 1991, p. 237-242.

Mutka et al. 2008 — A. Mutka, D. Miklic, et I. Draganjac, « A low cost vision based localization system using fiducial markers », p. 9528-9533, 2008.

Naimark & Foxlin 2002 — L. Naimark et E. Foxlin, « Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker », Proc. - Int. Symp. Mix. Augment. Reality, ISMAR 2002, no Ismar, p. 27-36, 2002.

Newcombe et al. 2011 — R. A. Newcombe, S. J. Lovegrove, et A. J. Davison, « DTAM: Dense tracking and mapping in real-time », in Proceedings of the IEEE International Conference on Computer Vision, 2011, p. 2320-2327.

Nguyen & Huynh 2013 — Nguyen, Trong-Nguyen, Huu-Hung Huynh, and Jean Meunier. "Static hand gesture recognition using artificial neural network." Journal of Image and Graphics 1, no. 1 (2013): 34-38.

Nguyen et al. 2016 — D.-D. Nguyen, A. El Ouardi, E. Aldea, et S. Bouaziz, « HOOFR: An Enhanced Bio-Inspired Feature Extractor », in 23rd International Conference on Pattern Recognition ICPR, 2016.

Nister & Bergen 2004 — D. Nister et J. Bergen, « Visual Odometry », Proc. 2004 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, 2004. CVPR 2004., vol. 1, p. I-652-I-659 Vol.1, 2004.

Noguchi & Yanai 2012 — A. Noguchi et K. Yanai, « A SURF-based spatio-temporal feature for feature-fusion-based action recognition », in Lecture Notes in Computer Science (including subseries Lecture

Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2012, vol. 6553 LNCS, no PART 1, p. 153-167.

Noh et al. 2016 — H. Noh, A. Araujo, J. Sim, et B. Han, « Image Retrieval with Deep Local Features and Attention-based Keypoints », arXiv: 1612.06321, déc. 2016.

O'hara et al. 2013 — K. O'hara, R. Harper, H. Mentis, A. Sellen, et A. Taylor, « On the naturalness of touchless », ACM Trans. Comput. Interact., vol. 20, no 1, p. 1-25, mars 2013.

Oberkampf et al. 1996 — D. Oberkampf, D. F. DeMenthon, et L. S. Davis, « Iterative Pose Estimation Using Coplanar Feature Points », Comput. Vis. Image Underst., vol. 63, no 3, p. 495-511, mai 1996.

Oberweger et al. 2015 — M. Oberweger, P. Wohlhart, et V. Lepetit, « Hands Deep in Deep Learning for Hand Pose Estimation », Proc. 20th Comput. Vis. Winter Work., p. 21-30, 2015.

Oda & Feiner 2012 — O. Oda et S. Feiner, « 3D Referencing Techniques for Physical Objects in Shared Augmented Reality », ISMAR 2012 - 11th IEEE Int. Symp. Mix. Augment. Real. 2012, Sci. Technol. Pap., p. 207-215, nov. 2012.

Olsson et al. 2009 — C. Olsson, F. Kahl, et M. Oskarsson, « Branch-and-bound methods for euclidean registration problems », IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no 5, p. 783-794, mai 2009.

Olwal & Feiner 2003 — A. Olwal et S. Feiner, « The flexible pointer: An interaction technique for selection in augmented and virtual reality », in Uist'03, 2003, vol. 3, p. 81-82.

Oszust 2016 — M. Oszust, « BDSB: Binary descriptor with shared pixel blocks », J. Vis. Commun. Image Represent., vol. 41, p. 154-165, 2016.

Otmane 2010 — S. Otmane, « Modèles et techniques logicielles pour l'assistance à l'interaction et à la collaboration en réalité mixte », Thèse HDR, Université d'Evry-Val d'Essonne, 2010.

Ouramdane et al. 2009 — N. Ouramdane, S. Otmane, et M. Mallem, « Interaction 3D en réalité virtuelle. Etat de l'art », Techniques et sciences informatiques, vol. 28, no 8. p. 1017-1049, 2009.

Papakostas et al. 2013 — G. a. Papakostas, D. E. Koulouriotis, E. G. Karakasis, et V. D. Tourassis, « Moment-based local binary patterns: A novel descriptor for invariant pattern recognition applications », Neurocomputing, vol. 99, p. 358-371, 2013.

Parker et al. 2016 — C. Parker, M. Daiter, K. Omar, G. Levi, et T. Hassner, « The CUDA LATCH Binary Descriptor: Because Sometimes Faster Means Better », in European Conference of Computer Vision ECCV, 2016, p. 685-697.

Petit et al. 2013 — M. A. Petit, M. E. Marchand, et M. K. Kanani, « Détection et suivi basé modèle pour des applications spatiales 1 Introduction » *Congrès francophone des jeunes chercheurs en vision par ordinateur, ORASIS'13*, Jun 2013, Cluny, France. pp.1-6, 2013.

Pierce et al. 1997 — J. S. Pierce, A. S. Forsberg, M. J. Conway, S. Hong, R. C. Zeleznik, et M. R. Mine, « Image plane interaction techniques in 3D immersive environments », In Proceeding Proceedings of the 1997 symposium on Interactive 3D graphics, I3D 1997, Pages 39

Pierce et al. 1999 — J. S. Pierce, B. C. Stearns, et R. Pausch, « Voodoo dolls: seamless interaction at multiple scales in virtual environments », in Proceedings of the 1999 symposium on Interactive 3D graphics - SI3D '99, 1999, p. 141-145.

Potter et al. 2013 — L. E. Potter, J. Araullo, and L. Carter. "The leap motion controller: a view on sign language." In Proceedings of the 25th Australian computer-human interaction conference: augmentation, application, innovation, collaboration, pp. 175-178. ACM, 2013.

Poupyrev & Billinghurst 1996 — I. Poupyrev et M. Billinghurst, « The go-go interaction technique: non-linear mapping for direct manipulation in VR », In Proceedings of the 9th annual ACM symposium on User interface software and technology UIST 1996, Pages 79-80.

Poupyrev & Ichikawa 1998 — I. Poupyrev, T. Ichikawa, S. Weghorst, et M. Billinghurst, « Egocentric Object Manipulation in Virtual Environments: Empirical Evaluation of Interaction Techniques », Comput. Graph. Forum, vol. 17, no 3, p. 41-52, août 1998.

Poupyrev & Ichikawa 1999 — I. Poupyrev et T. Ichikawa, « Manipulating Objects in Virtual Worlds: Categorization and Empirical Evaluation of Interaction Techniques », J. Vis. Lang. Comput., vol. 10, no 1, p. 19-35, 1999.

Priyal & Bora 2013 — S. P. Priyal et P. K. Bora, « A robust static hand gesture recognition system using geometry based normalizations and Krawtchouk moments », Pattern Recognit., vol. 46, no 8, p. 2202-2219, 2013.

Qian et al. 2014 — C. Qian, X. Sun, Y. Wei, X. Tang, et J. Sun, « Realtime and robust hand tracking from depth », in Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, 2014, p. 1106-1113.

Quan & Lan 1999 — L. Quan et Z. Lan, « Linear N-point caméra pose determination », IEEE Trans. Pattern Anal. Mach. Intell., vol. 21, no 8, p. 774-780, 1999.

Radkowski & Stritzke 2012 — R. Radkowski and C. Stritzke. "Interactive hand gesture-based assembly for augmented reality applications." In ACHI 2012, The Fifth International Conference on Advances in Computer-Human Interactions, pp. 303-308. 2012.

Raginsky & Lazebnik 2009 — M. Raginsky et S. Lazebnik, « Locality-sensitive Bbinary Codes from Shift-invariant Kernels », Conf. Neural Inf. Process. Syst., p. 1509-1519, 2009.

Rautaray & Agrawal 2012 — S. S. Rautaray et A. Agrawal, « Vision based hand gesture recognition for human computer interaction: a survey », Artif. Intell. Rev., vol. 43, no 1, 2012.

Reale et al. 2011 — M. J. Reale, S. Canavan, L. Yin, K. Hu, and T. Hung. "A multi-gesture interaction system using a 3-D iris disk model for gaze estimation and an active appearance model for 3-D hand pointing." IEEE Transactions on Multimedia 13, no. 3 (2011): 474-486.

Rice et al. 2006 — A. C. Rice, A. R. Beresford, et R. K. Harle, « Cantag: An open source software toolkit for designing and deploying marker-based vision systems », Proc. - Fourth Annu. IEEE Int. Conf. Pervasive Comput. Commun. PerCom 2006, vol. 2006, p. 12-21, 2006.

Rosin 1999 — P. L. Rosin, « Measuring Corner Properties », Comput. Vis. Image Underst., vol. 73, no 2, p. 291-307, févr. 1999.

Rosten et al. 2006 — E. Rosten et T. Drummond, « Machine learning for high-speed corner detection », in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2006, vol. 3951 LNCS, p. 430-443.

Rosten et al. 2010 — E. Rosten, R. Porter, et T. Drummond, « Faster and better: A machine learning approach to corner detection », IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no 1, p. 105-119, 2010.

Rublee et al. 2011 — E. Rublee, V. Rabaud, K. Konolige, et G. Bradski, « ORB: an efficient alternative to SIFT or SURF », in International Conference on Computer Vision ICCV, 2011, p. 2564-2571.

Saha & Démoulin 2012 — S. Saha et V. Démoulin, « ALOHA: An efficient binary descriptor based on Haar features », in Proceedings - International Conference on Image Processing, ICIP, 2012, p. 2345-2348.

Sandor et al. 2015 — C. Sandor et al., « Breaking the Barriers to True Augmented Reality », arXiv: 1512.05471, December, p. 1-13, 2015.

Schmid et al. 2000 — C. Schmid, R. Mohr, et C. Bauckhage, « Evaluation of interest point detectors », Int. J. Comput. Vis., vol. 37, no 2, p. 151-172, 2000.

Shi & Tomasi 1994 — Jianbo Shi et C. Tomasi, « Good features to track », in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition CVPR-94, 1994, p. 593-600.

Shui & Zhang 2013 — P.-L. Shui et W.-C. Zhang, « Corner detection and classification using anisotropic directional derivative representations. », IEEE Trans. Image Process., vol. 22, no 8, p. 3204-18, août 2013.

Sibley et al. 2010 — G. Sibley, C. Mei, I. Reid, et P. Newman, « Vast-scale Outdoor Navigation Using Adaptive Relative Bundle Adjustment », Int. J. Rob. Res., vol. 29, no 8, p. 958-980, juill. 2010.

Simonyan et al. 2014 — K. Simonyan, A. Vedaldi, et A. Zisserman, « Learning local feature descriptors using convex optimisation », IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no 8, p. 1573-1585, août 2014.

Simo-Serra et al. 2015 — E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, et F. Moreno-Noguer, « Discriminative Learning of Deep Convolutional Feature Point Descriptors », in Proceedings of the IEEE International Conference on Computer Vision, 2015, p. 118-126.

Smith & Brady 1997 — S. Smith et J. Brady, « SUSAN—a new approach to low level image processing », Int. J. Comput. Vis., vol. 23, no 1, p. 45-78, 1997.

Stanimirovic et al. 2014 — D. Stanimirovic, N. Damasky, S. Webel, D. Koriath, A. Spillner, et D. Kurz, « A Mobile Augmented Reality System to Assist Auto Mechanics », in ISMAR 2014 - IEEE International Symposium on Mixed and Augmented Reality - Science and Technology 2014, Proceedings, 2014, no September, p. 305-306.

Stergiopoulou & Papamarkos 2009 — E. Stergiopoulou et N. Papamarkos, « Hand gesture recognition using a neural network shape fitting technique », Eng. Appl. Artif. Intell., vol. 22, no 8, p. 1141-1158, 2009.

Sternberger et al. 2006 — L. Sternberger, « Interaction en réalité virtuelle », Thèse de doctorat, Université Louis Pasteur (Strasbourg), 2006.

Stoakley et al. 1997 — R. Stoakley, M. J. Conway, et R. Pausch, « Virtual reality on a WIM », Proc. SIGCHI Conf. Hum. factors Comput. Syst. - CHI '95, p. 265-272, 1995.

Strecha et al. 2012 — C. Strecha, A. M. Bronstein, M. M. Bronstein, et P. Fua, « LDAHash: Improved matching with smaller descriptors », IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no 1, p. 66-78, 2012.

Supancic et al. 2015 — J. S. Supancic III, G. Rogez, Y. Yang, J. Shotton, et D. Ramanan, « Depth-based hand pose estimation: methods, data, and challenges », arXiv1504.06378, p. 1868-1876, 2015.

Sutherland 1968 — I. E. Sutherland, « A head-mounted three dimensional display », in Proceedings of the AFIPS '68 (Fall, part I), 1968, p. 757-764.

Ta et al. 2009 — D. N. Ta, W. C. Chen, N. Gelfand, et K. Pulli, « SURFTrac: Efficient tracking and continuous object recognition using local feature descriptors », in 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009, 2009, p. 2937-2944.

Tang et al. 2015 — A. Tang, K. Lu, Y. Wang, J. Huang, et H. Li, « A Real-Time Hand Posture Recognition System Using Deep Neural Networks », ACM Trans. Intell. Syst. Technol., vol. 6, no 2, p. 21:1--21:23, 2015.

Tanriverdi & Jacob 2000 — V. Tanriverdi et R. J. K. Jacob, « Interacting with eye movements in virtual environments », Proc. SIGCHI Conf. Hum. factors Comput. Syst. - CHI '00, vol. 2, no 1, p. 265-272, 2000.

Taylor et al. 2016 — J. Taylor et al., « Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences », ACM Trans. Graph., vol. 35, no 4, p. 1-12, juill. 2016.

Tola et al. 2010 — E. Tola, V. Lepetit, P. Fua, et S. Member, « DAISY: An efficient dense descriptor applied to wide-baseline stereo », IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no 5, p. 815-830, mai 2010.

Tompson et al. 2014 — J. Tompson, K. Perlin, Y. LeCun, et M. Stein, « Real-Time Continuous Pose Recovery of Human Hands Using Convolutional Networks », ACM Trans. Graph., vol. 33, no 5, p. 110-169, 2014.

Trzcinski & Lepetit 2012 — T. Trzcinski et V. Lepetit, « Efficient discriminative projections for compact binary descriptors », in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2012, vol. 7572 LNCS, no PART 1, p. 228-242.

Trzcinski et al. 2013 — T. Trzcinski, M. Christoudias, P. Fua, et V. Lepetit, « Boosting binary keypoint descriptors », Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., p. 2874-2881, 2013.

Valenzuela et al. 2014 — R. E. G. Valenzuela, W. R. Schwartz, et H. Pedrini, « Linear dimensionality reduction applied to scale invariant feature transformation and speeded up robust feature descriptors », J. Electron. Imaging, vol. 23, no 3, p. 33017, juin 2014.

Várkonyi-Kóczy & Tusor 2011 — A. R. Várkonyi-Kóczy, and B. Tusor. "Human–computer interaction for smart environment applications using fuzzy hand posture and gesture models." IEEE Transactions on Instrumentation and Measurement 60, no. 5 (2011): 1505-1514.

Ventura & Höllerer 2012 — J. Ventura et T. Höllerer, « Wide-area scene mapping for mobile visual tracking », ISMAR 2012 - 11th IEEE Int. Symp. Mix. Augment. Real. 2012, Sci. Technol. Pap., no November, p. 3-12, 2012.

Wagner & Schmalstieg 2007 — D. Wagner et D. Schmalstieg, « ARToolKitPlus for Pose tracking on Mobile Devices », Proc. 12th Comput. Vis. Winter Work. CVWW07, p. 139–146, 2007.

Wagner et al. 2008 — D. Wagner, G. Reitmayr, A. Mulloni, T. Drummond, et D. Schmalstieg, « Pose tracking from natural features on mobile phones », in Proceedings - 7th IEEE International Symposium on Mixed and Augmented Reality 2008, ISMAR 2008, 2008, p. 125-134.

Wang & Popović 2009 — R. Y. Wang et J. Popović, « Real-time hand- tracking with a color glove », ACM Trans. Graph., vol. 28, no 3, p. 1, 2009.

Wang & Ye 2000 — Y. Wang et A. Ye, « Maxicode data extraction using spatial domain features », US Pat. 6,053,407, 2000.

Wang et al. 2011 — R. Y. Wang, S. Paris, et J. Popovi, « 6D Hands: Markerless Hand tracking for Computer Aided Design », ACM User Interface Softw. Technol., 2011.

Wang et al. 2016 — S. Wang, J. Song, J. Lien, I. Poupyrev, et O. Hilliges, « Interacting with Soli: Exploring Fine-Grained Dynamic Gesture Recognition in the Radio-Frequency Spectrum », in Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16, 2016, p. 851-860.

Wigdor & Wixon 2011 — D. Wigdor et D. Wixon, « Brave NUI World: Designing Natural User Interfaces for Touch and Gesture », Brave NUI World Des. Nat. User Interfaces Touch Gesture, p. 253, 2011.

Willis & Sui 2009 — A. Willis et Y. Sui, « An algebraic model for fast corner detection », in Proceedings of the IEEE International Conference on Computer Vision, 2009, p. 2296-2302.

Winder & Brown 2007 — S. Winder et M. Brown, « Learning local image descriptors », in 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2007. pp 1-8.

Wu et al. 2013 — J. Wu, Z. Cui, V. S. Sheng, P. Zhao, D. Su, et S. Gong, « A Comparative Study of SIFT and its Variants », Meas. Sci. Rev., vol. 13, no 3, p. 122-131, 2013.

Wu et al. 2016 — D. Wu et al., « Deep Dynamic Neural Networks for Multimodal Gesture Segmentation and Recognition », IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no 8, p. 1583-1597, 2016.

Xia et al. 2015 — Yan Xia, K. He, P. Kohli, et J. Sun, « Sparse projections for high-dimensional binary codes », 2015 IEEE Conf. Comput. Vis. Pattern Recognit., p. 3332-3339, 2015.

Xu et al. 2014 — X. Xu, L. Tian, J. Feng, et J. Zhou, « OSRI: A rotationally invariant binary descriptor », IEEE Trans. Image Process., vol. 23, no 7, p. 2983-2995, 2014.

Yang & Cheng 2014 — X. Yang et K. T. T. Cheng, « Local difference binary for ultrafast and distinctive feature description », IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no 1, p. 188-194, 2014.

Yao & Li 2013 — Y. Yao et C. T. Li, « Real-time hand gesture recognition for uncontrolled environments using adaptive SURF tracking and hidden conditional random fields », in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2013, vol. 8034 LNCS, no PART 2, p. 542-551.

Yi et al. 2016 — K. M. Yi, E. Trulls, V. Lepetit, et P. Fua, « LIFT: Learned invariant feature transform », in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016, vol. 9910 LNCS, p. 467-483.

Yun et al. 2012 — L. Yun, Z. Lifeng, et Z. Shujun, « A Hand Gesture Recognition Method Based on Multi-Feature Fusion and Template Matching », Procedia Eng., vol. 29, p. 1678-1684, 2012.

Zenati et al. 2013 — N. Zenati, S. Benbelkacem, M. Belhocine, et A. Bellarbi, « A new AR interaction for collaborative E-maintenance system », in IFAC Proceedings Volumes (IFAC-PapersOnline), 2013, vol. 46, no 9, p. 619-624.

Zenati et al. 2015 — N. Zenati, M. Hamidia, A. Bellarbi, et S. Benbelkacem, « E-maintenance for photovoltaic power system in Algeria », in 2015 IEEE International Conference on Industrial Technology (ICIT), 2015, p. 2594-2599.

Zenati-Henda et al. 2014 — N. Zenati-Henda, A. Bellarbi, S. Benbelkacem, et M. Belhocine, « Augmented reality system based on hand gestures for remote maintenance », in International Conference on Multimedia Computing and Systems -Proceedings, 2014, p. 5-8.

Zendjebil 2010 — I. Zendjebil, « Localisation 3D basée sur une approche de suppléance multi-capteurs pour la Réalité Augmentée Mobile en Milieu Extérieur », Thèse de doctorat, Université d'Evry-Val d'Essonne, 2010.

Zeng et al. 2016 — Z. Zeng, H. Liang, M. Su, et C. Zeng, « Performance Evaluation of Binary Descriptors for Mobile Robots », 2016 IEEE 11th Conf. Ind. Electron. Appl., no 51575412, p. 568-573, juin 2016.

Zhai et al. 1994 — S. Zhai, W. Buxton, et P. Milgram, « Investigating The "Silk Cursor": Transparency for 3D Target Acquisition », Hum. Factors Comput. Syst., vol. 1, p. 459-465, 1994.

Zhang 2000 — Z. Zhang, « A flexible new technique for caméra calibration », IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no 11, p. 1330-1334, 2000.

Zhang et al. 2010 — X. Zhang, H. Wang, A. W. B. Smith, X. Ling, B. C. Lovell, et D. Yang, « Corner detection based on gradient correlation matrices of planar curves », Pattern Recognit., vol. 43, no 4, p. 1207-1223, 2010.

Zhang et al. 2014 — S. Zhang, Q. Tian, Q. Huang, W. Gao, et Y. Rui, « USB: Ultrashort binary descriptor for fast visual matching and retrieval », IEEE Trans. Image Process., vol. 23, no 8, p. 3671-3683, 2014.

Zhao et al. 2012 — X. Zhao, Z. Song, J. Guo, Y. Zhao, et F. Zheng, « Real-time hand gesture detection and recognition by random forest », Commun. Comput. Inf. Sci., vol. 289 CCIS, no PART 2, p. 747-755, 2012.

Zheng et al. 2009 — Y. T. Zheng et al., « Tour the World: Building a web-scale landmark recognition engine », in 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009, 2009, p. 1085-1092.

Zheng et al. 2013 — Y. Zheng, Y. Kuang, S. Sugimoto, K. Astrom, et M. Okutomi, « Revisiting the PnP problem: A fast, general and optimal solution », in Proceedings of the IEEE International Conference on Computer Vision, 2013, p. 2344-2351.

Zhou et al. 2008 — F. Zhou, H. B. L. Dun, et M. Billinghurst, « Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR », in Proceedings - 7th IEEE International Symposium on Mixed and Augmented Reality 2008, ISMAR 2008, 2008, p. 193-202.

Zhou et al. 2015 — W. Zhou, H. Li, R. Hong, Y. Lu, et Q. Tian, « BSIFT: Toward Data-Independent Codebook for Large Scale Image Search », IEEE Trans. Image Process., vol. 24, no 3, p. 967-979, 2015.

Annexe

Cette annexe comporte deux parties. La première partie présente les deux questionnaires de l'évaluation utilisés dans le chapitre quatre. La deuxième partie de l'annexe résume mes activités de recherches et de développement au sein du CDTA, qui ne rentrent pas dans le cadre de la thèse. Néanmoins, ces travaux concernent le domaine de la réalité augmentée et de l'interaction 3D.

Annexe A

A.1. Questionnaire d’évaluation USE

1. Utilité

- Cela m'aide à être plus efficace.
- Cela m'aide à être plus productif.
- C'est utile.
- Elle rend les choses que je veux accomplir plus facile à faire.
- Cela m'économise du temps quand je l'utilise.
- Elle répond à mes besoins.
- Elle fait tout ce que je m'attends à ce qu'il fasse.

2. Facilité de l’utilisation

- C'est facile à utiliser.
- Elle est simple à utiliser.
- Elle est conviviale.
- Elle exige le moins d'étapes possible pour accomplir ce que je veux faire.
- Elle est flexible.
- Son utilisation nécessite le moindre d’effort.
- Je peux l'utiliser sans instructions écrites.
- Je ne remarque aucune incohérence au moment où je l'utilise.
- Je peux me remettre rapidement des erreurs.
- Je peux l'utiliser avec succès à chaque fois.

3. Facilité de l’apprentissage

- J'ai appris à l'utiliser rapidement.
- Je me souviens facilement comment l'utiliser.
- Elle est facile d'apprendre à l'utiliser.
- Je deviens rapidement habile avec elle.

4. Satisfaction

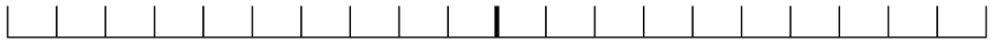
- J'en suis satisfait.
- Je la recommande à un ami.
- C'est amusant à utiliser.
- Elle fonctionne de la façon dont je veux qu’elle fonctionne.
- C'est merveilleux.
- Je sens que je dois l'avoir.
- Elle est agréable à utiliser.

A.2. Questionnaire NASA-TLX

Exigence Mentale La tâche vous a-t-elle paru simple, nécessitant peu d'attention (faible) ou complexe, nécessitant beaucoup d'attention (élevée) ?

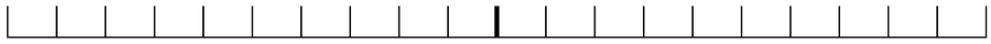
Faible Élevée

Exigence Physique La tâche vous a-t-elle paru facile, peu fatigante, calme (faible) ou pénible, fatigante, active (élevée) ?



Faible Élevée

Exigence Temporelle Quelle a été l'importance de la pression temporelle causée par la rapidité nécessitée pour l'accomplissement de la tâche ? Etait-ce un rythme lent et tranquille (faible) ou rapide et précipité (élevé) ?

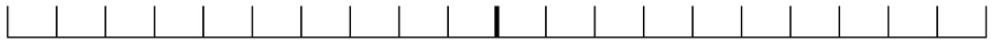


Faible Élevée

Performance Quelle réussite pensez-vous avoir eu dans l'accomplissement de votre tâche ? Comment pensez-vous avoir atteint les objectifs déterminés par la tâche ?

Bonne Mauvaise

Effort Quel degré d'effort avez-vous dû fournir pour exécuter la tâche demandée, (mentalement et physiquement) ?



Faible Élevée

Frustration Pendant l'exécution du travail vous êtes-vous senti satisfait, relaxé, sûr de vous (niveau de frustration faible), ou plutôt découragé, irrité, stressé, sans assurance (niveau de frustration élevé) ?

Faible Élevée

Annexe B

B.1. Technique d'interaction 3D low-cost pour la RA

1. Contexte général

Dans le cadre du projet IM@REV (Interaction 3D Multimodale & Collaborative, dans un environnement de REalité Virtuelle et augmentée) initié au CDTA au sein de l'équipe IRVA, nous avons participé à différents travaux notamment dans le domaine de l'interaction. L'objectif étant d'offrir à l'utilisateur un moyen facile et simple d'utilisation lui permettant d'interagir en 3D avec son environnement virtuel ou augmenté, nous avons proposé une technique d'interaction 3D simple, low-cost que nous avons nommé « 2in1 Marker ».

2. Description de la technique

« 2in1 Marker » (Bellarbi et al. 2014a), est une technique d'interaction 3D low-cost basée sur la métaphore « main virtuelle simple ». L'idée repose sur l'utilisation des marqueurs d'Artoolkit (Hirokazu & Billinghurst 1999). Cette technique permet de sélectionner et de manipuler en 3D des objets virtuels, ainsi que le contrôle d'application en utilisant un stylet composé de deux marqueurs, celui-ci permet à l'utilisateur de déclencher un click lorsqu'il souhaite sélectionner un objet, suite à cette sélection, le déplacement du stylet permettra la manipulation de l'objet. (Figure 1).

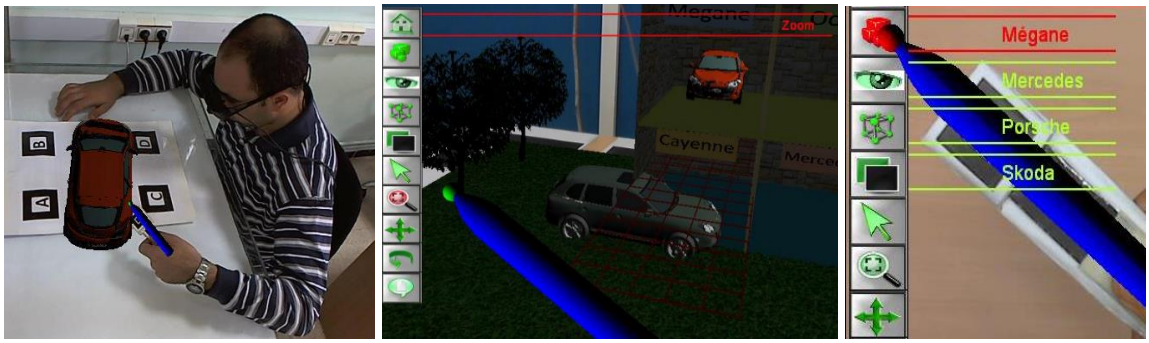


Figure 1. Technique d'interaction «2in1 Marker » (Bellarbi et al. 2014a).

3. Principe de fonctionnement

a. Conception du stylet

Le principe de cette technique repose sur l'utilisation d'un stylet (en papier) constitué de deux marqueurs d'Artoolkit superposé l'un sur l'autre, avec la possibilité de basculer entre les deux par un simple geste du doigt de l'utilisateur (figure 2). Le changement entre les marqueurs

est utilisé pour la sélection et le contrôle de l'application, tout en garantissant le suivi de la main de l'utilisateur. Toutefois, la taille des marqueurs utilisés ainsi que la limite de la librairie Artoolkit, ont une influence majeure sur la qualité de suivi et de contrôle de l'application.

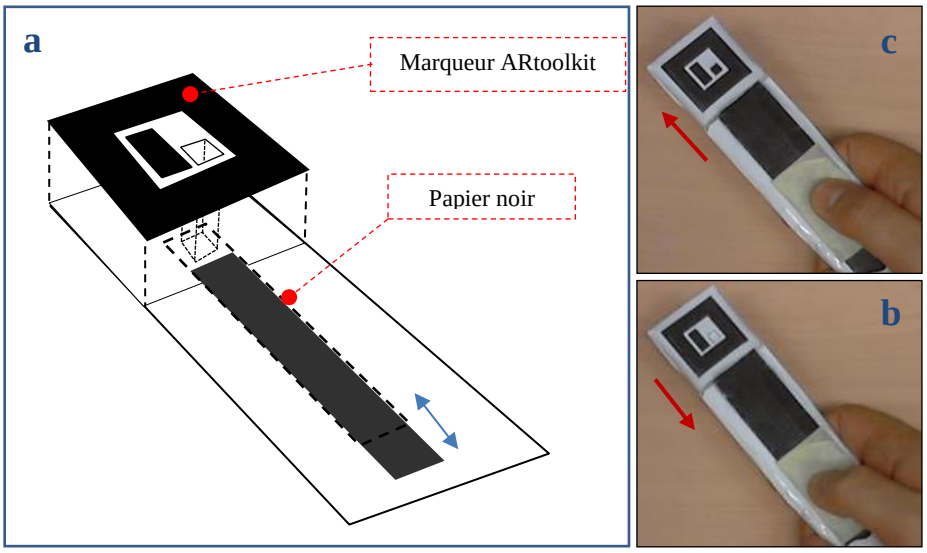


Figure 2. Principe de la technique « 2in1 Marker ». a) Le schéma du stylet avec les deux marqueurs. b et c) Le changement entre les deux marqueurs (changement d'état) par le doigt de l'utilisateur (Bellarbi et al. 2014a).

Ainsi, le changement d'état d'un marqueur à l'autre nous permet de déclencher un évènement. Nous avons donc reproduit les deux fonctions : "MouseDown" et "MouseUp".

b. Stylet Virtuel

Lors du processus d'interaction, l'utilisateur peut se perdre et oublier la dernière action effectuée (l'état du stylet). Afin d'aider l'utilisateur nous avons introduit un stylet virtuel qui s'affiche sur le stylet réel. Celui-ci représente la main virtuelle. Le stylet virtuel est doté d'une tête de forme sphérique qui change de couleur lors de la sélection. Nous avons donc défini deux modes : actif et inactif avec respectivement deux couleurs différentes (rouge et vert), (voir figure 3).

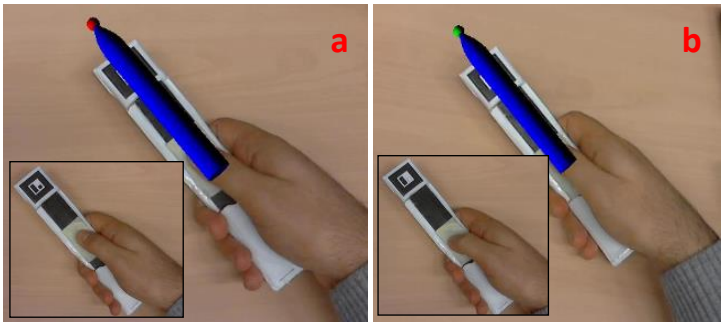


Figure 3. Stylet Virtuel à deux états: a) mode actif, b) mode inactif.

c. Contrôle d'application

L'application est dotée d'un menu virtuel positionné à gauche de l'écran. Le menu est composé de plusieurs boutons qui permettent de sélectionner un objet, le mode d'affichage (plein, fil de faire), ou encore le type de manipulation (rotation, translation, changement d'échelle), (voir figure 4).

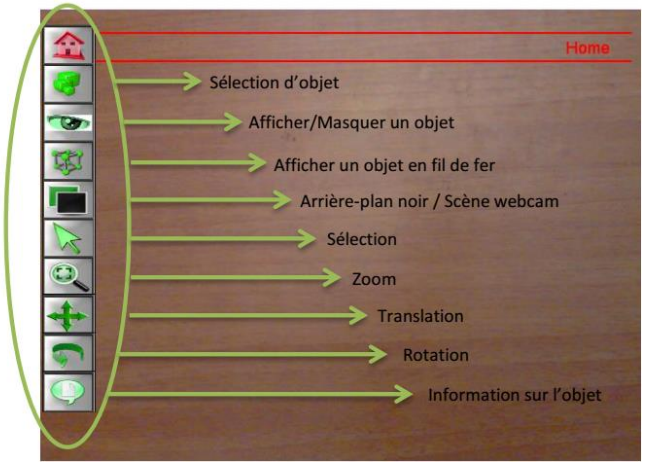


Figure 4. Menu virtuel.

Afin de permettre à l'utilisateur d'interagir avec le menu, nous avons appliqué la technique Ray-Casting. Ainsi pour chaque frame nous calculons la position 2D du stylet sur l'écran en projetant ses coordonnées 3D (équation 1 et 2).

$$x2d = ((x3d/z3d)* \text{ largeur\_ecran}) \quad \ldots\ldots(1)$$

$$y2d = ((y3d/y3d)* \text{ Longeur\_ecran}) \quad \ldots\ldots(2)$$

Pour simplifier la sélection nous avons introduit un guide visuel en attribuant trois couleurs aux boutons : vert au repos, orange lorsque le stylet passe au-dessus du bouton, et rouge lorsque le bouton est sélectionné via le stylet (figure 5).

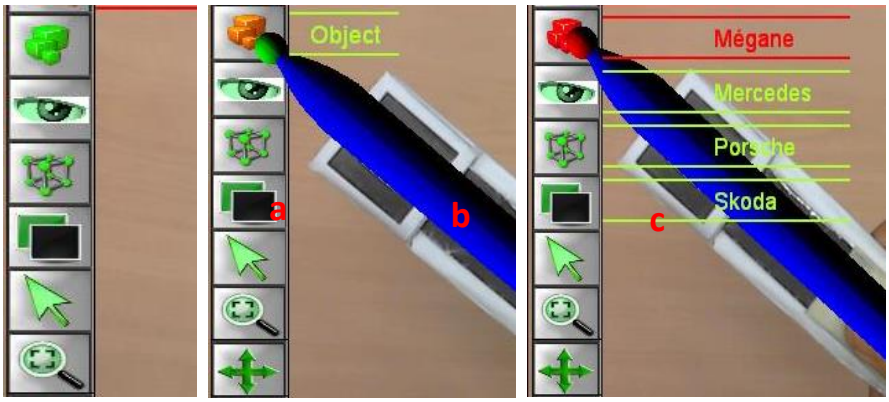


Figure 5. Différents états du menu : a) état normal, b) état entré en zone, c) état sélection.

d. Manipulation de l'objet

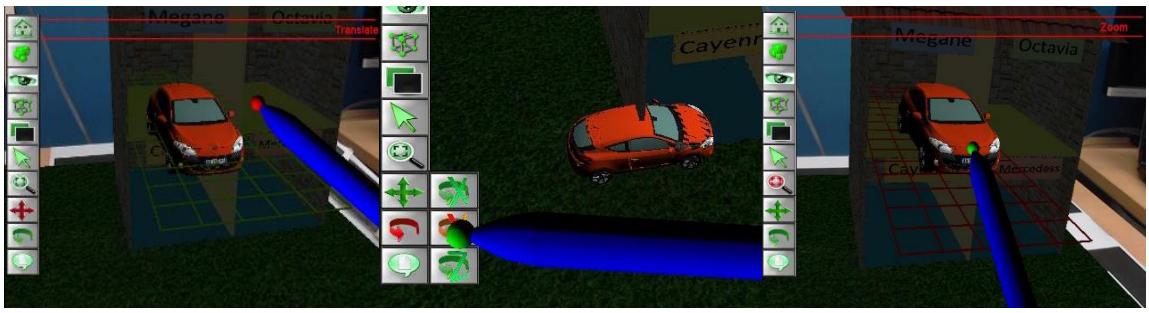
Une fois l'objet sélectionné via le menu, l'utilisateur sélectionne la tâche à exécuter : translation, rotation ou encore changement d'échelle, dans le cas de la rotation il doit sélectionner également l'axe de la rotation souhaité.

En ce qui concerne la translation, la manipulation se fait avec la métaphore de la main virtuel simple. Ainsi pour chaque frame nous calculons la position actuelle du stylet puis la distance avec la position précédente, cette distance est ensuite ajoutée la position de l'objet sélectionné.

$$Objet\_3D_i\ (x,y,z) = Objet\_3D_{i-1}\ (x,y,z) + (Stylet\_3D_i\ (x,y,z) - Stylet\_3D_{i-1}\ (x,y,z) \quad \ldots\ldots\ldots\ldots\ldots\ldots (3)$$

Dans le cas d'une rotation, nous projetons la position 3D du stylet sur l'écran puis nous calculons la distance avec la position précédente projetée. Pour calculer l'angle de rotation, nous considérons que la taille de l'écran correspond à 360° puis nous calculons l'angle à appliquer via la règle de trois.

Avec le même principe que pour la rotation, l'utilisateur peut effectuer un changement d'échelle pour l'objet sélectionné. La figure (figure 6) ci-dessous représente les différentes tâches de manipulation.



a) Translation b) Rotation c) Changement d'échelle

Figure 6. Manipulation d'un objet 3D.

4. Evaluation Préliminaire

Afin d'évaluer notre technique, nous avons créé un environnement 3D qui est constitué d'un garage à quatre places pour quatre voitures différentes. Chaque voiture à une place qui lui est dédiée.

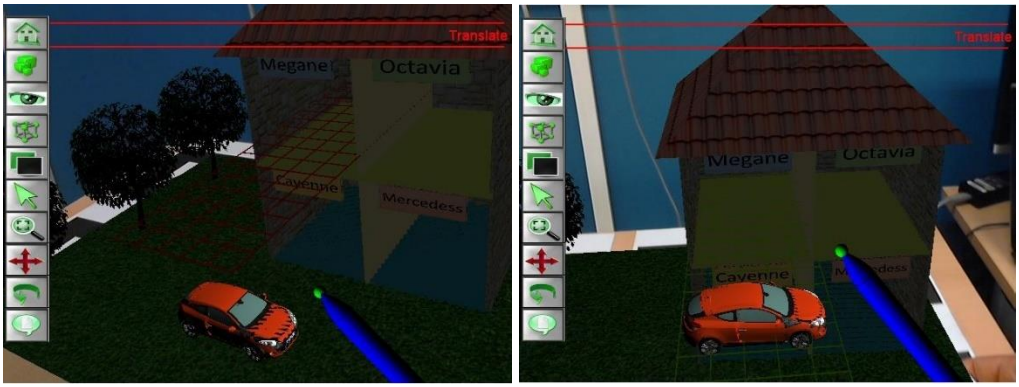
Nous avons fait appel à un ensemble de (12) participants avec une tranche d'âge qui varie de 25 à 45 ans. Chacun des participants a testé la technique d'interaction selon (03) scénarios différents et sur (03) dispositifs d'affichage différents : casque optique, casque vidéo, tablette. Ces

tests nous permettent d'une part d'évaluer le temps d'exécution et la précision et d'autre part de voir l'influence que pourrait avoir le dispositif d'affichage sur notre technique d'interaction.

Afin d'aider l'utilisateur, nous avons introduit un guide visuel : une grille rouge à l'endroit où la voiture doit être placée. L'utilisateur utilise alors le stylet pour déplacer la voiture, une fois quelle est dans la bonne position la grille change de couleur et devient verte.

- Premier Sénario :

Le premier scénario, représente une tâche simple qui est de positionner l'objets 3D via une simple translation. Ainsi, pour le premier véhicule à déplacer, l'utilisateur pourra le mettre dans le bon emplacement via une simple translation (figure 7).

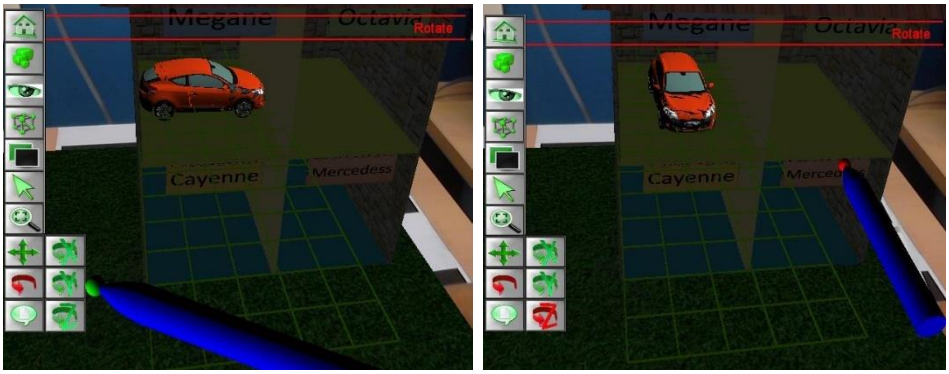


Cible Destination

Figure 7. Positionnement de l'objet 3D selon le scénario 1.

- Deuxième scénario : Tâche complexe : rotation + translation

Pour le second scénario, l'utilisateur doit effectuer une rotation et une translation pour pouvoir mettre l'objet 3D au bon endroit (figure 8).



Cible Destination

Figure 8. Rotation et translation de l'objet 3D.

- Troisième scénario : Tâche complexe : Changement d'échelle + translation

Pour le troisième scénario, l'utilisateur doit d'abord effectuer un changement d'échelle pour diminuer la taille du véhicule puis une translation pour pouvoir mettre l'objet 3D au bon endroit (figure 9).

Cible Destination

Figure 9. Changement d'échelle et translation de l'objet 3D.

A la fin des tests, les utilisateurs doivent remplir un questionnaire (voir Tableau 1)

Tableau 1. Questionnaire

Q1	**Comment trouvez-vous la manipulation 3D en termes de rapidité?**
Q2	Pensez-vous que les manipulations 3D sont précises?
Q3	Comment évalueriez-vous la technique d'interaction 3D du point de vue facilité d'utilisation?
Q4	Combien jugez-vous l'intuitivité de l'interaction 3D en termes de translation 3D, de rotation 3D et de mise à l'échelle?
Q5	Pensez-vous que la taille d'affichage est suffisante?
Q6	Comment trouvez-vous la quantité d'informations affichées à l'écran?
Q7	Le système est-il confortable pour effectuer les tâches d'interaction?
Q8	Comment évaluez-vous l'utilité de la technique d'interaction dans un environnement de réalité mixte?

Une fois que tous les utilisateurs ont effectué leurs tests et ont rempli les questionnaires, nous avons fait une évaluation quantitative et une évaluation subjective à partir de ces données récoltées.

a. Evaluation quantitative

Nous avons considéré deux aspects : le premier concerne l'évaluation de la technique à partir des données récoltées des sujets. Le second aspect consiste en l'évaluation de la technique en fonction de la tâche exécutée. Ainsi, nous avons évalué deux critères : le temps d'accomplissement d'une tâche et le nombre d'étapes nécessaire pour accomplir la tâche.

La figure 10 représente la moyenne du temps de l'accomplissement d'une tâche pour chacun des dispositifs d'affichage. Ainsi, le temps d'accomplissement d'une tâche est plus important avec une tablette qu'avec un dispositif main libre (casque optique, casque vidéo). Aussi, selon les participants, il est plus difficile d'exécuter une tâche en ayant une main occupée avec la tablette.

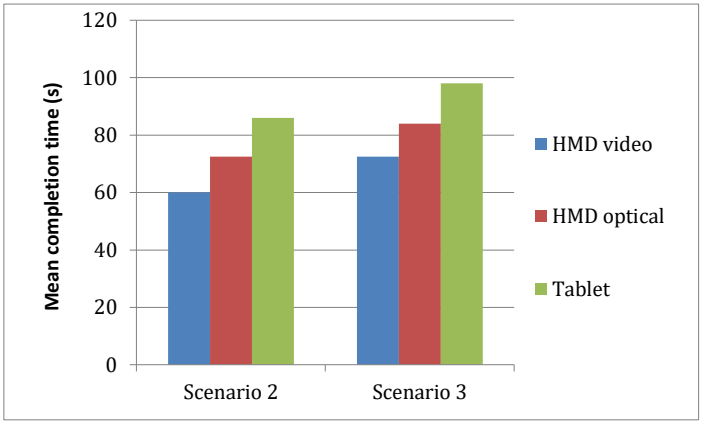


Figure 10. Moyenne du temps d'accomplissement d'une tâche.

Les résultats de l'évaluation du nombre de manipulation ont montré que les utilisateurs sont capables d'effectuer des tâches d'interaction avec un nombre de manipulation sensiblement moins important en utilisant le casque vidéo pour le scénario 3 comme illustré sur la figure 11. On a observé ici, en particulier, que la différence réside dans les tâches de positionnement et de mise à l'échelle. Pour le casque vidéo, ce scénario est réalisé en 5 étapes, contrairement au casque optique et à la tablette où le nombre d'étapes est respectivement de 6 et 7.

Nous avons également constaté une différence significative entre le scénario 1 et le scénario 2, entre les deux casques HMD (vidéo et optique) et la tablette pour l'exécution des tâches correspondantes (voir figure 11). Dans la même situation, les casques présentent la même performance vue que pour les deux dispositifs HMD, le nombre d'étapes pour le scénario 1 est égal à 3 et le nombre d'étapes pour le scénario 2 est égal à 4. La figure 11 montre qu'il n'y a pas de différence pour le positionnement à l'aide d'un guide visuel et pour les tâches de positionnement et de rotation pour les dispositifs portables.

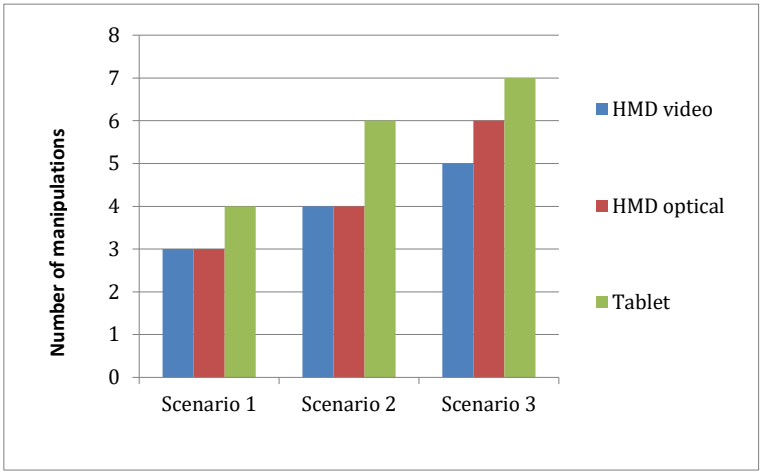


Figure 11. Moyenne du nombre de manipulation par scénario.

b. Evaluation Subjective

Cette étude est basée sur les données recueillies à partir des résultats du questionnaire. Les participants ont rempli le questionnaire après les tests : celui-ci est composé de huit (8) questions. Les réponses de chaque question varient sur une échelle de 1 à 5 (5 : fortement d'accord, 4 : d'accord, 3 : moyenne 2 : légèrement en désaccord, 1 : complètement en désaccord). Un champ "description" est mis à la disposition des participants pour donner leurs commentaires après les tests. Les mêmes dispositifs ont été utilisés au cours des essais. Selon le questionnaire, en répondant aux questions Q1, Q2 et Q3, les sujets sont capables de réaliser les tâches canoniques d'interaction 3D (positionnement, rotation, sélection, mise à l'échelle) dans un environnement de réalité mixte, comme le montre la figure 12.

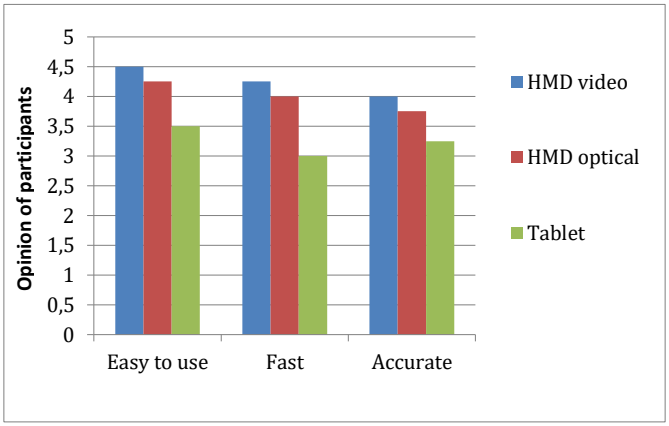


Figure 12. Note moyenne attribuée par les utilisateurs.

L'analyse de la facilité d'utilisation, de la rapidité et de la précision, regroupées par l'expérience de l'utilisateur, a révélé des notes significativement meilleures pour les dispositifs HMD comparativement à la tablette. Les participants ont trouvé plus de difficultés à manipuler et interagir en utilisant la tablette.

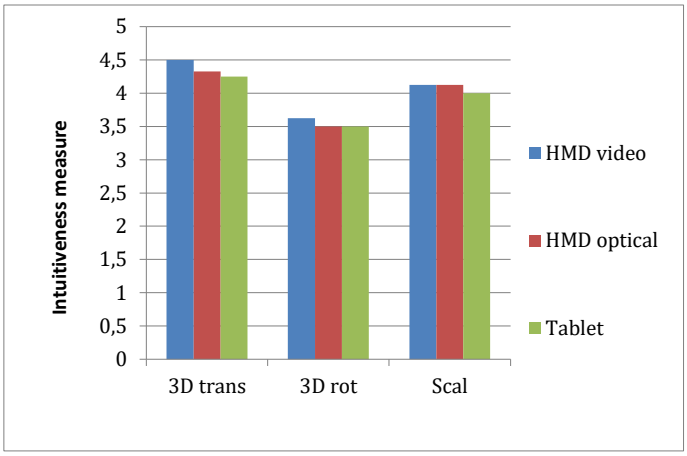


Figure 13. L'intuitivité des utilisateurs.

Pour la question Q4, le classement de l'intuitivité des manipulations canoniques 3D en termes de translation 3D, de rotation 3D et de mise à l'échelle pour le premier, le deuxième et le troisième dispositif est montré dans la Fig. 13. En analysant l'intuitivité de chaque manipulation, groupée par l'expérience des utilisateurs, nous avons observé que le dispositif n'influence pas l'intuitivité pour la translation 3D, la rotation 3D et la mise à l'échelle.

Les résultats montrent qu'il n'y a pas une différence significative concernant le temps global d'accomplissement de la tâche, en particulier pour le scénario 1 lors de l'utilisation de dispositifs main libre (casque vidéo et optique). Par contre, la différence a été observée pour la tablette qui a présenté moins de performances en utilisant la technique d'interaction 3D. La situation similaire a été observée en ce qui concerne la facilité d'utilisation, et la précision de la technique d'interaction 3D. Dans notre cas, nous avons observé que l'intuitivité est le seul aspect non influencé par le type de dispositifs utilisés.

B.2. Le projet « Remote Gestures »

5. Contexte général

Dans le cadre du projet IM@REV (Interaction 3D Multimodale & Collaborative, dans un environnement de REalité Virtuelle et augmentée) initié au CDTA au sein de l'équipe IRVA, nous avons participé à travaux dans le domaine de la reconnaissance de geste. L'objectif est d'offrir un moyen d'interaction naturel qui simplifie l'exécution des tâches.

6. Description du projet

Dans diverses industries, des technologies complexes sont mises en place pour améliorer la productivité. Parce que les tâches d'entretien peuvent être très complexes, les techniciens peuvent être assistés par des experts à distance pour effectuer leurs tâches physiques. L'assistance à la maintenance peut être prise en charge par la réalité augmentée, une technologie puissante qui lie directement les instructions sur la façon d'effectuer des tâches de maintenance.

Nous avons donc réalisé un système de collaboration en temps réel pour l'assistance à distance où des gestes de la main sont utilisés pour guider un travailleur distant à effectuer des tâches manuelles.

Ce système comprend trois parties (voir figure 1) :

1. Un travailleur mobile qui peut utiliser soit une tablette ou un casque vidéo,
2. Une interface d'expert fixe composée d'une table interactive et,
3. Un service web qui assure une communication entre les acteurs cités ci-dessus.

Cette plate-forme distribuée permet la collaboration entre les techniciens et les experts à distance en temps réel via un serveur Web, qui assure les communications entre le technicien et l'expert en transférant la vidéo et les gestes de la main (scénario de maintenance).

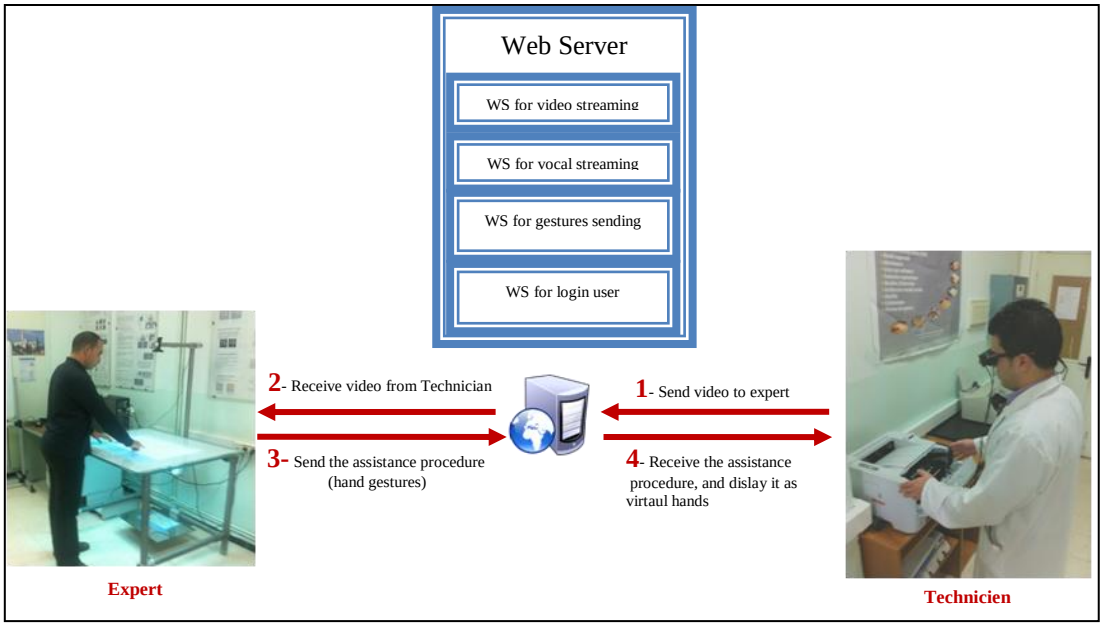


Figure 1. Architecture globale du système « Remote Gestures » (Zenati-Henda et al. 2014).

Ainsi, lorsque que le technicien a besoin d'aide, il utilise une tablette ou un HMD, pour communiquer avec l'expert distant. Il envoie donc le flux vidéo à l'expert. L'interface de l'expert est dotée d'un module de reconnaissance des gestes de la main qui détecte et reconnaît les gestes effectués par l'expert. En temps réel, la description des gestes et la position des mains sont envoyées au technicien. Le système d'affichage prend en compte la description et la position des mains expertes gestes reçus, et visualise des mains virtuelles faisant les mêmes gestes (figure2).

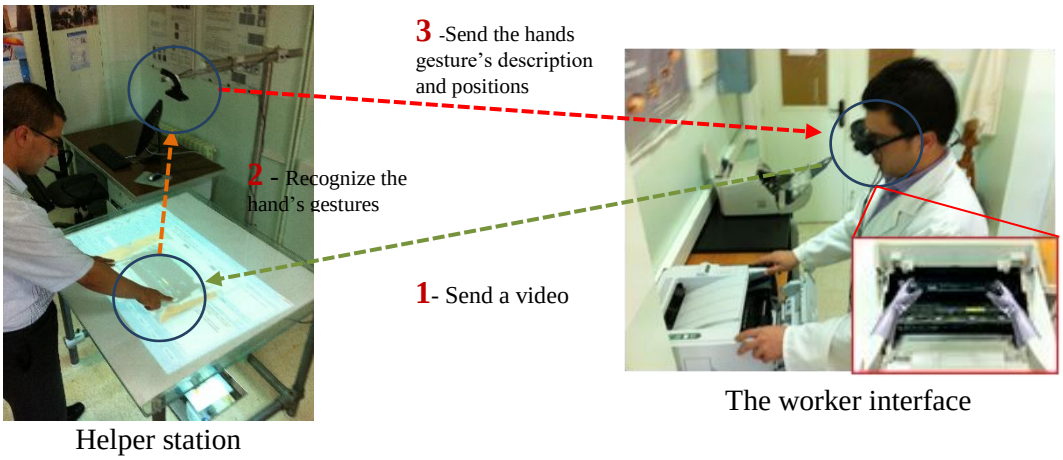


Figure 2. Capture des gestes et transmission au technicien (Zenati-Henda et al. 2014).

3. Table interactive

L'architecture de table est présentée comme suit (figure 3): L'espace de travail de la table est un verre semi transparent avec un projecteur sous la table, les utilisateurs sont assis / debout devant la table. En haut, une caméra vidéo est fixe, et surveille la zone de travail et les mains de l'utilisateur. Le retour visuel de la détection des mains, la reconnaissance des gestes et le résultat de l'action de contrôle sont projetés directement sur la table (Bellarbi et al. 2013b).

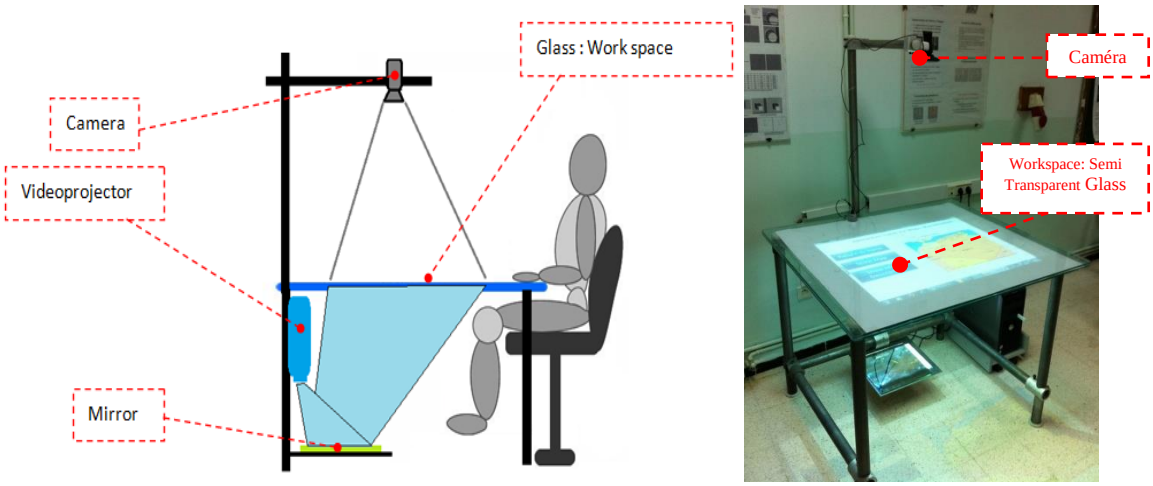


Figure 3. Architecture de la table interactive (Bellarbi et al. 2013b).

- ***Détection et extraction de la main***

Une fois que l'image est captée par la caméra ci-dessus, une étape de binarisation est effectuée. L'architecture de la table fait que l'image capturée est entièrement illuminée par la rétroprojection, sauf l'endroit où est posée la main qui couvre cette projection. Ainsi, l'application d'une technique de binarisation adaptative nous permet de garder uniquement la main sur l'image après sa binarisation. Un ensemble de prétraitements est également appliqué sur cette image binarisée pour réduire les bruits qui apparaissent sur cette dernière (Figure 4).

Figure 4. Binarisation de l'image capturée.

Afin d'extraire la main de l'image binarisée, nous avons proposé un algorithme (Bellarbi et al. 2013b) basé sur un calcul des histogrammes vertical et horizontal de l'image hybridée avec la technique d'Otsu (Otsu 1979). Cette approche nous a permis une extraction efficace de la main

(figure 5). Une fois la main extraite, nous avons normalisé la taille de la main extraite afin de la rendre invariante au changement d'échelle.

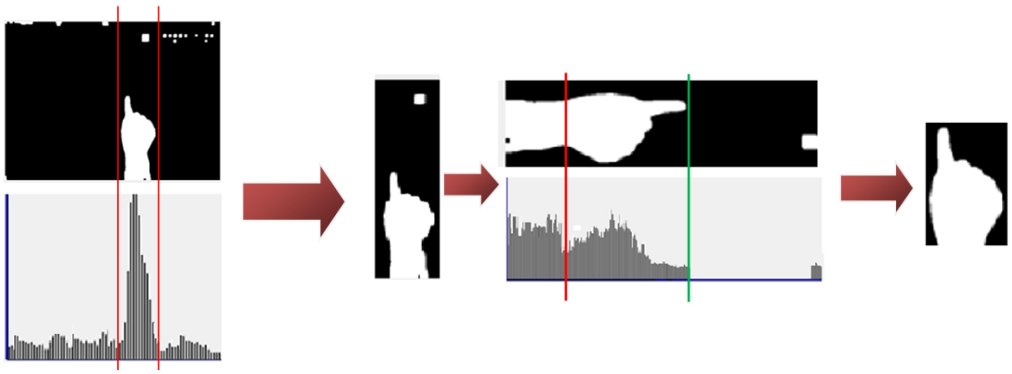


Figure 5. Extraction de la main (Bellarbi et al. 2013b).

- ***Extraction des paramètres et reconnaissance des gestes***

Une fois la main extraite, nous procédons à la phase de l'extraction des paramètres du geste afin de faciliter sa reconnaissance. Pour cela, nous avons utilisé la méthode code chainé (Freeman Chain Code) (Bellarbi et al. 2013b) pour coder l'image de la main extraite en une chaîne de chiffres. La figure ci-dessous (Figure 6) décrit le principe de l'algorithme du code chaîné.

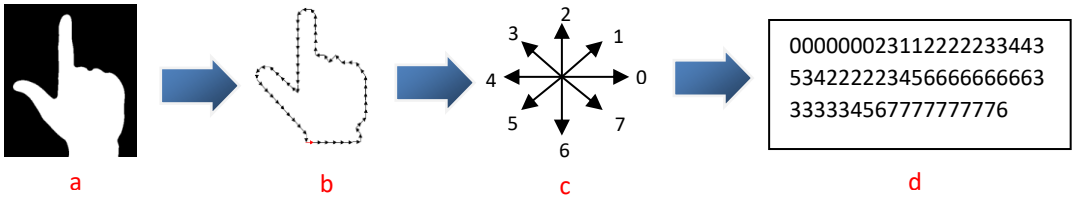


Figure 6. Principe du Chain Code ; a) main extraite, b) contour de la main, c: les huit (8) directions codes, d) chaine de code calculée. (Bellarbi et al. 2013b).

Une fois que le code du geste est calculé, une mise en correspondance est effectuée entre les chaînes dans la base de données des gestes et le geste effectué afin de le reconnaître. Pour cela nous avons appliqué une version modifiée de la technique « Approximate String Matching » (Bellarbi et al. 2013b) en rajoutant un coefficient ($c_{i,j}$) (Equation 1) pour chaque direction afin de minimiser l'erreur des non-correspondances.

$$c_{i,j} = \begin{cases} |a_1(i) - a_2(j)|, if\ |a_1(i) - a_2(j)| \le 4 \\ 8 - |a_1(i) - a_2(j)|, otherwise \end{cases} \quad \text{.....(1)}$$

Cette méthode donne la plus petite distance pour la séquence directionnelle la plus proche. En outre, l'appariement est invariant à la rotation de la main.

4. Test du projet « Remote Gestures »

Nous avons utilisé cette table interactive comme une interface d'expert de la maintenance. Ce dernier, quand il reçoit une demande d'assistance de la part d'un technicien (vidéo en temps réel), peut le guider en montrant des gestes (Figure 7), qui sont à leur tour reconnus et envoyés au technicien distant. Ainsi, en utilisant une tablette ou un casque HMD (Figure 8), le technicien peut voir des modèles 3D des gestes reproduisant les gestes faits par l'expert (voir Figure 9).

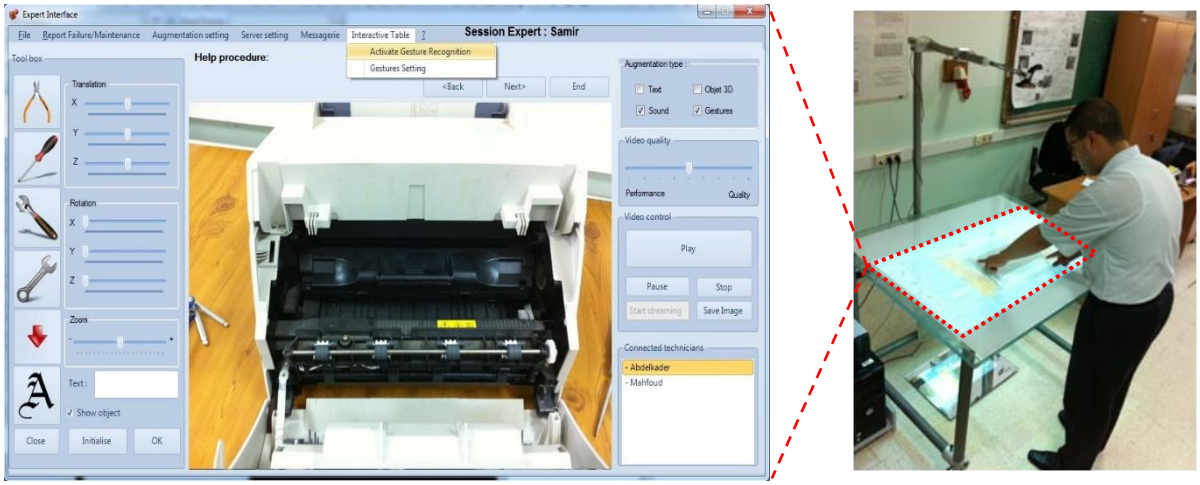


Figure 7. Un expert utilise la table interactive.

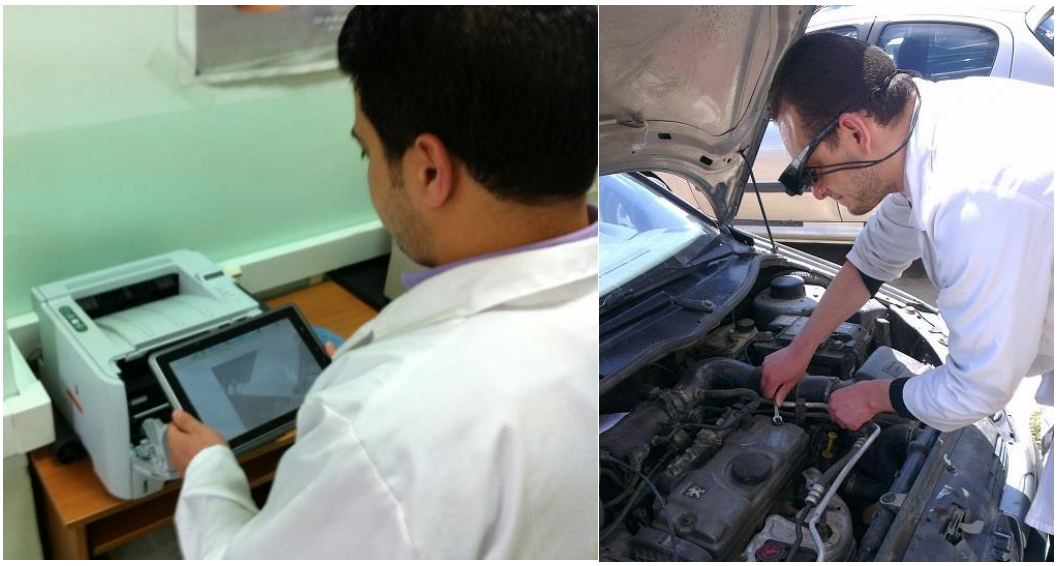


Figure 8. Gauche : Un technique utilise une tablette pour réparer une imprimante. Droite : un technicien utilise un casque pour réparer un moteur d'une voiture.

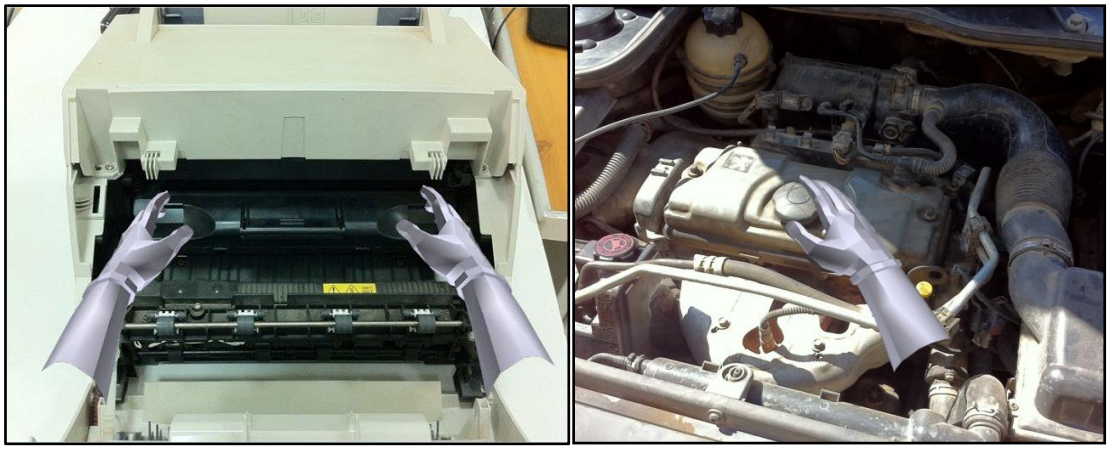


Figure 9. Gestes reçu par l'expert montrant une procédure de maintenance. Gauche : cas d'un changement du toner d'une imprimante. Droite : procédure de changement d'huile d'un moteur.

Ce projet « Remote Gestures » est en cours de développement au sein de l'équipe IRVA du CDTA. Des premiers tests et des évaluations préliminaires sur ce projet ont été présentés dans (Zenati-Henda et al. 2014).

université PARIS-SACLAY | ÉCOLE DOCTORALE Sciences et technologies de l'information et de la communication (STIC)

Titre : Vers l'immersion mobile en réalité augmentée : une approche basée sur le suivi robuste de cibles naturelles et sur l'interaction 3D.

Mots clés : réalité augmentée, interaction 3D, vision par ordinateur, descripteur et estimation de pose.

Résumé : L'estimation de pose et l'interaction 3D sont les fondements de base d'un système de réalité augmentée (RA). L'objectif de cette thèse étant de traiter ces deux problématiques, nous présentons dans ce mémoire un état de l'art qui regroupe : approches, techniques et technologies relatives à l'estimation de pose et à l'interaction 3D en RA. Puis nous faisons le bilan sur les travaux menés jusqu'à aujourd'hui. A cet effet, nos contributions dans ce vaste domaine sont dans les deux parties : vision et interaction 3D. Nous avons proposé un nouveau détecteur et descripteur binaire nommé MOBIL qui effectue une comparaison binaire des moments géométriques. Par la suite nous avons proposé deux améliorations de notre descripteur. MOBIL_2B et POLAR_MOBIL.

En outre, nous avons utilisé notre descripteur avec l'approche PTAM (Parallel tracking and Mapping) afin d'assurer le recalage des objets virtuels en immersion mobile de l'utilisateur en RA.

Nous avons également proposé une technique d'interaction pour la RA, appelée « Zoom-in » qui facilite la sélection et la manipulation des objets virtuels distants. Cette technique est basée sur le zoom de l'image et des objets virtuels recalé sur l'image. Les objets virtuels sont mis à la portée de l'utilisateur en gardant le recalage par rapport à la scène.

Ce mémoire se termine par une conclusion générale qui fait le point sur l'essentiel de ce travail et ouvre de nouvelles perspectives.

Title: Toward mobile immersion in augmented reality: An approach based on robust natural feature tracking and 3D interaction.

Keywords: augmented reality, 3D interaction, computer vision, binary descriptor.

Abstract: Pose estimation and 3D interaction are the essential basis for any Augmented Reality (AR) system. We aim to treat those two fields in order to offer a pertinent AR system that allows a mobile immersion and natural interaction. In this optic, this thesis provides an overall consistent state of the art in both pose estimation and 3D interaction for AR.

In addition, this thesis details our contributions that consists of MOBIL: a binary descriptor that compares geometric moments of the patch through a binary test. Two improvements of this descriptor: MOBIL_2B and POLAR_MOBIL are proposed in order to enhance its robustness. We used this descriptor with PTAM technique to ensure the user pose estimation respectively for the selection/manipulation task and the navigation task.

On the other hand, we proposed a novel 3D interaction technique called "Zoom-In", designed for augmented reality applications. This technique is based on the zoom of the captured image. It calculates the 3D transformation relative to the selected object. This technique allows user selecting and manipulating distant virtual objects by bringing them within the user arm's reach by zooming in the captured image, and re-estimating the user pose thanks to our proposed descriptor.

Finally, we present a conclusion that describes the essential of this work and provide perspective and future work.