République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Mouloud Mammeri de Tizi-Ouzou (UMMTO).

Faculté du Génie Electrique Informatique.

Département informatique.

C. D. T. A.

Mémoire de Master

Pour l'obtention du diplôme de Master recherche en Informatique

Option : Systèmes informatique (SI)

Thème :

Approche de localisation basée sur les points d'intérêts : Application à la réalité augmentée

Réalisé par :

- Hayet BELGHIT

Proposé par :

- Mme Nadia Henda-Zenati
- M. Abdelkader BELLARBI

Encadré par :

- M. Yazid CHAIEB

Devant le jury composé de :

- Mme Amirouche …… Présidente
- Mr A. Dib…… Examinateur
- Melle Y. Yesli …….…… Examinatrice

- **Promotion : 2014/2015**

RESUME

Le monde des interfaces homme-machine tend à évoluer avec l'avancée qu'a connue la réalité augmentée durant cette dernière décennie. La réalité augmentée (RA) consiste à insérer des objets virtuels dans une scène réelle. Ce paradigme offre un moyen d'enrichir notre environnement à l'aide d'objets virtuels (texte, flèches ou autres) qui peuvent être des indications visuelles permettant de guider l'utilisateur à l'accomplissement d'une tache donnée.

Notre travail s'inscrit dans le cadre d'un projet de recherche initié au niveau du CDTA au sein de l'équipe IRVA dont les principaux axes de travail sont : la réalité augmentée, la réalité virtuelle et l'interaction homme-système.

Notre travail consiste en le suivi d'un objet rigide en se basant sur les points d'intérêts afin de pouvoir déterminer le point de vue de l'utilisateur et insérer un objet virtuel dans la scène.

Ce présent mémoire englobe de ce fait les concepts de base permettant de réaliser une application de RA. Il est vrai que l'état de l'art est plus que riche en concepts, méthodes et techniques pour tous ce qui concerne la RA, néanmoins, nous avons essayé à travers les deux premiers chapitres de présenter les définitions et concepts nécessaires pour la compréhension de ce qu'est la réalité augmentée, nous avons également présenté une classification des outils nécessaires à la réalisation d'une application de RA.

La deuxième partie du mémoire montre d'un point de vue pratique, comment mettre en œuvre les outils présentés précédemment ? Quels sont les outils matériels et logiciels à utiliser ? L'objectif de cette démarche est d'initier les éventuels lecteurs de ce document à la réalité augmentée et de leurs permettre de se lancer dans ce domaine passionnant et qui plus est d'actualité.

ABSTRACT

The world of HMI (Human-Machine Interfaces) tends to evolve within the progress of augmented reality during the last decade. Augmented reality (AR) consists of inserting virtual objects in the real world. This paradigm provides a way to enrich our environment with virtual objects (text, arrows … etc.) which can be hopeful to be used as a visual indications to guide the user to accomplish a given task.

Our work is part of a research project initiated at the CDTA within IRVA team whose main areas of work are: augmented reality, virtual reality and human-system interaction. Our work consists of tracking a rigid object based on points of interest in order to determine the user view point for an augmented reality application. The main objectives arising are: tracking an object in the scene, estimating the camera pose (location) and inserting a virtual object in the scene.

This work encompasses the basic concepts for performing an AR application. The state of the art is very rich in concepts, methods and techniques of AR, so, we tried through the first two chapters to present the main definitions and concepts that aim to familiarize the reader with augmented reality, we also presented a classification of the main tools used on an implementation (application) of an AR System. The second part of this work shows how to implement the tools presented above? Which equipment can be used? Which software tools should be used? Our aim is to initiate the potential readers of this document to the exciting domain of augmented reality and allow them to start with this relevant domain.

ملخص

نقصد بالحقيقة المعززة إضافة مجسم إفتراضي في العالم الحقيقي مع مراعات أبعاد العالم الحقيقي.

هذا العمل يدخل في نطاق برنامج بحث لفريق ضمن مركز تنمية التكنولوجيات المتطورة. إن أهم مجالات إهتمام الفريق تدور حول العالم المعزز، العالم الإفتراضي والإنفعال بين الإنسان و الآلة.

الهدف من هذا العمل إنجاز برنامج آلي يعمل في وقت حقيقي على تتبع مجسم إعتمادا على نقاط خاصة و بإستعمال المعلومات الناتجة عن التتبع يمكننا إحتمال موقع الكميرا في العالم الحقيقي وبالتالي نستطيع إضافة مجسم خيالي وجعله مندمج مع الواقع وكذا نحصل على واقعنا المعزز.

هذه المذكرة تقدم من الناحية الأولى المفاهيم و المبادئ الأساسية للعالم المعزز إضافة إلى الوسائل النظرية المستعملة في هذا المجال. من الناحية الأخرى نقدم الجانب التطبيقي وكيفية إستغلال المبادئ النظرية لتحقيق برنامج يولد عالما معززا.

REMERCIEMENT

Ce mémoire que nous avons réalisé en quelques mois de travail, est en fait le résultat des efforts de plusieurs personnes que nous tenons à remercier.

Tout d'abord, nous exprimons toute notre gratitude aux deux institutions qui nous ont permis de faire ce master, l'Université de Mouloud Mammeri de Tizi-Ouzou (UMMTO) d'une part et le Centre de Développement des Technologies Avancées (CDTA) d'une autre part.

Nous tenons aussi à remercier respectivement, notre promoteur de l'université Mr Yazid Chaieb et nos encadreurs au niveau du CDTA Mme Nadia Henda-Zenati et Mr Abdelkader Bellarbi.

Nous adressons également nos remerciements, à tous les enseignants, qui se sont succédé durant toute notre formation,

Nous remercions très sincèrement, les membres du jury d'avoir bien voulu accepter d'examiner ce travail,

Pour finir, merci à toute personne ayant contribuée de près ou de loin à l'accomplissement de ce modeste travail.

DEDICACE

Je dédie ce mémoire à…

A ma très chère mère, honorable, aimable : Tu représentes pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple du dévouement qui n'a pas cessé de m'encourager et de prier pour moi.

Je te dédie ce travail en témoignage de mon profond amour. Puisse Dieu, le tout puissant, te préserver et t'accorder santé, longue vie et bonheur.

A mon très cher père, rien au monde ne vaut les efforts fournis jour et nuit pour mon éducation et mon bien être. Ce travail est le fruit des sacrifices que tu as consentis pour mon éducation et ma formation.

Aucune dédicace ne saurait exprimer l'amour, l'estime, le dévouement et le respect que j'ai toujours eu pour toi.

A la mémoire de mes deux grands-mères "Yaya Nora" et "Yaya Dahbia" que Dieu ait pitié de leurs âmes.

A toutes les personnes chères à mon cœur : frères, belles sœurs et beaux-frères, oncles et tantes, cousins et cousines, à mes amis et toutes ma famille. Un petit clin d'œil particulier à Kad qui n'a cessé de m'encourager pour travailler et avancer, je te remercie infiniment du soutien que tu m'as apporté.

A toutes les personnes que j'aime, en témoignage des liens qui nous unissent, je vous dédie ce travail et je vous souhaite une vie pleine de santé et de bonheur.

TABLES DES MATIERES

Contents

LISTE DES FIGURES

INTRODUCTION GENERALE

La vision par ordinateur (aussi appelée vision artificielle ou vision numérique) est une branche de l'intelligence artificielle qui désigne la compréhension d'une scène ou d'un phénomène à partir d'informations « images », liant intimement perception, comportement et contrôle. En effet le principale but de cette discipline est de permettre à une machine d'analyser, traiter et comprendre une ou plusieurs images prises par un système d'acquisition, de ce fait, la vision par ordinateur repose en grande partie sur les méthodes de traitement d'image.

Par traitement d'image, on entend l'ensemble des techniques permettant de modifier une image dans le but de l'améliorer ou d'en extraire des informations. Les domaines d'application du traitement d'images sont très variés, nous pouvons citer par exemple : la robotique, la télé-opération, la vidéo-surveillance, la reconnaissance d'objet ou particulièrement de visage, l'imagerie médicale…etc.

L'évolution du traitement d'image et de la vision par ordinateur a permis l'apparition d'un nouveau paradigme appelé « Réalité Augmentée ». Ce dernier, a pris un grand élan cette dernière décennie vue les avantages qu'il présente en matière d'assistance et d'orientation dans différents domaines.

La réalité augmentée est définit comme étant le fait de rehausser l'environnement réel par des éléments virtuels tels que des informations textuelles ou encore des objets virtuels, mais en laissant l'utilisateur en contact avec son environnement réel.

Une composition réaliste du réel et du virtuel demande de résoudre essentiellement trois catégories de problèmes : la détermination du point de vue, la prise en compte des occultations entre éléments réels et virtuels, ainsi que les interactions lumineuses entre ces éléments. La première catégorie a été l'objet de plusieurs travaux, ces travaux traitent deux types de problèmes : l'alignement spatial (calibrage) et l'alignement temporel ou le suivi (tracking).

Notre travail rentre dans le cadre du projet « IM@REV » (Pour Interaction 3D Multimodale et Collaborative dans un environnement de Réalité Virtuelle et Augmentée) initié au sein de l'équipe IRVA (Interaction Homme-Système et Réalité Virtuelle/Augmentée) de la division robotique et productique du centre de développement des technologies avancées CDTA. Il s'agit de réaliser une application de réalité augmentée qui assure le suivi d'un objet rigide afin de pouvoir déterminer le point de vue de l'utilisateur et insérer des objets virtuels dans la scène réelle. Le suivi est assuré par la détection d'indices visuels sur l'image (points d'intérêts), les informations extraites seront utilisées pour calculer les paramètres de la caméra, à savoir les paramètres intrinsèques qui définissent les caractéristiques internes de la caméra et les paramètres extrinsèques qui caractérisent la position de la camera (appelée aussi la pose de la camera) par rapport à un repère 3D fixe. Ces paramètres permettent d'insérer l'objet virtuel 3D dans l'image en cours et l'aligner correctement par rapport à la scène réelle.

De ce fait, le présent mémoire est décomposé en quatre chapitres :

- Chapitre 1 : Réalité Augmentée (Concepts et Définitions)

Ce premier chapitre nous permet d'introduire la réalité augmentée, les différents problèmes rencontrés pour la réalisation d'un système de *RA*, ses domaines d'application ainsi que les dispositifs utilisés dans ce genre de système.

- Chapitre 2 : Approches de Localisation en Réalité Augmentée

Nous abordons dans ce second chapitre le problème de suivi d'un objet rigide basé sur la reconnaissance de l'objet à suivre en utilisant les points d'intérêts et le problème de recalage 3D des objets virtuels, nous commençons d'abord par présenter les différents détecteurs de points d'intérêts et les descripteurs existants, ensuite nous passons, à la modélisation de la camera et aux méthodes de calcul de pose de la camera.

- Chapitre 3 : Modélisation du Système

Nous présentons dans cette partie, les éléments essentiels rentrant dans la modélisation de notre système, afin de dégager les différents modules et de choisir les méthodes à implémenter.

- Chapitre 4 : Implémentation et Testes

Ce chapitre nous permet d'aborder le travail effectué et de présenter quelques résultats qui sont la concrétisation de la partie modélisation et de l'implémentation.

Nous terminerons finalement par une conclusion générale où nous résumons l'apport essentiel de ce travail.

Chapitre 1 : La Réalité Augmentée, Concepts et Définitions

LA REALITE AUGMENTEE

1. Introduction

Ce présent chapitre nous permet de dégager les concepts de base, et de comprendre la notion de Réalité Augmentée (RA). En effet dans un premier temps nous définissons la Réalité Virtuelle, ensuite nous passons en revue quelques définitions de la Réalité Augmentée, puis un lien entre ces deux concepts est établi. Nous aborderons également les problèmes liés à la RA, puis nous présenterons les domaines d'applications, les dispositifs d'entré sortie et nous terminerons par une brève conclusion.

2. Définitions

2.1.La réalité virtuelle

La réalité virtuelle communément abrégée RV, est considérée par certains auteurs comme une extension des Interfaces Homme-Machine classiques. Les interfaces résultantes dites « avancées » simulent des environnements réalistes et permettent à des participants d'interagir avec ceux-ci :

"Virtual Reality is an advanced human-computer interface that simulates a realistic environment and allows participants to interact with it [Ellis et al., 1994]."

[Fuchs et al., 2006] donne plusieurs définitions de la RV dont:

- Définition fonctionnelle :

« La réalité virtuelle va permettre de s'extraire de la réalité physique pour changer virtuellement de temps, de lieu et (ou) de type d'interaction : interaction avec un environnement simulant la réalité ou interaction avec un monde imaginaire ou symbolique ».

- Définition technique :

« La réalité virtuelle est un domaine scientifique et technique exploitant l'informatique et des interfaces comportementales en vue de simuler dans un monde virtuel le comportement d'entités 3D, qui sont en interaction en temps réel entre elles et avec un ou plusieurs utilisateurs en immersion pseudo-naturelle par l'intermédiaire de canaux sensori-moteurs ».

A partir de toutes les définitions de la RV nous pouvons dire que la finalité est de permettre à une ou plusieurs personne (s) des activités sensori-motrices dans un monde

artificiel, qui est soit imaginaire ou symbolique, soit une simulation de certains aspects du monde réel.

2.2.La réalité augmentée

Les systèmes de réalité augmentée permettent la superposition des images virtuelles sur des scènes réelles. Le concept de la réalité augmentée enrichit notre perception du monde réel, en y ajoutant des informations de manière dynamique et interactive. L'objectif de la réalité augmentée est d'apporter un réalisme et une cohérence visuelle entre le flux réel et virtuel (voir Figure 1. 1).

La réalité augmentée a vu le jour avec les travaux de Sutherland [Sutherland, 1965] [Sutherland, 1968], qui a réalisé le premier système de réalité augmentée, basée sur un casque de réalité virtuelle transparent «See Through System». En effet, Sutherland a introduit le concept de contact entre l'homme et la machine, en plaçant un utilisateur à l'intérieur d'un environnement en trois dimensions généré par ordinateur. Ce système permet à l'utilisateur de visualiser et de naviguer autour d'éléments virtuels positionnés dans l'espace réel. Durant les années 80, le concept de réalité augmentée a été surtout utilisé dans un cadre militaire, pour l'affichage d'information sur les visières des casques des pilotes d'avions « Head-Up Display ».

La réalité augmentée regroupe l'ensemble des techniques permettant d'intégrer des éléments virtuels (images de synthèse, objets virtuels, graphiques, etc) dans un monde réel. Ronald Azuma [Azuma, 1997] a défini les trois règles de base nécessaires pour le fonctionnement d'un système de réalité augmentée, à savoir :

- Combiner le réel et le virtuel.
- Respecter les contraintes d'interactivité et de temps.
- Respecter l'homogénéité et la cohérence entre les deux mondes réel et virtuel.

D'une manière générale, la réalité augmentée consiste à augmenter la scène réelle avec des informations virtuelles supplémentaires. Cette augmentation peut prendre différentes formes.

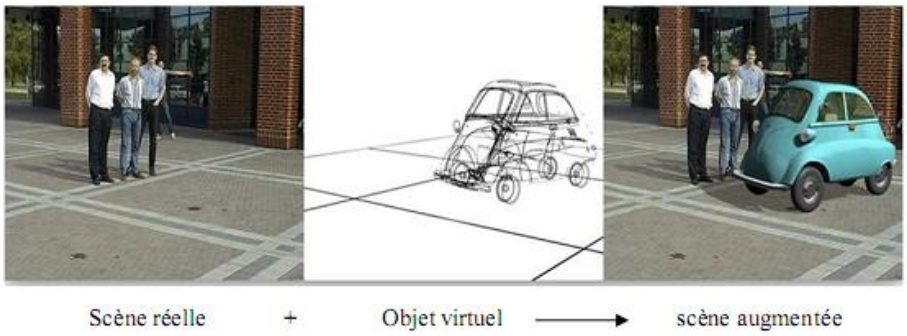


Figure 1. 1 Exemple de la réalité augmentée.

2.3.Continuum Réalité Augmentée/Virtuelle

Afin d'unifier la notion de RA et RV, dans leur travaux [Milgram et al., 1995] ont défini un continuum qui relie les deux domaines tout en mettons en avant la notion de Réalité Mixte (voir Figure 1. 2), la RA et la RV sont tous deux distinguables par leur Environnement dominant. Dans le cas de la virtualité augmentée, l'environnement est majoritairement virtuel avec insertion d'objets réels, de même en réalité augmentée, l'environnement est majoritairement réel avec insertion d'objets virtuels.

Ainsi la taxonomie décrit la relation entre la RA et la réalité virtuelle (RV). Dans la RV, l'utilisateur est immergé dans un monde synthétique, et sa perception est isolée de l'environnement réel. De ce fait, le continuum « Réalité-Virtualité » (Figure 1. 2) est présenté de la manière suivante : le monde réel et un environnement complètement virtuel sont les deux extrémités de ce continuum, dont la région intermédiaire est nommée Réalité Mixte.

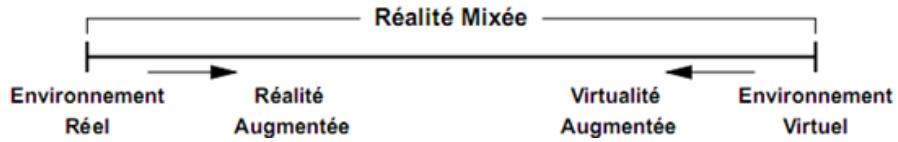


Figure 1. 2 Continuum Réalité-Virtualité [Milgram et al., 1995].

3. Challenges de la réalité augmentée

Les systèmes de RA ont beaucoup évolué durant cette dernière décennie avec l'avancement technologiques et l'évolution des algorithmes de traitement d'image qui ont répondu jusqu'à un certain point aux besoins de ce genre d'applications. Néanmoins, les challenges de la RA restent toujours les même, de ce fait, la réalisation de ce genre de systèmes tel que présentée dans [Rabbi et al., 2013], demande la prise en compte d'un ensemble de problèmes, à savoir : l'alignement, la visualisation, l'interaction, la performance et la mobilité.

3.1. Alignement spatio-temporel

Le problème de l'alignement revient à faire correspondre les perspectives de l'objet virtuel avec celle de la scène réelle, en d'autres termes positionner correctement les objets virtuelle par apport au point de vue de l'utilisateur.

De ce fait l'alignement spatio-temporel revient à déterminer le point de vu de l'utilisateur qui est généralement considéré comme étant identique au point de vue de la camera. Ce qui se traduit en un calcul des paramètres intrinsèques et extrinsèques de la camera.

Un mauvais alignement peut engendrer de nombreuses conséquences, ces dernières dépondent du domaine d'applications, prenant l'exemple de la médecine ou les conséquences d'une erreur de positionnement des objets virtuels peuvent être graves.

3.2.Visualisation

Le challenge de la visualisation incluse deux catégories de problèmes, la première est liée aux problèmes d'affichage, contraste, résolution, luminosité et le champ de vision. Ainsi, l'éclairage des objets virtuels et celui du monde réel devrait être similaire, la prise en compte de l'ombrage des objets virtuels par exemple ajouterai du réalisme à la scène augmentée.

La seconde catégorie consiste en la gestion des occultations, en effet déterminer quelle surface ou parties ne sont pas visibles ou sont visibles à partir d'un certain point de vue est un problème majeur à prendre en compte dans les applications de RA.

La figure ci-après présenté par [Simon et al., 2006] illustre les deux défis présentés précédemment. Celle-ci propose quatre images A, B, C, et D, dans lesquelles on essaye d'intégrer une voiture et de corrigé un problème à chaque image.

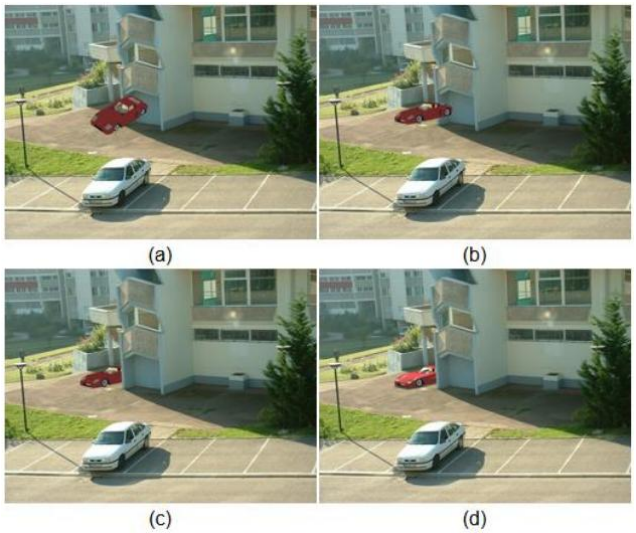


Figure 1. 3 Illustration des deux challenges (Alignement et Visualisation) [Simon et al., 2006].

L'image (a), représente une intégration arbitraire de l'objet virtuel.

L'image (b), correction du problème d'alignement.

L'image (c), relève le défi d'occultation.

L'image (d), respect de la cohérence (luminosité, contraste).

3.3.Les performances

Le traitement d'image, les traitements mathématiques et les modèle 3D, sont tous des traitements gourmant en ressources ce qui atténue rudement les performances et rend le traitement en temps réel difficile, or ce sont des traitements nécessaires pour la réalisation d'application de RA, de ce fait l'amélioration de performances est un défi majeur de la RA.

3.4.L'interaction

Le défi de l'interaction en RA consiste en l'interaction des utilisateurs avec les objets virtuels, autrement dit sélection et manipulation des objets virtuels ou encore « action/réaction » entre l'utilisateur et les objets virtuels. L'évolution de ce domaine risque de bouleverser le monde de l'interaction et des interfaces homme-machine.

3.5.La mobilité

Le défi consiste en la portabilité des systèmes de RA, en effet les applications de RA se veulent de plus en plus mobile que ce soit de la RA en intérieur ou extérieur. Ainsi l'évolution technologique des dispositifs mobile à rendu la « RA mobile » envisageable mais toutefois non acquise.

4. Domaines d'application

De par son originalité et sa pluridisciplinarité en matière de technologie, la Réalité Augmentée (RA) 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 cette section nous présenterons en détail quelques applications de RA.

4.1.Médecine

De nombreux travaux de RA ont été réalisés dans le domaine médicale, nous pouvons citer ceux de Haoucine et al., qui propose dans [Haouchine et al., 2013] 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 calculée à partir des données préopératoires peuvent être superposées sur la vidéo laparoscopique.

La figure 1. 4 montres 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).

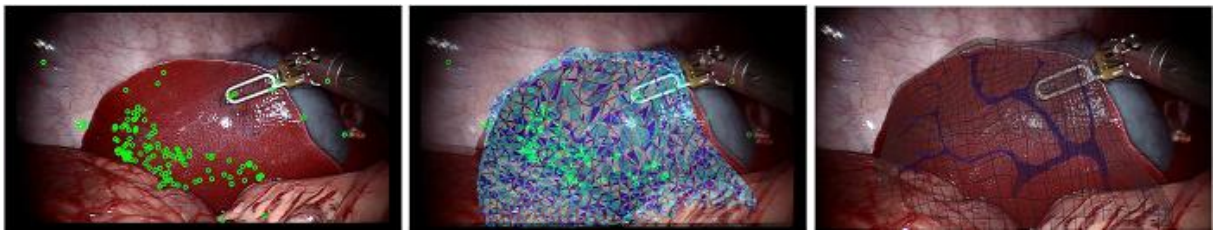


Figure 1. 4 Superposition des vaisseaux sanguins sur la vidéo laparoscopique [Haouchine et al., 2013].

4.2.Architecture

Nasman et al., présentent dans [Nasman et al., 2012] une étude d'une interface utilisateur tangible (TUI) pour la conception architecturale et l'analyse de l'éclairage naturel. Cet outil offre un moyen intuitif pour les architectes et les futurs occupants du bâtiment pour construire rapidement des modèles physiques et puis afficher une simulation de l'éclairage naturel dans le modèle à taux interactifs (voir Figure 1. 5).

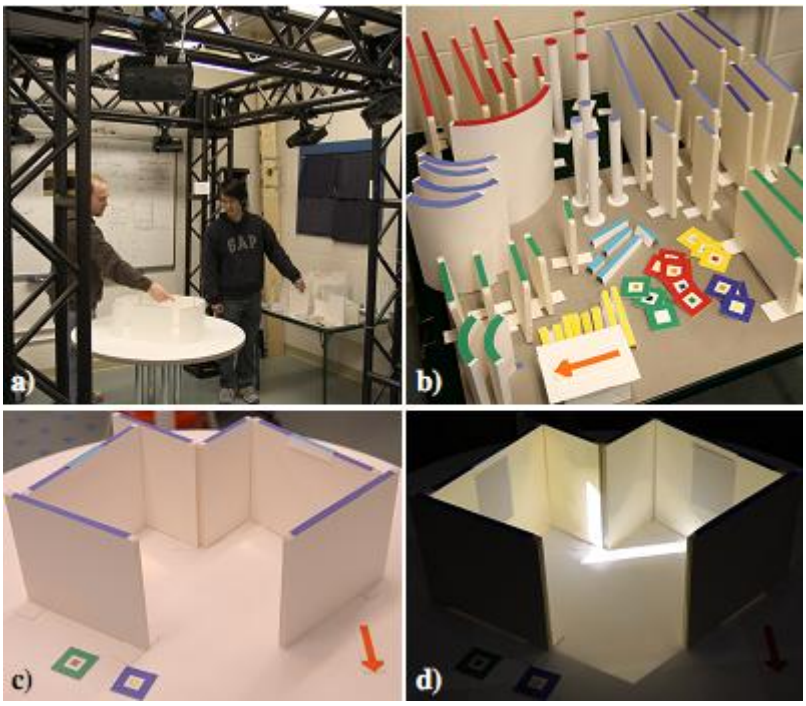
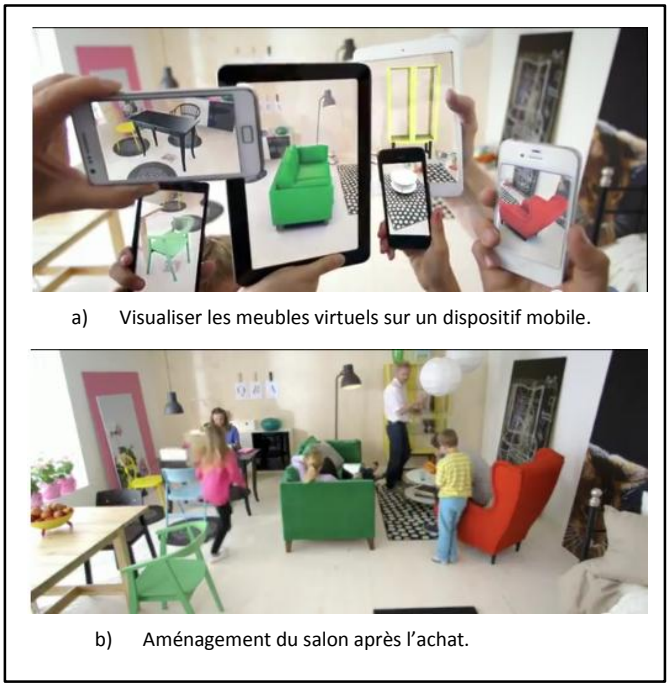


Figure 1. 5 interface utilisateur tangible pour la conception de l'éclairage naturel architecture avec : a) un environnement de croquis physique, b) un ensemble de primitives de murs et de fenêtres et matérielles, c) construction d'une esquisse d'une conception architecturale, d) simulation de l'éclairage naturel est projetée sur les surfaces [Nasman et al., 2012].

4.3.Marketing

Dans le domaine du marketing, nous pouvons citer IKEA [IKEA, 2015] qui offre la possibilité de placer des meubles virtuels dans votre propre maison par la numérisation des pages sélectionnées dans le catalogue IKEA 2015 imprimé à la demande de catalogue IKEA (disponible pour iOS et Android) ou en parcourant les pages dans le catalogue numérique sur votre smartphone ou tablette. Il suffit ensuite de placer le catalogue IKEA imprimée où vous voulez mettre les meubles dans votre chambre ou salon, choisir un produit à partir d'une sélection de la gamme IKEA et voir à quoi ressemblerai la pièce avec l'élément ajouté.



a) Visualiser les meubles virtuels sur un dispositif mobile.

b) Aménagement du salon après l'achat.

Figure 1. 6 Visualisation de meubles virtuels dans un salon, application commerciale d'IKEA [IKEA, 2015].

4.4.Culture et tourisme

Kounavis et al., traitent dans [Kounavis et al., 2012] de l'utilisation de la réalité augmentée (AR) pour les besoins du tourisme. Ils décrivent l'évolution de la technologie des applications pilotes dans les applications mobiles commerciales. Ils abordent les aspects techniques de développement d'applications de RA mobile, en mettant l'accent sur les technologies qui rendent la fourniture de contenu augmentée possible. Le document examine l'état de l'art, fournissant une analyse concernant le développement et les objectifs de chaque application reconnaissant les diverses limitations technologiques obstacles à l'adoption de l'utilisateur final substantielle de RA, le document propose également un modèle pour le développement d'applications mobiles de RA pour le domaine du tourisme, visant à libérer le plein potentiel de la RA dans le domaine.

Comme exemple d'application dans le domaine de la culture et du tourisme nous pouvons parler de l'application mobile citée dans [Hugues, 2011] qui permet aux visiteurs de la ville de Philadelphie d'observer l'ambiance de la ville en fonction des époques comme elle existait dans le passé (voir Figure1. 7).

Figure 1. 7 Visualisation de la ville de Philadelphie au passé [Hugues, 2011].

4.5.Militaire

Livingston [Livingston et al., 2011] examine les avantages et les exigences militaires qui ont conduit à une série d'efforts de recherche en réalité augmentée (RA) et des systèmes connexes au cours des dernières décennies. Ce travail met en évidence quelques-uns des projets de recherche qui ont fait progresser le domaine au cours des dernières décennies.

Sur le terrain, les militaires sont très concentrés et à l'affut des différentes informations tactiques. C'est pourquoi la société Tanagram Partners développe une technologie qui aide les militaires à distinguer les différentes informations en fonction de leur dangerosité. Pour ce faire, les soldats ont des capteurs légers (caméras, microphone, GPS, etc.) qui se trouvent dans leur casque. Les données sont ensuite retransmises à un serveur où elles sont traitées. Puis, de nouvelles données, cette fois en trois dimensions, sont envoyées dans les lunettes des soldats. Elles s'affichent alors sur l'écran des lunettes et permettent aux soldats de distinguer les informations les plus importantes (voir Figure 1.6). Il faut également que le serveur soit puissant, car ce système complexe d'échange d'informations doit se faire en temps réel (en quelques millisecondes) et en tenant compte de l'environnement réel. [Larue et al., 2012].

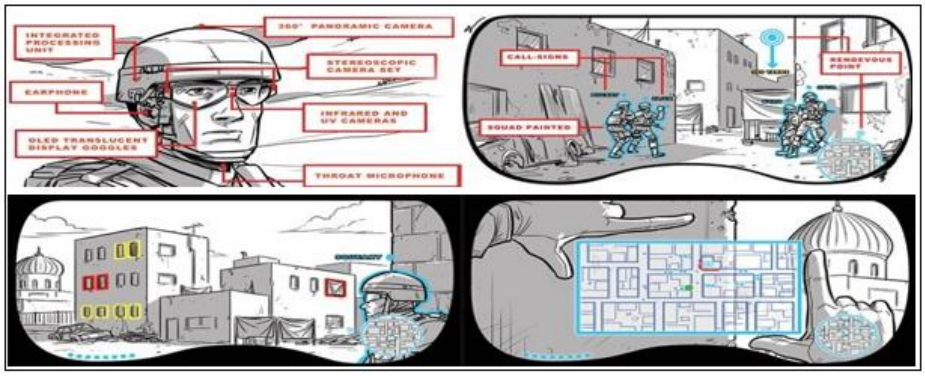


Figure 1. 8 Application de la RA au domaine militaire, issue de [Larue et al., 2012].

4.6.Industrie et maintenance

Parmi les nombreux domaines que peut servir la réalité augmentée nous pouvons citer celui de l'industrie et de la maintenance. Le CDTA (Centre de Développement des Technologies avancées) propose dans [Benbelkacem et al., 2011] une plateforme collaborative de réalité augmentée qui permet l'assistance à la maintenance par le biais des web services. L'idée est de faire appel à un expert distant qui reçoit en temps réel le flux vidéo capturé par le dispositif qu'utilise le technicien, l'opération d'assistance à la maintenance se fait en ajoutant des objets virtuels (flèches, texte, outils..) à titre indicatif pour pouvoir guider au mieux le technicien dans la procédure de réparation.

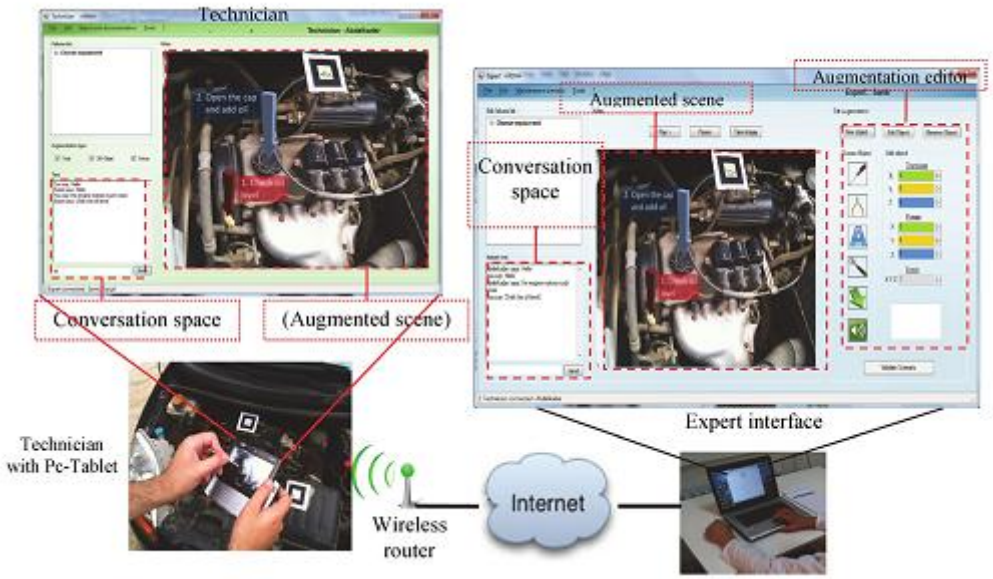


Figure 1. 9 Assistance à la maintenance (cas d'étude : le domaine de l'automobile) *[Benbelkacem et al., 2011]*.

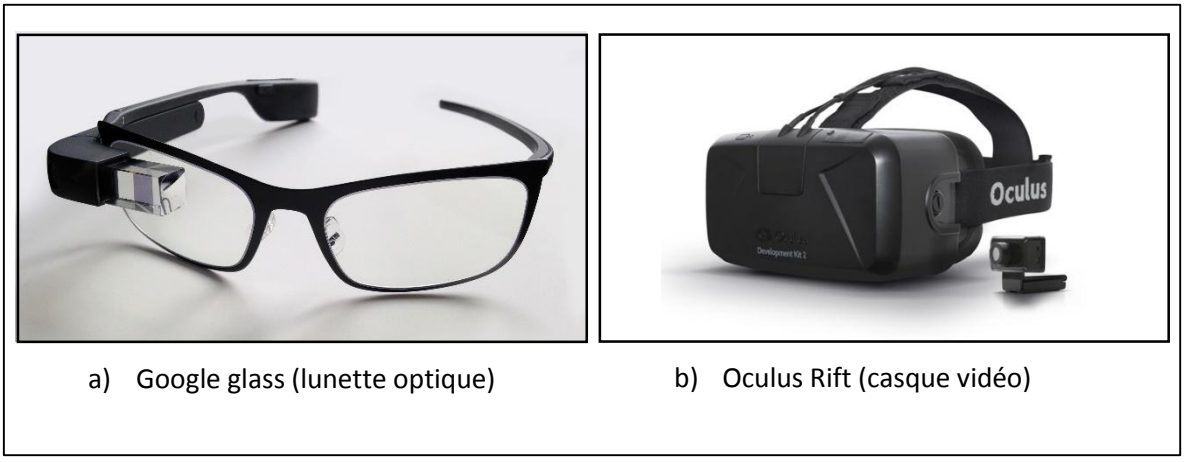
5. Dispositifs d'entrées sorties

Plusieurs dispositifs et capteurs sont utilisés pour réaliser un système de réalité augmentée. Néanmoins, les dispositifs les plus employés sont : les systèmes d'affichage qui jouent un rôle primordial pour le rendu visuel et la génération des scènes virtuelles, et les systèmes de suivi nécessaires à la localisation dans le repère du monde. La plupart des capteurs utilisés en réalité augmentée sont dédiés pour des applications en intérieur. Cependant, de nouvelles technologies nécessitant l'utilisation de nouveaux dispositifs qui sont destinées pour des applications en extérieur ou sans-fil.

5.1. Systèmes de visualisation

Les systèmes d'affichage sont utilisés pour visualiser des objets virtuels. Des dispositifs d'affichage dédiés permettent alors de mixer le réel et le virtuel. On peut distinguer différentes classes de systèmes :

- Les afficheurs de types visiocasque (HMD : Head Mounted Display) : le système est porté sur la tête de l'utilisateur, et se distingue en deux catégories. Les casques dits semi-transparent optique (optical see-through HMD), constitué d'un écran couplé à un miroir semi-transparent (Figure 1.9 (a)) dont le mixage réel et virtuel est fait par l'œil de l'utilisateur. Il y a aussi les casques dits vidéo (video see-through HMD) dont le mixage est fait entre un rendu graphique et l'image provenant d'une caméra (Figure 1.9 (b)), ce mélange étant alors présenté à l'utilisateur.

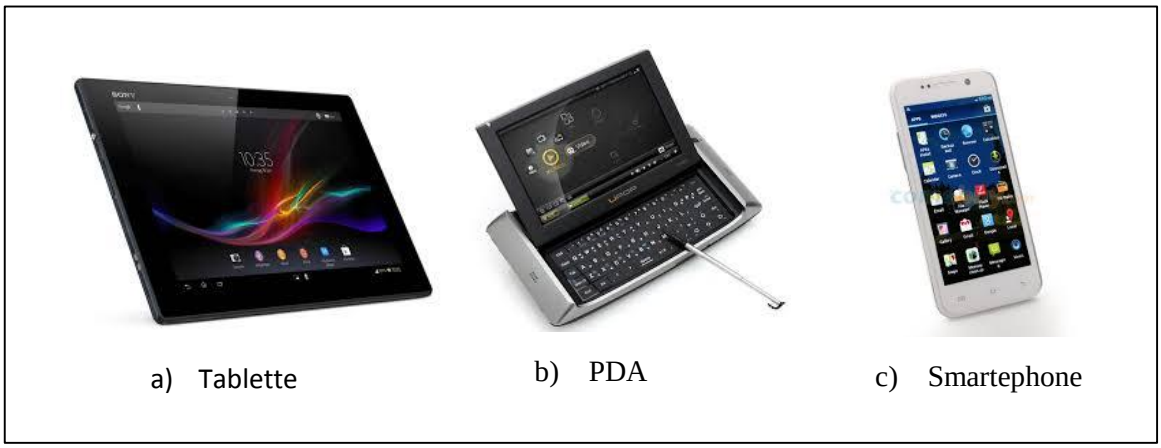


a) Google glass (lunette optique) b) Oculus Rift (casque vidéo)

Figure 1. 10 Examples de casques HMD (Head Mounted Display).

- Les afficheurs de type écran : la visualisation est faite à travers un écran, à l'aide d'une caméra couplée à cet élément. Les dispositifs basés moniteur offrent la possibilité d'une perception de la scène Augmentée sans avoir besoin à porter d'autre matériels sur sois même. Dans cette Catégorie on retrouve notamment: les écrans classique entre autre les écrans LCD, Et pour la mobilité on retrouve le dispositif dit (Hand-

held), commercialisés sur différentes Gammes de produit, tel que les tablettes pc, PDA, smartphone (voirfigure1.10).



a) Tablette b) PDA c) Smartephone

Figure 1. 11 Afficheurs de Type Ecran.

5.2.Capteurs de position et d'orientation

La camera est un capteur indispensable dans une application de réalité augmentée, elle permet d'acquérir des informations concernant l'environnement de l'utilisateur et déterminer sa position par rapport à un repère dans la scène (en utilisant des algorithmes de vision), l'image acquise est restituée sur écran à l'utilisateur avec insertion d'objets virtuels (dans le cas RA-video).

Il existe toutefois, un ensemble de capteurs de position et d'orientation qui peuvent être utilisés en RA afin de localiser un objet rigide en mouvement par rapport à un repère absolu dans l'espace. On s'en sert principalement pour mesurer le mouvement de l'utilisateur. En général, on mesure une série de 6 grandeurs (3 positions et 3 angles) avec une fréquence d'échantillonnage compatible avec la précision recherchée. Il existe de nombreux types de capteurs. On peut les classer par catégories : mécaniques, magnétiques, ultrasonores, optiques, il y'a aussi le capteur GPS (Global Positionning System) : Utilisé principalement pour les applications de réalité augmentée en extérieur. Nous pouvons citer également les technologies Radio Frequency IDentification (RFID) qui mémorisent et récupèrent à distance des données en utilisant des étiquettes et des lecteurs RFID. Les critères de comparaissant utilisés sont les suivants : les degrés de libertés (ddl), la résolution, l'erreur, le temps de réponse, l'espace de travail et la sensibilité.

6. Conclusion

La réalité augmentée est un domaine dans beaucoup de travaux sont en cours. Son approche pluridisciplinaire converge vers un environnement rassemblant plusieurs champs de recherche.

Ce chapitre nous a permis de définir la RA et la RV, et d'établir un lien entre ces deux concepts, nous avons également abordé les problèmes liés au domaine de la RA, les différents domaines d'application ainsi que les dispositifs utilisés en RA.

Le chapitre suivant sera consacré à la localisation en RA, nous aborderons les différentes techniques et méthodes qui peuvent nous aider pour déterminer le point de vu de l'utilisateur.

Chapitre 2 : Approches de Localisation en Réalité Augmentée

1. Introduction

Le problème d'alignement reste le chalenge le plus important des applications de RA. Obtenir le point de vue exacte de la camera et donc primordiale afin d'avoir une scène augmentée cohérente. En effet, une bonne estimation des paramètres de localisation permet de modéliser une caméra virtuelle qui a les mêmes caractéristiques que la caméra réelle, permettant ainsi d'aligner correctement les deux mondes réel et virtuel. L'estimation de la position et de l'orientation de la caméra est appelée, estimation de pose.

L'estimation de pose peut être décomposée en deux étapes:

- Une étape de traitement d'images, visant à obtenir des mesures du mouvement de la scène ou des indices de la position d'un objet,
- Une étape d'estimation proprement dite utilisant les mesures précédentes pour déterminer la pose.

Afin de recouvrir les caractéristiques extrinsèques de la caméra on utilise des modèles de projection, le plus connus est le modèle sténopé. En ayant une connaissance au préalable sur l'environnement on peut automatiser le processus d'alignement par des algorithmes, cependant la portée des applications de RA est restreinte, d'où l'apparition de nouvelles méthodes qui ne tiennent pas compte d'une connaissance totale de l'environnement. Dans cette partie du rapport nous aborderons la modélisation du capteur (camera) et nous passerons en revue les différentes méthodes.

2. Modélisation du capteur

Un modèle de caméra représente le processus de formation des images. Il permet d'établir la relation analytique entre les coordonnées d'un point dans l'espace objet, et celle du point correspondant dans le plan image au moyen des paramètres choisis. Plus précisément, il s'agit de déterminer la matrice de projection qui transforme un point de l'espace 3D en un point du plan que forme l'image [Didier, 2005].

Ils existent une variété de modèles dans la littérature, leurs caractéristiques géométriques et leurs propriétés nous permettent de les considéré davantage pour certains types d'applications, nous pouvons citer les modèles : affine, stéréoscopique et sténopé.

Le modèle de caméra le plus fréquemment utilisé dans les systèmes de vision par ordinateur est le modèle de projection perspective (dit sténopé); L'hypothèse géométrique fondamentale de ce modèle consiste à supposer que tous les rayons qui lient un point dans l'espace avec sa projection correspondante sur le plan image concours vers un point nommé : "le centre de projection perspective"[Vigueras, 2007] (voir Figure 2. 1).

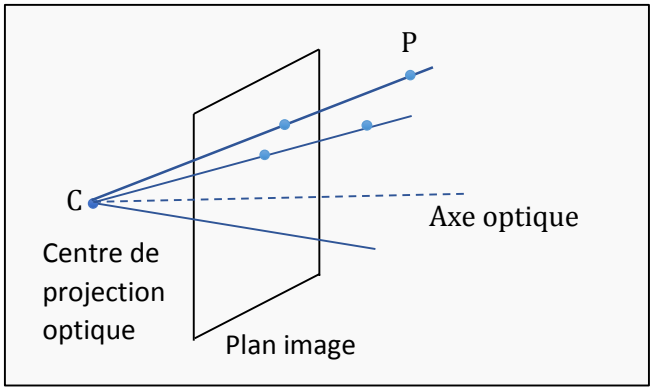


Figure 2. 1 Le centre de projection perspective « C ».

Comme indiqué dans la figure (2. 2) la notation : (1) représente une transformation entre le repère du monde réel (repère d'origine O) et le repère de la caméra (dont l'origine C est située au centre optique de la camera) cette transformation est lié aux paramètres extrinsèques (ou externes) de la camera. Les transformations, notée (2) et (3) sont liées aux paramètres intrinsèques (ou internes) de la camera et permettent de relier le repère de la camera au repère du plan rétinien (plan image).

De manière générale, les coordonnées d'un point dans l'espace P(X, Y, Z) et sa projection sur le plan image Q(u, v) sont liées par la matrice de projection (M) selon la relation [Vigueras, 2007]:

$$Q = M\,P \qquad (2.1)$$

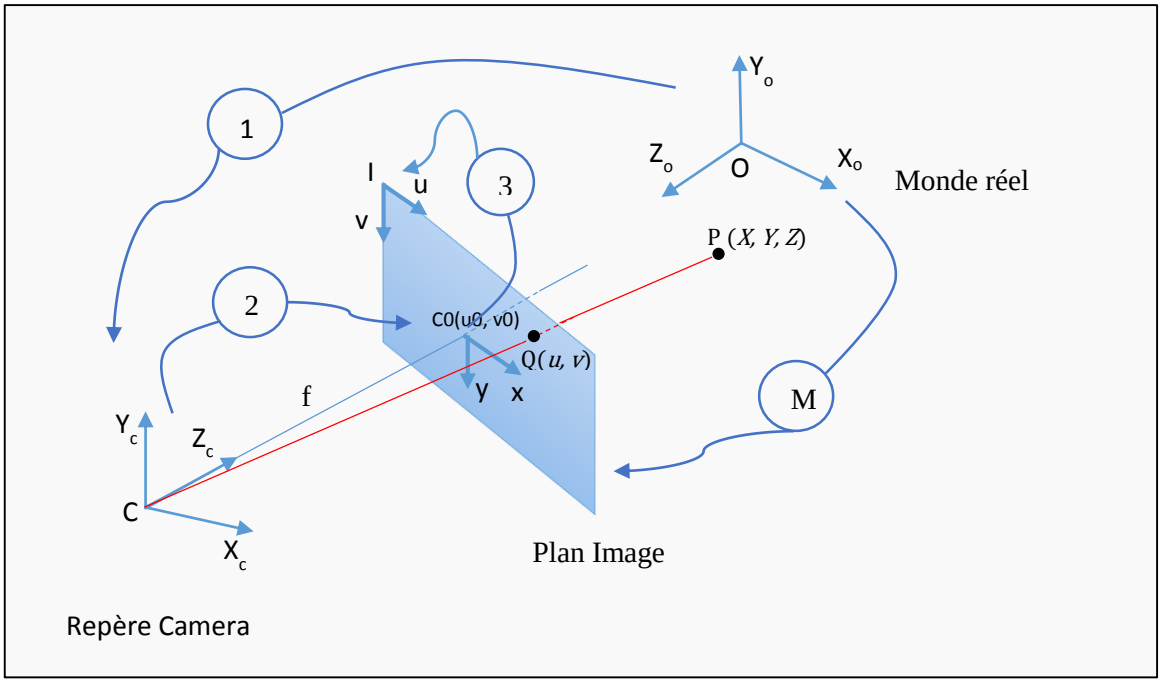


Figure 2. 2 Le modèle sténopé.

Les paramètres externes

Si le point P a pour coordonnées (X,Y,Z) dans le repère (O,X_o,Y_o,Z_o) de la scène, ses coordonnées (X_c, Y_c, Z_c) dans le repère (C, X_c, Y_c, Z_c) de la camera sont données par la relation [Toscani, 1987] :

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = \mathrm{R} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \mathrm{t} = (\mathrm{R}\ \mathrm{t}) \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{2.2}$$

Ou (R t) exprime le déplacement rigide entre les deux repères (rotation et translation). La rotation R est souvent exprimée en fonction des angles : γ, β, α autour respectivement des trois axes X_o, Y_o, Z_o (Figure 2. 3):

$$\mathrm{R} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{pmatrix} \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.3}$$

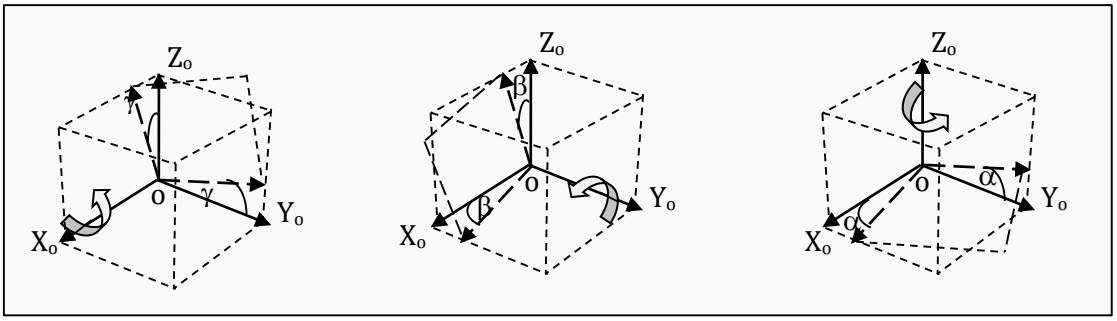


Figure 2. 3 Représentation des rotations d'angles : γ, β, α sur les axes respectif X_o,Y_o et Z_o.

Ces angles sont appelés angles d'Euler. Les paramètres de changement de repère sont donc au nombre de six : les trois angles d'Euler de R et les trois composantes du vecteur de translation t. Ces paramètres, définissent l'orientation et la position de la camera par rapport au repère de la scène, et sont appelés paramètres externes (ou paramètres extrinsèques) de la camera. [Lepetit, 2001]

Les paramètres internes

Soit (C, X_c, Y_c, Z_c) le repère 3D lié à la caméra et soit (c_0, x, y) le repère 2D du plan image (voir Figure 2. 2). Nous avons les relations suivantes :

$$\frac{f}{Z_c} = \frac{x}{X_c} = \frac{y}{Y_c} \tag{2.4}$$

Si on change les unités de mesure de l'axe des x et de l'axe des y sur le plan image (ce qui correspond à l'échantillonnage (Figure 2. 4) :

$$x \Rightarrow \frac{u}{\mathrm{k_u}}$$

$y \Rightarrow \frac{v}{k_v}$

Et on translate l'origine :

$u \longrightarrow u - u_0$

$v \longrightarrow v - v_0$

On a les relations suivantes :

$$x = \frac{u - u_0}{k_u} \quad (2.5)$$

$$y = \frac{v - v_0}{k_v} \quad (2.6)$$

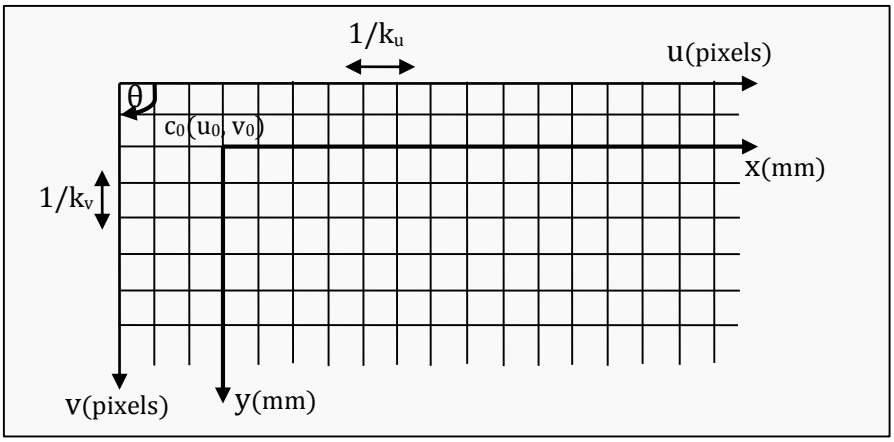


Figure 2. 4 Les paramètres internes [Lep01].

Avec :

- k_u et k_v le nombre de pixels par unité de longueur suivant chacun des axes (>0).
- u_0 et v_0 les coordonnées pixel du point principal c (intersection de l'axe optique $(C, \overrightarrow{k_c})$ avec le plan image).
- θ l'angle entre les deux axes du repère image.

Les paramètres k_u, k_v, f, u_0, v_0, θ sont les paramètres internes (ou intrinsèques) de la camera. En pratique, l'angle θ est très bien contrôlé et peut être considéré égal à $\frac{\Pi}{2}$. D'autre part, il n'est pas possible de séparer les paramètres k_u et k_v de la distance focale f [Lepetit, 2001], on pose alors :

$$\alpha_u = k_u.f \quad (2.7)$$

$$\alpha_v = k_v.f \quad (2.8)$$

Nous considérons donc le modèle simplifié à quatre paramètres αu, αv, u0 et v0. D'après les équations (2.2), (2.4), (2.5), (2.6), (2.7) et (2.8), nous avons finalement.

$$s\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}}_{A} \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = A \underbrace{(R\ t)}_{T} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.9)$$

On note "A" la matrice des paramètres internes, et "T" la matrice des paramètres externes. La matrice M = A (R t) est appelée matrice de projection perspective : elle permet d'exprimer directement la projection d'un point 3D de la scène en coordonnées pixel de l'image. Il s'agit d'une matrice 3 × 4 définie à un facteur d'échelle près, et possédant 11 paramètres indépendants. [Toscani, 1987]

$$M = A\,T = \begin{pmatrix} \alpha_v & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{pmatrix} = \begin{pmatrix} \alpha_u r_1 + u_0 r_3 & \alpha_u t_x + u_0 t_z \\ \alpha_v r_2 + v_0 r_3 & \alpha_v t_y + v_0 t_z \\ r_3 & t_z \end{pmatrix}$$

$$= \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{pmatrix} = \begin{pmatrix} m_1 & m_{14} \\ m_2 & m_{24} \\ m_3 & m_{34} \end{pmatrix} \quad (2.10)$$

Avec : « r_i » le $i^{ème}$ vecteur ligne de la matrice de rotation « R » et « m_i » le $i^{ème}$ vecteur ligne de la matrice « M » privé de la dernière coordonnée.

D'après (2.9) et (2.10) : Tout point $P_i(X_i, Y_i, Z_i)$ dans l'espace et sa projection sur le plan image $Q_i(u_i, v_i)$ sont liés par la relation:

$$s\begin{pmatrix} u_i \\ v_i \\ 1 \end{pmatrix} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{pmatrix} \begin{pmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{pmatrix} \quad (2.11)$$

3. Approche de localisation

Selon les données dont nous disposant (connaissance totale, partielle ou nulle de l'environnement), on peut classer les méthodes d'estimation de pose en deux classes: les approches avec ou sans connaissance apriori.

3.1. Approches avec connaissance apriori

Les approches avec connaissances apriori mettent en relation la scène réelle avec l'image courante de celle-ci. On pourra classer les méthodes existantes en deux catégories selon la nature des informations extraites, la première se base sur des marqueurs artificiels placés sur la scène, quand à la seconde, elle exploite des informations naturelles. Nous détaillerons dans ce qui suit chacune des deux approches.

3.1.1. Méthodes à base de cibles codées (marqueurs artificiels)

Plusieurs systèmes d'identification de cibles codées ont été présentés dans la littérature. L'idée est de placer des marqueurs visibles différents par leurs couleurs ou leurs formes dans l'environnement réel. En se basant sur le fait que les marqueurs sont connus apriori, 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.

Parmi les librairies de marqueurs existantes nous pouvons citer Cantag [Rice et al., 2006] qui est une librairie OpenSource, il y'a aussi la librairie d'Intersense [Naimark et al., 2002] qui utilise des marqueurs circulaires et Cybercode [Rekimoto et al., 2000] basé sur des cibles codées. Toutefois les marqueurs planaires ont été démocratisés avec l'avènement de la bibliothèque ARToolkit [Kato et al., 1999] ,ou 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'amélioration tel que Artag [Fiala, 2005] et ArtoolkitPlus [Wagner et al., 2007]. Il y'a aussi le code QR ouvert au public, ce dernier peut contenir différentes informations tel que du texte ou un lien web, proposé dans un premier temps pour suivre le chemin des pièces détachées dans les usines puis son utilisation s'est étendu à plusieurs domaines dont la RA [Kan et al., 2009].

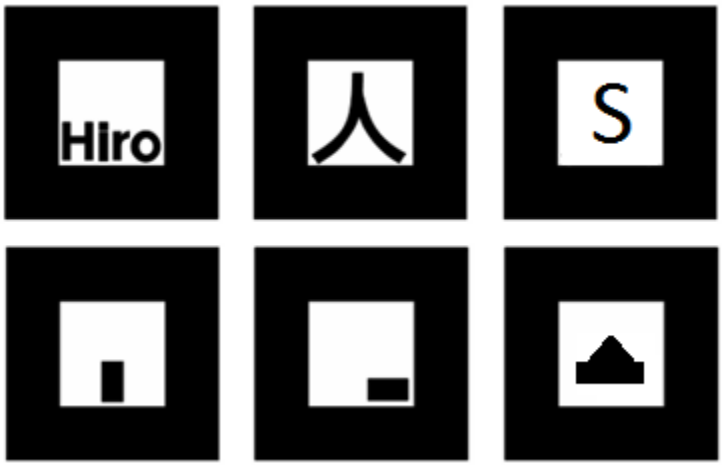


Figure 2. 5 Modèles de marqueurs d'ARToolKit.

Bien que les approches basées marqueurs donnent des résultats précis, il faut noter cependant que la connaissance exacte des marqueurs est un inconvénient majeur pour les applications dans un environnement vaste.

3.1.2. Méthodes à base de cibles naturelles (marqueurs naturels)

Les marqueurs artificiels tel que présentés précédemment suscitent peut d'intérêt lorsqu'on peut utiliser des marqueurs naturels en exploitant les caractéristiques naturelles existantes dans la scène réelle telle que des coins, des contours, des segments de droites, etc. Ces approches utilisent des modèles qui constituent une connaissance apriori de l'environnement, les données 2D extraites de l'image de la scène sont mises en correspondance avec le modèle.

Les méthodes les plus utilisées se basent sur la détection et description des points d'intérêts, car elles sont plus robustes, et sont applicables à tous types d'environnements. L'étape de la description sert à coder le point et la région qui l'entoure (le patch) sous forme d'un vecteur descriptif, cette description nous permet de comparer les points extraits d'une image avec ceux du modèle afin de déterminer la position de l'objet correspondant au modèle sur l'image en question (Voir Figure 2. 6).

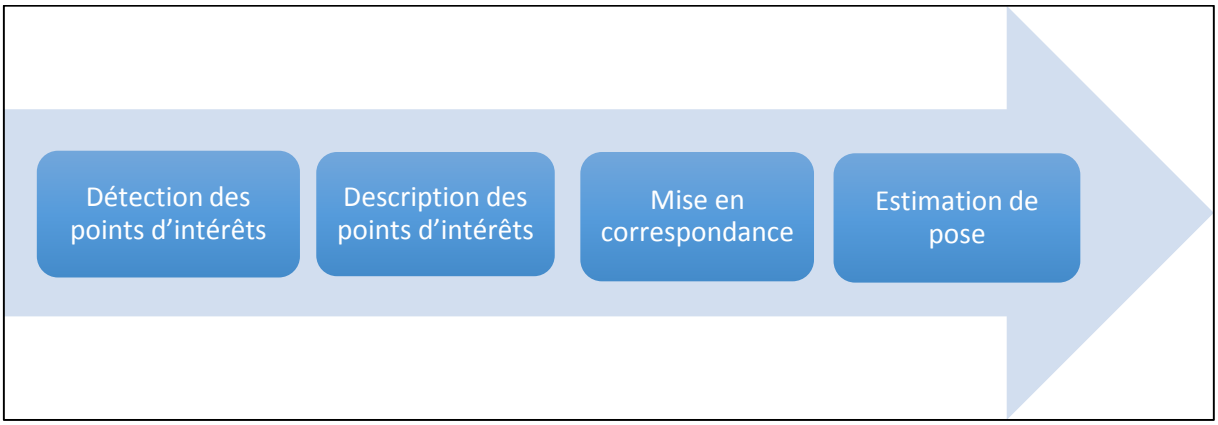


Figure 2. 6 Processus d'une technique de reconnaissance d'objet basée sur les points d'intérêt.

Nous allons présenter brièvement dans ce qui suit, les techniques de détection, de description et de calcul de la pose.

A. Techniques de détection de points d'intérêts

La détection de points d'intérêts (ou coins) est, au même titre que la détection de contours, une étape préliminaire pour de nombreux processus de vision par ordinateur. Les points d'intérêts, dans une image, correspondent à des doubles discontinuités de la fonction d'intensités. Ce sont par exemple : les coins, les jonctions en T ou les points de fortes variations de texture.

De nombreux détecteurs de points sont présents dans la littérature, nous pouvons citer notamment le détecteur de Moravec [Moravec, 1980] et le détecteur d'Harris [Harris et al., 1988] qui se basent tous les deux sur l'autocorrélation. Il y'a aussi, Kitchen et Rosenfeld qui ont proposé dans [Kitchen et al., 1982] un detecteur qui se base sur une méthode de topologie. Smith et Brady [Smith et al., 1997] ont proposé SUSAN (Smallest Univalue Segment Assimilating Nucleus) un détecteur de coins et de contours en même temps qui se base davantage sur la structure de l'image que sur ses propriétés mathématiques. Trajkovic et Hedley [Trajkovic et al., 1998] proposent un détecteur de coins en adoptant la même définition que celle d'Harris et de Moravec « un coin et un point ou le changement d'intensité est élevé dans toutes les directions ». Mokhtarian et Suomela [Mokhtarian et al., 1998] proposent le CSS (Curvature Scale Space) un détecteur de coins qui se base sur la recherche en locale des maxima des courbures des contours détectés sur l'image. FAST (Features from Accelerated Segment Test) [Rosten et al., 2006] est un des algorithmes de détection de coins récemment développé. Il est beaucoup plus rapide que les autres détecteurs de coin existants et présente une très bonne répétabilité sous de grands changements et pour différents types de caractéristiques.

Les algorithmes de détection de points d'intérêt se focalisent en général sur des points particuliers des contours, sélectionnés selon un critère précis. Ainsi, les coins (*corners*) sont les points de l'image où le contour change brutalement de direction, comme par exemple aux quatre sommets d'un rectangle. Il s'agit de points particulièrement stables, et donc intéressants pour la répétabilité.

B. Techniques de description de point d'intérêts

Dans la littérature, la majorité des travaux se focalisent sur deux points essentiels : la description des points d'intérêt, et la mesure de similarité. Il est à noter également, que les algorithmes proposés avant la naissance des descripteurs binaires, se sont basés sur le calcul à base de virgule flottante (Floating point).

Dans ce qui suit, nous allons regrouper les travaux en deux grandes familles, les techniques à base de virgule flottante, et les descripteurs à base binaire.

B.1. Techniques de vision 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 première étape est 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-clé. 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.

Une Version améliorée de cette méthode est proposé en 2004 [Lowe 2004] afin d'augmenter sa robustesse à l'invariance, mais l'inconvénient majeur de cette méthode est le temps de calcul qu'elle prend.

Se basant sur les propriétés de SIFT, Bay et al [Bay et al., 2006] ont proposé en 2006 la méthode SURF (pour 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, elle donne des résultats moins robuste que SIFT, Plusieurs variantes de SURF ont été proposées, à savoir SURF avec flot optique [Noguchi et al., 2012], SURF avec Kalman [Ta et al., 2009]

Une amélioration de SURF est proposée par l'équipe IRVA du CDTA [Hamidia et al., 2013], cette dernière sert à minimiser la zone de recherche par SURF. L'idée de base est d'appliquer autour de l'objet détecté une région d'intérêt, afin de minimiser la zone de recherche pour la prochaine frame. Cette méthode a permis un gain en temps de calcul, cependant, elle génère des mauvais résultats lors de mouvements rapides.

L'algorithme de suivi proposé par Benhimane et Malis [Benhimane et al., 2004] est basé sur la minimisation de la SSD (la somme des carrés des différences, en anglais : Sum of squared differences) entre un modèle donné et l'image courante en appliquant l'algorithme ESM (Efficient Second order Minimisation).

Comport et al [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.

Lepetit et al [Lepetit et al., 2006], ont utilisé une approche robuste basée sur les arbres de classification aléatoires (Randomized Trees). Cette méthode permet dans un premier temps de sélectionner lors de la phase de l'apprentissage les points d'intérêt les plus significatifs (reconnaissable), puis de les mettre en correspondance. Cette méthode est connue par sa rapidité dans la recherche.

Wagner Daniel et al, de l'Université de Graz [Wagner et al., 2008] ont proposé des techniques pour le suivi en temps réel sur les téléphones mobiles. L'idée de base est d'hybrider le descripteur SIFT avec la méthode de classification FERNS, en rajoutant un algorithme de tracking basé sur la technique de Template-matching.

Herling, Jan et al [Herling et al., 2012], ont proposé une approche qui se base sur la recherche aléatoire des points d'intérêt en utilisan6t SIFT comme descripteur, avec initialisation de pose, cette technique a permis un calcul plus ou moins rapide par rapport à SIFT.

Les techniques présentées ci-dessus, utilisent un codage basé sur la virgule flottante pour le calcul de vecteurs descriptifs tel que l'histogramme des gradients orientés (HOG) comme dans SIFT et SURF, ou pour le calcul de la distance tel que l'ESM. Cependant, ces techniques restent toujours limitées en termes de temps de calcul et de robustesse.

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

B.2. Descripteurs binaires

En général, les descripteurs binaires sont composés de trois parties : La modélisation du pattern (le patch), la compensation de l'orientation et la modélisation des paires. Considérons un petit patch centré par un point-clé. Nous souhaitons le décrire comme une chaîne binaire.

La première étape est alors de modéliser le pattern autour du point-clé, par exemple des points répartis sur un ensemble de cercles concentriques, et ceci comme dans le cas de BRISK [Leutenegger et al., 2011] et FREAK [Alahi et al., 2012] (voir Figure 2. 7).

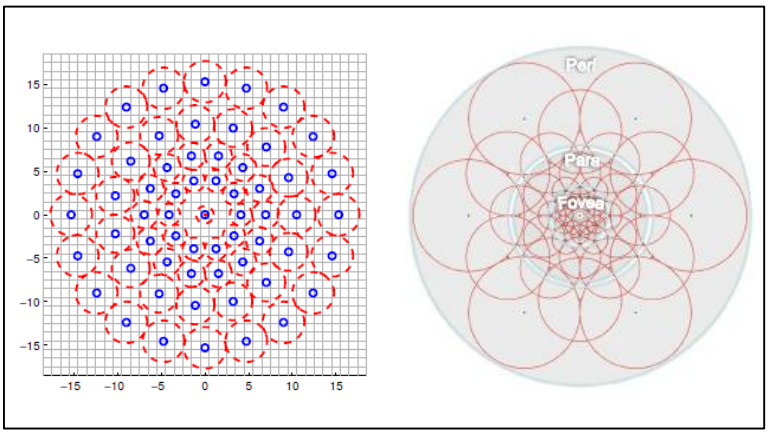


Figure 2. 7 Modélisation du pattern dans BRISK et FREAK

Ensuite, on choisit N paires de points (512 paires de points dans le cas de BRISK) sur ce modèle d'échantillonnage. On compare par la suite, sur toutes les paires des points la valeur d'intensité du premier point de la paire avec la valeur de l'intensité du deuxième point de la paire. Si la première valeur est plus grande que la seconde, on met '1' dans la

chaîne, sinon on met '0'. Après avoir effectué le test pour les N paires, nous aurons une chaîne de N caractères, composée de '1 'et '0' qui codent les informations locales autour du point-clé. Chaque descripteur binaire a sa propre méthode de modélisation du patch, sa propre méthode de calcul de l'orientation et son propre ensemble de paires d'échantillonnage.

a. Le descripteur BRIEF

Présenté en 2010 [Calonder et al., 2012], BRIEF a été le premier descripteur binaire publié. Il ne dispose pas 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.

BRIEF ne prend que les informations à emplacement unique pixels pour construire le descripteur. Alors, les paires peuvent être choisies à tout moment d'une manière aléatoire sur le patch SxS. Pour construire un descripteur BRIEF de longueur n, nous devons déterminer n paires (Xi, Yi). Notons X et Y les vecteurs de points Xi et Yi, respectivement.

Ainsi, afin de rendre le patch choisi moins sensible au bruit on applique un filtre gaussien. Dans [Calonder et al., 2012], les auteurs considèrent cinq (5) méthodes pour déterminer les vecteurs X et Y:

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 $0{,}04 * S^2$, et les Yi sont prélevés en utilisant une distribution gaussienne par rapport à Xi et 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.

b. Descripteur ORB

Le descripteur ORB [Rublee et al., 2012] est un peu similaire à BRIEF. Il ne dispose pas d'un modèle d'échantillonnage élaboré tel que BRISK ou FREAK. Cependant, il y'a deux différences principales entre ORB et BREF:

1. ORB utilise un mécanisme de compensation de l'orientation, qui le rend invariant à la rotation.

2. Les paires de points optimales sont apprises pour ORB, alors que dans BRIEF elles sont choisies au hasard.

Compensation de l'orientation :

ORB utilise la mesure du centre de gravité de l'intensité afin de calculer la rotation du patch. Tout d'abord, les moments d'un patch sont définis comme suit (Equ. 2.12):

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y) \tag{2.12}$$

En utilisant cette équation, on peut calculer le centre de gravité comme suit (Equ. 2.13):

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right) \tag{2.13}$$

Ensuite, on construit un vecteur du centre du patch "O", vers le centre de gravité "C". L'orientation du patch alors, est donnée par (Equ. 2.14):

$$\theta = atan^2(m_{01}, m_{10}) \tag{2.14}$$

Une fois l'orientation est calculée, le patch est tourné à une rotation canonique, puis on calcule le descripteur.

c. Descripteur BRISK

Le descripteur BRISK [Leutenegger et al., 2011] est différent des descripteurs dont nous avons parlé plus tôt (BRIEF et ORB), par son modèle de paires de points. Ce dernier est composé de cercles concentriques (Figure 2. 8). 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.

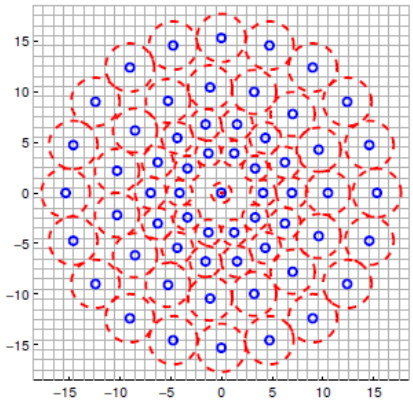


Figure 2. 8 Model du pattern de BRISK.

Apprentissage des paires (Paires longues et paires courtes)

En utilisant cette technique de modélisation du patch (patch sampling), nous distinguons entre les paires, de courtes, et de longues paires. Les Paires courtes sont des paires de points avec une distance qui est inférieure à un certain seuil d_max, et les longues paires sont des paires de points avec une distance qui est supérieure à un certain seuil d_min, où d_min> d_max, donc il n'y a pas de paires courtes qui soient aussi longues.

Les paires à longues distances sont utilisées pour déterminer l'orientation et les paires à courtes distances sont utilisées pour les comparaisons d'intensité qui construisent le descripteur, comme dans BRIEF et ORB.

Les figures suivantes (Figure 2.9) illustrent la configuration des paires de points à courtes distances, chaque ligne rouge représente une paire. Chaque figure montre 100 paires de points.

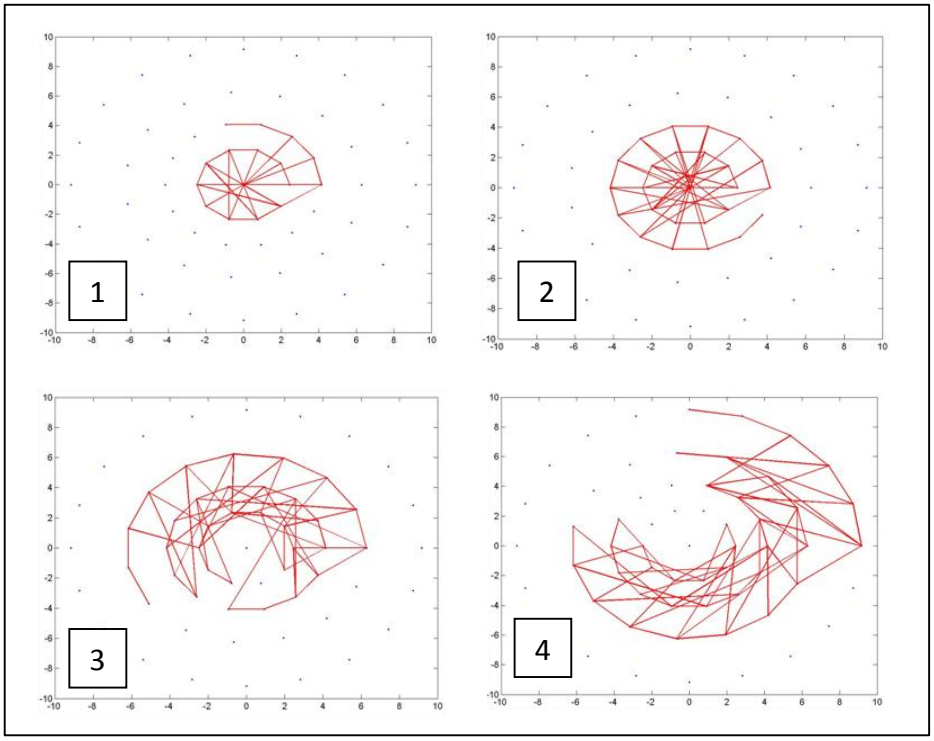


Figure 2. 9 Model du pattern de BRISK (Paires à courtes distances).

Calcul de l'orientation

Pour le calcul de l'orientation du point-clé, BRISK utilise des gradients locaux entre les paires de points qui ont une longue distance. Les gradients locaux sont définis par (Equ. 2.15):

$$g(p_i, p_j) = (p_j - p_i).\frac{I(p_j, \sigma_j) - I(p_i, \sigma_i)}{||p_j - p_i||^2} \qquad (2.15)$$

Où $g(p_i, p_j)$ est le gradient local entre la paire des points (p_i, p_j), I est l'intensité lissée par une gaussienne dans le point de prélèvement correspondant à l'écart-type approprié (voir la figure 2.9 ci-dessus de modèle d'échantillonnage BRISK).

Pour calculer l'orientation, on somme l'ensemble des gradients locaux entre toutes les paires de long distances et on prend $\arctan(g_y / g_x)$ (l'arc tangente de la composante y du gradient divisé par la composante x du gradient). Cela donne l'angle du point-clé.

Maintenant, il suffit de faire tourner le patch par cet angle pour aider le descripteur à devenir moins variant à la rotation. On notera que, BRISK Utilise seulement les paires à longues distances pour le calcul de l'orientation.

Construction du descripteur

Comme avec tous les descripteurs binaires, la construction du descripteur est réalisée en effectuant des comparaisons de l'intensité. BRISK prend l'ensemble des couples courts, tourne les paires par l'orientation calculé auparavant, et fait des comparaisons en appliquant la formule suivante (Equ. 2.16):

$$b = \begin{cases} 1, & Si\ I\left(p_j^{\propto}, \sigma_j\right) > I(p_i^{\propto}, \sigma_i) \\ 0, & Si\ non \end{cases} \qquad (2.16)$$

Ce qui signifie que pour chaque paire courte, il prend l'intensité lissée, des points d'échantillonnage et il vérifie si l'intensité du premier point de la paire est plus grande que celle du second point. Si c'est le cas, alors il écrit 1 dans le bit correspondant du descripteur, et 0 sinon. Rappelons que BRISK utilise uniquement les courtes paires pour la construction du descripteur.

Ainsi, la distance entre deux descripteurs est définie comme étant le nombre de bits différents des deux descripteurs, et ceci peut être facilement calculé comme la somme, de l'opérateur XOR entre les deux descripteurs.

La figure ci-dessous (Figure 2.10) montre un exemple d'utilisation de BRISK pour correspondre entre images avec le changement de point de vue. Les cercles rouges sont les points clés détectés. Les lignes vertes sont des correspondances valides.

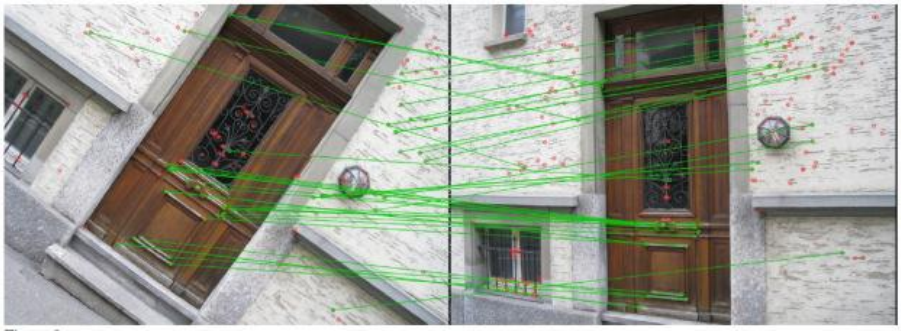


Figure 2. 10 Exemple d'application du descripteur BRISK.

d. Descripteur FREAK

En résumé, FREAK [Alahi et al., 2012] est similaire à BRISK en ayant un modèle d'échantillonnage, et également similaire à ORB en utilisant des techniques d'apprentissage automatique pour apprendre l'ensemble optimal des paires d'échantillonnage. FREAK possède également un mécanisme d'orientation qui est similaire à celui de BRISK.

De nombreux modèles d'échantillonnage sont proposés pour comparer les intensités des pixels. Comme présenté précédemment BRIEF utilise des paires aléatoires, ORB utilise des paires apprises et BRISK utilise un modèle circulaire où les points sont équidistants sur des cercles concentriques, semblables à DAISY [Tola et al., 2010].

FREAK propose d'utiliser la grille d'échantillonnage de la rétine, d'où vient son nom (Fast Retina Key-point), qui est également circulaire avec une densité plus élevée de points près du centre. La densité de points baisse de façon exponentielle pour les cercles extérieurs comme on peut le voir dans la figure ci-dessous (Figure 2.11). Chaque point sélectionné est lissé avec un noyau gaussien où le rayon du cercle illustre la taille de l'écart-type du noyau.

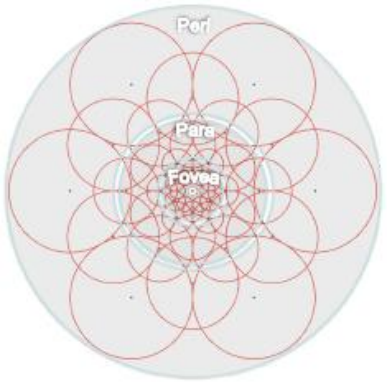
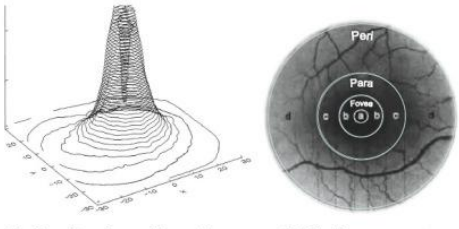


Figure 2. 11 Modélisation du pattern de FREAK.

La figure suivante (Figure 2.12), montre que la grille d'échantillonnage suggérée correspond à la distribution des champs récepteurs sur la rétine:



(a) Density of ganglion cells over the retina (b) Retina areas

Figure 2. 12 Distribution des champs récepteurs sur la rétine.

Choix des paires

Avec quelques dizaines de points d'échantillonnage, des milliers de paires d'échantillonnage peuvent être envisagées. Cependant, un grand nombre de paires peuvent ne pas être utile pour décrire de manière efficace un patch. Une stratégie possible, est de suivre l'approche de BRISK [Leutenegger et al., 2011] et sélectionner les paires en fonction de leur distance spatiale. Cependant, les paires sélectionnées peuvent être fortement corrélés et non discriminante. Par conséquent, FREAK suit l'approche d'ORB [Rublee et al., 2011] et cherche à apprendre les paires en maximisant la variance des paires et en prenant les paires qui ne sont pas corrélés.

Intéressant de noter, que l'approche FREAK, est inspirée de la compréhension du modèle de la rétine humaine. Les premières paires qui sont sélectionnées sont comparées principalement avec les points d'échantillonnage dans les cercles extérieurs de la structure où les dernières paires se comparent principalement aux points dans les cercles internes du patch. Ceci est similaire à la manière dont fonctionne l'œil humain, car il utilise d'abord les champs récepteurs péri-fovéolaires pour estimer l'emplacement d'un objet. Ensuite, la validation est effectuée avec les champs récepteurs les plus densément répartis dans la région de la fovéa [Alahi et al., 2012].

Les paires d'échantillonnage sont illustrées dans la figure suivante (Figure 2.13), où chaque figure contient 128 paires (de gauche à droite, de haut en bas):

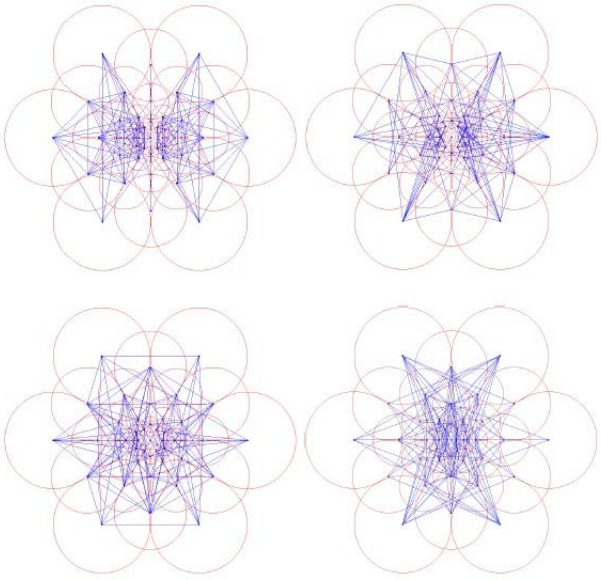


Figure 2. 13 Différents types de choix des paires de points pour le descripteur FREAK.

FREAKS prend avantage de cette structure pour accélérer encore la mise en correspondance en utilisant une approche en cascade : lors de l'appariement de deux patchs, nous comparons tout d'abord que les 128 premiers bits. Si la distance est inférieure à un seuil, on continue encore la comparaison avec les 128 bits suivants. Par

conséquent, une cascade de comparaison est effectuée en accélérant encore la mise en correspondance en tant que plus de 90% des candidats sont mis au rebut avec les 128 premiers bits du descripteur.

Calcul de l'orientation

Pour compenser les variations de rotation, FREAK mesure l'orientation du point-clé et tourne les paires d'échantillonnage par l'angle mesuré. Le mécanisme de FREAK qui permet de mesurer l'orientation est similaire à celui de BRISK, seulement au lieu d'utiliser des paires de longues distances, FREAK utilise un ensemble prédéfini de 45 paires symétriques d'échantillonnage (Figure 2.14):

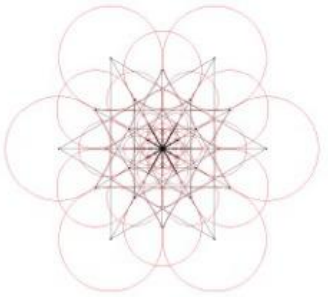


Figure 2. 14 Les 45 paires présélectionnées pour la mesure de l'orientation.

e. MOBIL, un descripteur binaire basé sur les moments

Malgré les avantages que présentent les descripteurs binaires, ils souffrent toujours de certaines anomalies. En effet, l'utilisation de la différence d'intensité pour les tests binaires donne une description insuffisante du patch.

Dans le cas général, pour qu'un descripteur soit robuste, il doit garantir que :

- Chaque point d'intérêt ait un seul vecteur descriptif dans l'ensemble de description.
- Et de l'autre côté, chaque vecteur descriptif doit être une description d'un point d'intérêt unique dans l'ensemble des points.

Formellement, soient K un ensemble de points d'intérêt, et k un élément de K, et V l'ensemble des vecteurs descriptifs.

On définit *d (voir équation (2.17))*, la fonction qui génère pour chaque point k de K sa description *d(k) dans V*.

$$\begin{array}{rcl} d : K & \rightarrow & V \\ k & \rightarrow & d(k) \end{array} \qquad (2.17)$$

On dit qu'un descripteur est robuste, quand la fonction *d* est une *application bijective*. Ce qui se traduit par (2.18) et (2.19) :

$$\forall\, k, k' \in K \left((\, k = k') \Rightarrow \left(d(k) = d(k') \right) \right) \qquad (2.18)$$

et

$$\forall\, k, k' \in K \left((\, k \neq k') \Rightarrow \left(d(k) \neq d(k') \right) \right) \tag{2.19}$$

La figure 2.15 présente deux patches différents avec une description binaires basée sur la différence d'intensité. Nous pouvons constater que l'utilisation de l'intensité pour les tests binaires va générer la même description pour les deux patchs, de ce fait l'équation (2.19) n'est pas vérifiée.

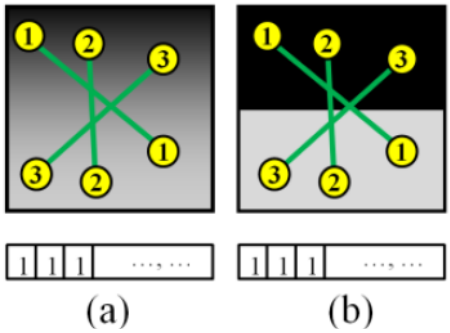


Figure 2. 15 Deux patchs différents donnent la même description quand on se base seulement sur l'intensité pour les tests binaires.

Afin de remédier à ce genre de problèmes, le descripteur binaire nommé « MOBIL : MOments based BInary differences for Local description » [Bellarbi et al. 2014] se base sur les propriétés géométriques et statistiques de l'image (les moments géométriques) pour calculer la différence binaires.

Choix des points d'intérêt

Les techniques de détection des points d'intérêt sont nombreuses. Les plus souvent utilisées, sont celles basées sur la détection des coins (Edge detection) car elles présentent une robustesse aux différentes transformations [Tuytelaars et al., 2008]. Parmi les détecteurs de coins nous avons FAST [Rosten et al., 2006] qui est connu par sa rapidité, et HARRIS [Harris et al., 1988] qui est connu par sa robustesse. MOBIL s'est inspiré de la technique d'ORB. De ce fait, l'auteur utilise FAST filtré par Harris pour avoir des points d'intérêt robustes.

Vu que ni FAST, ni Harris ne sont invariants au changement d'échelle, MOBIL applique pour chaque image capturée une pyramide à multi-échelle (Multi-scale pyramid), afin de reproduire plusieurs tailles de l'image, puis on applique l'algorithme de détection pour chaque image.

La description

L'idée de base du descripteur MOBIL, est de diviser chaque patch en 4 x 4 sous-régions égales, puis on calcule pour chacune les moments fondamentaux (M_{00}, M_{01}, M_{10}, M_{02}, M_{20}) en appliquant la formule suivante (Equ 2.20) :

$$m_{pq} = \sum_{y=0}^{M-1} \sum_{x=0}^{N-1} x^p y^q I(x,y) \tag{2.20}$$

Avec :

- M_{00} : est la somme des valeurs des pixels dans l'image.
- M_{10} : est le centre de gravité de la région par rapport à l'axe X.
- M_{01} : est le centre de gravité de la région par rapport à l'axe Y.
- M_{20} : est la distribution des pixels par rapport à l'axe X.
- M_{02} : est la distribution des pixels par rapport à l'axe Y.

Une fois les moments calculés, on applique des tests binaires entre les moments de chaque deux sous-régions adjacentes dans le patch (5 * 51 = 255 tests binaires), on aura à la fin un vecteur descriptif binaire de taille 255 bits (voir Figure 2.16).

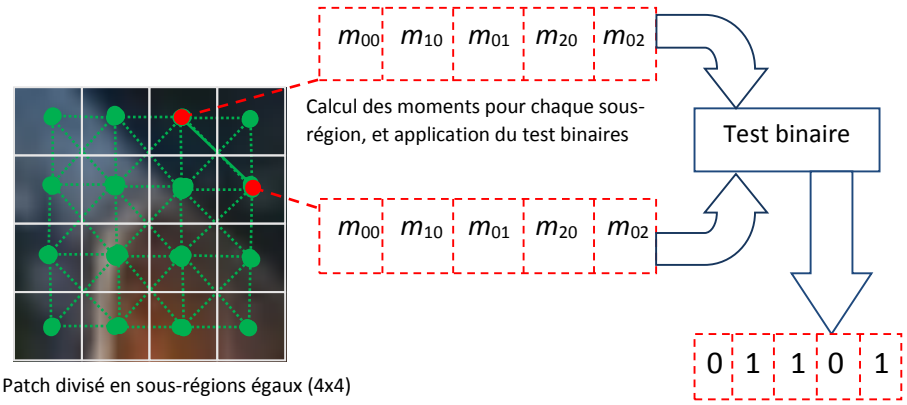


Figure 2. 16 Principe de fonctionnement du descripteur MOBIL.

MOBIL est également invariant à la rotation, il estime une orientation dominante du patch et aligne le patch à cette orientation avant le calcul de son descripteur.

Plusieurs approches d'estimation de l'orientation ont été proposées dans la littérature. Celle appliquée par MOBIL est la méthode du centre de gravité introduit par [Rublee et al., 2012] pour sa bonne performance et efficacité. (Voir la partie : compensation de la rotation, dans la section (**B.2. (b) ORB**).

Cependant, après certains tests sur cette version de MOBIL, nous avons remarqué que le descripteur souffre de certains problèmes liés à la distinction entre les patches des images, ce qui peut générer des erreurs de reconnaissance.

Afin de remédier à ces faiblesses, nous avons proposé des améliorations du descripteur MOBIL que nous présentons dans ce qui suit.

f. Améliorations du descripteur MOBIL.

La première version du descripteur MOBIL proposée dans [Bellarbi et all, 2014] a introduit la comparaison entre des moments géométriques calculés pour les sous-régions du patch afin de générer une description binaires de ce dernier. Cependant, après une série de tests sur des images différentes, nous avons constaté que le nombre réduit de tests binaires effectués cause une perte d'informations lors de la description.

MOBIL_2B, MOBIL avec deux bits

Dans cette version améliorée de MOBIL, qui est appelée MOBIL_2B [Bellarbi et all, 2015], l'amélioration principale apportée est d'affecter deux (2) bits pour chaque test binaire au lieu d'un seul bit, et ceci afin d'améliorer la précision de la description.

Étant donné deux patches extraits de deux images distinctes, nous voulons calculer les moments géométriques d'ordre 1 (M_{10}) pour les sous-régions des patches afin de faire la comparaison binaire. Nous pouvons constater dans la figure ci-dessous que les deux patches sont visuellement différents, en revanche les tests binaires effectués entre les moments des sous-régions donnent le même résultat.

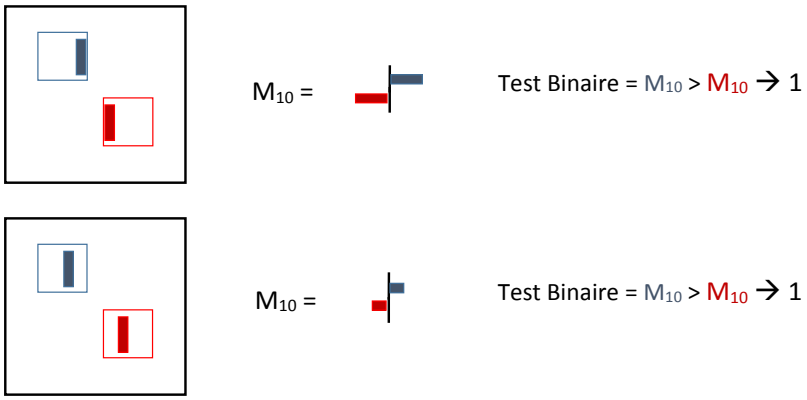


Figure 2. 17 Deux patches différents génèrent la même description binaire.

Afin de remédier à ce problème, nous avons rajouté un seuil t de la différence entre les moments, ceci génère quatre possibilités au lieu de deux, et donne plus de précision dans la description, donc le résultat du test sera affecté à 2 bits.

Du point de vue mathématique, soit (m_{pq}, m'_{pq}) 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 (2.21) :

$$v = \begin{cases} 00 & if \quad m_{pq} > m'_{pq} + t \\ 01 & if \quad m'_{pq} < m_{pq} < m'_{pq} + t \\ 10 & if \quad m'_{pq} > m_{pq} > m'_{pq} - t \\ 11 & if \quad m_{pq} < m'_{pq} - t \end{cases} \tag{2.21}$$

Nous pouvons résumer le principe de fonctionnement de ce descripteur par le schéma suivant :

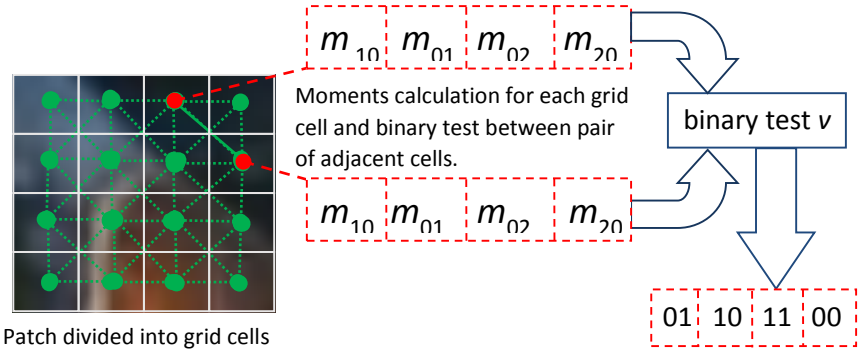


Figure 2. 18 Principe de fonctionnement du descripteur MOBIL_2B

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 après les avoir divisé par M_{00}, nous avons choisi le seuil comme un demi de la taille de la sous-région D, donc la valeur *t* est calculée comme suit :

$$t = M_{00} \times D / 2 \tag{2.22}$$

- 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 un quart de la taille D de la sous-région, donc la valeur *t* est calculée comme suit :

$$t = M_{00} \times D^2 / 4 \tag{2.23}$$

Vu que cette version du descripteur MOBIL_2B est l'amélioration de la première version de MOBIL, nous avons laissé les mêmes propriétés concernant les points d'intérêt (nous utilisons les points d'intérêt d'ORB [Rublee et al. 2012]) avec application de la technique de pyramide à multi-échelles afin d'avoir l'invariance à l'échelle.

Ainsi, pour l'invariance à la rotation, nous avons appliqué la méthode du centre de gravité utilisé dans la première version du MOBIL.

Implémentation et tests

Nous avons implémenté le descripteur MOBIL_2B en C# sous l'environnement Visual Studio de Microsoft. Nous avons effectué des tests sur la base de données des images [Mikolajczyk et al., 2005]. Cette base contient six types de transformations (Changement de luminosité, changement de point de vue, changement de la rotation et de l'échelle, compression Jpeg et le flou). Chaque type contient six images avec différents degrés pour la même transformation (Figure 2.19).

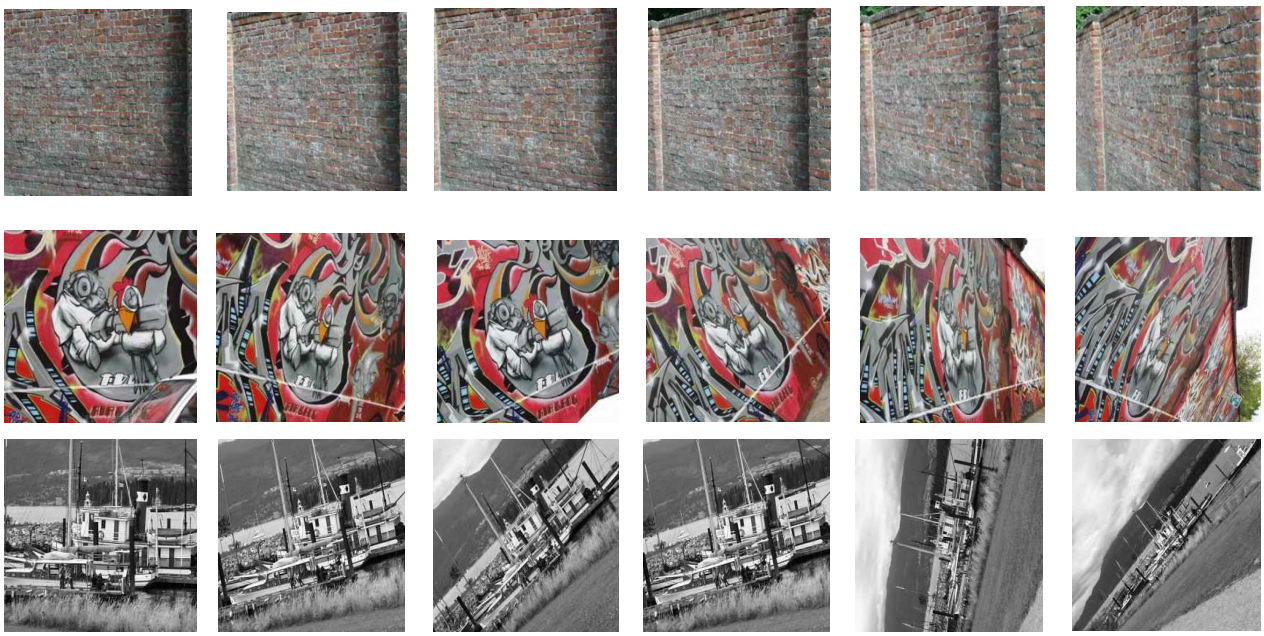


Figure 2. 19 Différents types de transformations dans la base de données de Mikolajczyk.

Nous avons tout d'abord testé notre descripteur MOBIL_2B avec les transformations affines (rotation, translation, et changement d'échelle) en utilisant les images « Bark » et « Boat » de la base de données de Mikolajczyk. Les résultats obtenus ont montré l'invariance du descripteur proposé devant ces différents types de transformations (voir figure 2.20).

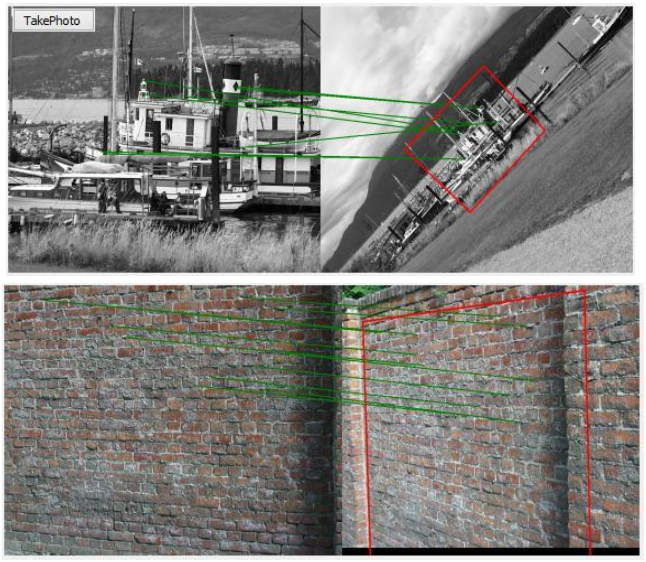


Figure 2. 20 Différents types de transformations dans la base de données de Mikolajczyk.

Nous avons également comparé la robustesse du descripteur MOBIL_2B avec la première version de MOBIL et avec les descripteurs les plus performants, en utilisant la base de données de Mikolajczyk. Les résultats obtenus sont présentés dans les graphes de la figure ci-dessous. Selon les graphes obtenus, nous concluons que l'introduction du test à 2 bits a amélioré la précision de la description par rapport à la première version de MOBIL.

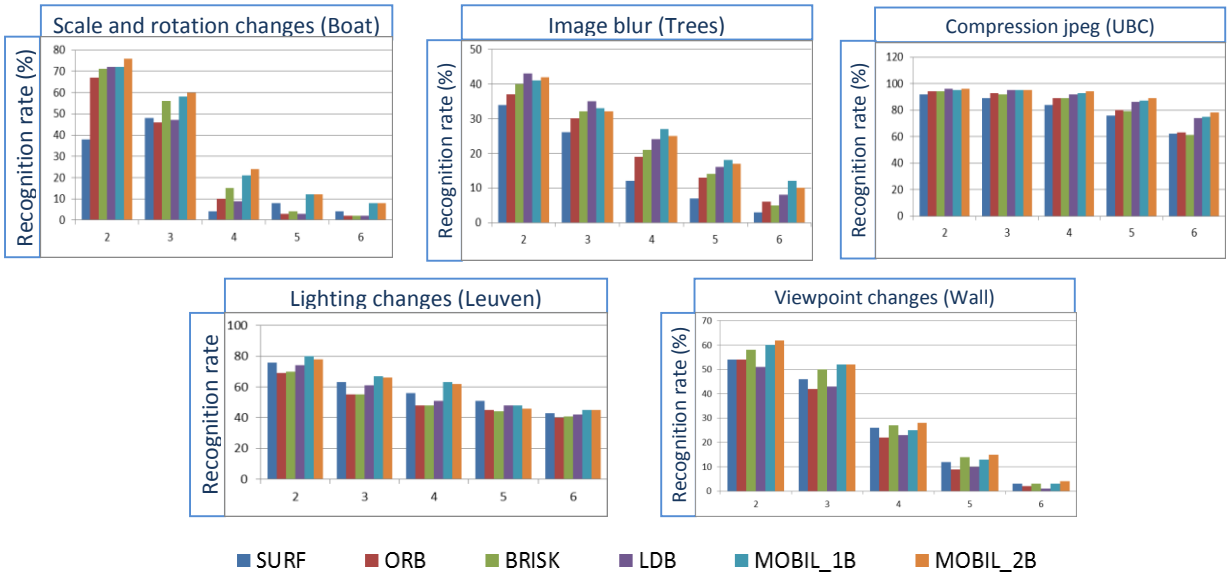


Figure 2. 21 Comparaison de MOBIL_2B avec MOBIL, Freak, ORB, BRISK, et SURF

Cependant, cette amélioration consomme plus de temps de calcul par rapport à la première version (0.131ms/patch pour MOBIL_2B vs 0.127 ms/patch pour MOBIL). (Voir Tableau 1).

Tableau 1 Temps de description pour les descripteurs testés et le descripteur MOBIL_2B

Descripteurs	Temps par description (ms/patch)
SURF	1.488
BRISK	0.062
ORB	0.146
LDB	0.139
MOBIL_1B	0.127
MOBIL_2B	0.131

3.2. Approches sans connaissance apriori

Lorsqu'on travaille sur un environnement à grande échelle, il est parfois quasi impossible d'avoir une vue globale de la scène, les approches dites sans connaissance apriori supposent une connaissance partielle de l'environnement telle que les techniques appelées SFM (Structure From Motion) [Nistér, 2005] et les SLAM (Simultaneous Localization And Mapping) [Mouragnon et al., 2006] [Mei et al., 2007] utilisées en robotique mobile.

Ces techniques ont l'avantage d'estimer la pose de la caméra et de reconstruire un modèle partiel de l'environnement. De telles approches permettent d'envisager d'utiliser des modèles 3D partiels de l'environnement et de les enrichir en ligne en reconstruisant les parties non modélisée. Toutes fois ces techniques restent gourmandes en temps de calcule et manquent de précision lorsqu'on utilise uniquement les images acquise de la camera, ainsi, elles sont en générale utilisées dans des systèmes multi capteurs, afin de bénéficier d'informations supplémentaires pour le mapping, qui permettent de gagner en précision et en temps de traitement.

Parmi les techniques existantes se basant sur les SLAM nous pouvons citer EKF pour (Extended Kalman Filter) [Chekhlov et al., 2007]. En utilisant les SLAM, l'auteur souhaite atteindre l'interactivité en temps réel et maintenir une estimation rigoureuse de l'incertitude du système, pour avoir une bonne estimation du mouvement de la camera.

Nous pouvons également citer la méthode RLS pour (Recurssive Least square method) [Imran et al., 2011], qui permet de calcule d'homographie robuste et récursive : Hi = f(Hi-1). Cette technique présente une capacité à traiter les mesures périodiques et erronées, et à fournir la meilleure solution possible. En outre, RLS a une certaine capacité à fournir des résultats fiables pour l'estimation des paramètres variables dans le temps. Souvent utilisée dans le contexte du traitement d'image en temps réel tel que la construction

d'images panoramiques « Image stiching » où la transformation optimale entre les plates-formes mobiles est susceptible de changer en raison du mouvement.

4. Estimation de la pose

Nous abordons à présent l'estimation proprement dite utilisant les mesures précédentes pour déterminer la pose de la camera. Déterminer le point de vue de la camera, revient à la calibrer.

On entend par calibrage de caméra : déterminer de manière analytique la fonction qui associe à un point de l'espace tridimensionnel 3D sa projection dans l'image donnée par la caméra. Cette projection n'est pas une transformation quelconque, elle est directement liée au modèle de caméra choisi.

Le calibrage est donc le procédé qui détermine les caractéristiques géométriques d'une caméra pour en déduire [Toscani, 1987]:

- Des informations tridimensionnelles à partir des coordonnées image bidimensionnelles et, réciproquement,
- les coordonnées image bidimensionnelles à partir d'informations tridimensionnelles.

Ces deux étapes correspondent respectivement à la détermination des paramètres intrinsèques (caractéristiques internes de la caméra) et extrinsèques (déplacement de la caméra). Dans le cas où les paramètres intrinsèques sont connus, le calibrage reviendrai à déterminer uniquement les paramètres extrinsèques.

La littérature offre un ensemble de techniques de calibrage, certaines nécessitent en entré des point tridimensionnels non coplanaires, nous pouvons citer alors la méthode de Toscani [Toscani, 1987] l'un des pionniers du domaine qui propose une résolution mathématique du problème de calibrage à base de la méthode des moindre carrées, la résolution du système d'équations nécessite au minimum six (6) points non coplanaires.

Une autre approche de résolution proposée par Demonthon [Dementhon et al., 1992] nommée POSIT (Pose from Orthographie and Scaling with ITerations) nécessite au minimum quatre (4) points non coplanaires. Elle se base sur une approximation de la projection perspective avec une projection orthogonale et calcule à partir de cette projection la rotation et translation de la camera, ensuite converge vers les paramètres relatives à la projection perspective, en quelques itérations de l'algorithme en utilisant la matrice de rotation et le vecteur de translation calculés lors de l'étape précédente.

D'autres méthodes sont apparues et ont connus un grand succès dû au fait qu'elles se basent sur des points coplanaires, nous pouvons citer la méthode de Zhang [Zhang, 2000] basée sur l'homographie que l'on peut observer entre un objet plan dans l'espace et son image. Il y'a eu également une version coplanaire de l'algorithme POSIT appelée Coplanar-POSIT proposée par Denis Oberkampf [Oberkampf et al., 1993] cette méthode permet d'estimer la pose en utilisant uniquement quatre (4) points coplanaires, l'aspect itérative permet 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.

5. Conclusion

Ce chapitre nous a permis de faire un tour d'horizon sur les approches et techniques du traitement d'images qui peuvent être utilisées pour la localisation en RA.

Nous avons déterminé deux aspects à prendre en compte pour la localisation en RA à savoir l'aspect imagerie qui permet d'extraire des informations à partir des images acquises par la camera et de leurs donner une signification, et l'aspect géométrie 3D, modélisation de caméra, et calcul de pose, qui nous permet de déterminer le point de vue de la camera pour pouvoir recaler correctement les objets virtuels sur la scène réelle.

La littérature étant riche en méthodes, techniques et algorithmes pour chacun des deux aspects, chaque méthode présente des forces et des faiblesses, le choix dépendra toujours du besoin en application. Nous avons essayé de citer les méthodes les plus connus et de donner une vue d'ensemble des parties à prendre en compte pour la localisation en RA.

Chapitre 3 : Approche Proposée (Modélisation du Système)

APPROCHE PROPOSEE

1. Introduction

La reconnaissance et le suivi d'objets, la réalité augmentée et la localisation 3D sont des domaines de recherche en plein essor. De nombreux travaux ont été réalisés dans ces domaines là comme nous avons pu le constater dans les deux premiers chapitres, aussi, de nombreux laboratoires de recherche travail encore sur ces thématiques.

L'objectif de ce présent travail est de déterminer le point de vue de l'utilisateur dans la scène ce qui revient à déterminer le point de vue de la camera afin de pouvoir insérer des objets virtuels d'une manière cohérente. Ce qui se traduit en un ensemble de modules et d'algorithmes que nous allons aborder dans ce qui suit.

Dans ce chapitre nous allons commencer par une présentation du projet et ses objectifs, puis nous passerons à la description du système en abordant les modules à développer lors de l'implémentation, nous présenterons également l'approche proposée pour la partie tracking (suivi) et les algorithmes choisit pour chaque module et nous terminerons par une conclusion.

2. Présentation du projet

Notre travail rentre dans le cadre d'un projet de recherche initié par l'équipe IRVA (Interaction homme système Réalité Virtuelle et Augmentée) au sein de la Division Robotique et Productique du Centre de Développement des Technologies Avancées (CDTA). Le projet est intitulé «Interaction 3D multimodale et collaborative dans un environnement de réalité virtuelle et augmentée».

L'objectif scientifique du projet est l'étude et la mise en œuvre de nouveaux modèles et techniques logicielles pour l'assistance à l'interaction et à la collaboration dans des environnements de réalité virtuelle et augmentée utilisant ou simulant des systèmes complexes.

Dans ce contexte, notre travail touche la partie insertion d'objets virtuels, c'est-à-dire : Comment insérer un objet virtuel dans la scène avec une cohérence spatio-temporelle ? Quelles sont les techniques issues de la vision par ordinateur qui nous permettent d'aboutir à ce résultat ?

3. Objectifs

A partir de l'intitulé de notre sujet qui est « Approche de localisation basée sur les points d'intérêts : Application à la réalité augmentée » nous pouvons extraire un ensemble de mots clés : « localisation », « points d'intérêts » et « réalité augmentée ».

Dans les applications de réalité augmentée la localisation consiste à déterminer le point de vue de l'utilisateur dans la scène, ce dernier est en générale considéré comme étant le point de vue de la camera, et dans certains cas calculé par rapport au point de vue de la camera. On entend par points d'intérêts : un ensemble d'indices visuels qu'on peut extraire de l'image et qui nous donnent des informations sur la scène.

De ce fait, la localisation basée sur les points d'intérêts consiste à utiliser les points d'intérêts pour extraire des informations de la scène et choisir un repère afin de déterminer le point de vue de l'utilisateur par rapport à ce repère.

A partir de la et des deux chapitres précédents nous pouvons déterminer deux objectifs principales :

1- Suivi d'un objet dans la scène à partir des indices visuels extrait de l'image capturée par la camera,
2- Estimation de la pose de la camera (localisation) à partir des données issues de la première phase et insertion de l'objet virtuel.

4. Description du système

Le schéma ci-dessous montre la structure globale de notre application. Le système de capture permet de gérer le flux vidéo (activation de la camera et acquisition d'image) et transmet l'image en question au système de tracking (système de suivi) et au système graphique, le système de tracking permet le suivi d'un objet dans la scène à travers les image capturées, les résultats issus du tracking sont transmis au module de localisation 3D qui permet de déterminer le point de vue de l'utilisateur, les résultats concernant la localisation sont remis au système graphique quant à ceux qui concerne le tracking, un rendu visuel est affiché sur l'interface de l'utilisateur. Le système graphique se charge de la composition du rendu graphique et permet le recalage correct des objets virtuels sur la scène réelle.

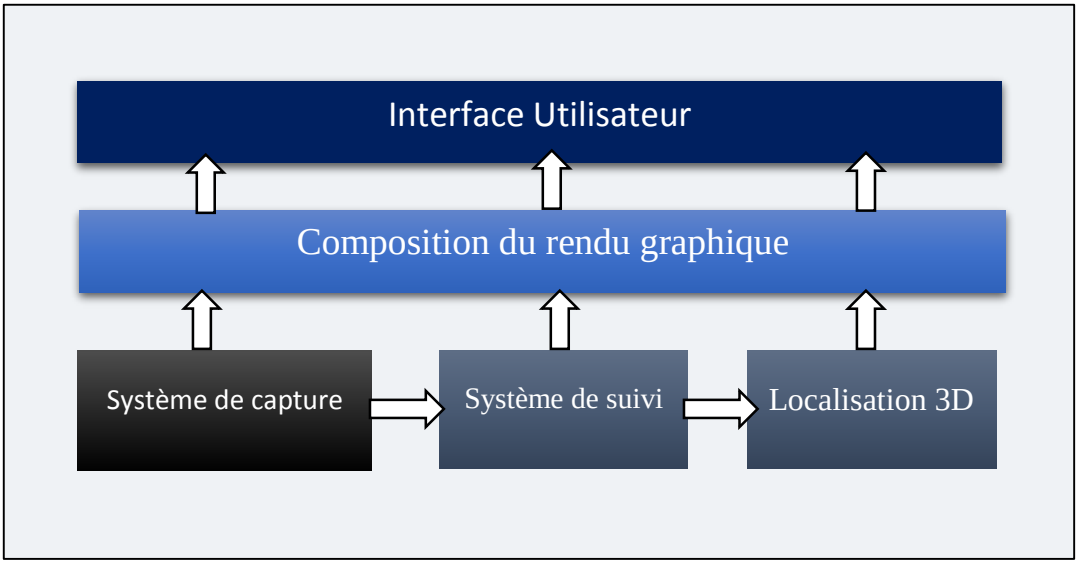


Figure 3. 1 Description globale du système.

4.1. Système de capture

Le système d'acquisition d'image est composé d'une caméra, d'un driver et d'une partie software permettant de gérer le flux vidéo (voir Figure 3.2).

Afin de pouvoir utiliser le flux vidéo, il est nécessaire d'avoir un modèle mathématique correspondant au processus d'acquisition d'images. Le modèle sténopé (chapitre 2-Section 2) est le modèle utilisé en réalité augmentée, en effet, connu pour sa simplicité, il correspond parfaitement aux applications de RA car il est basé sur le principe de projection perspective.

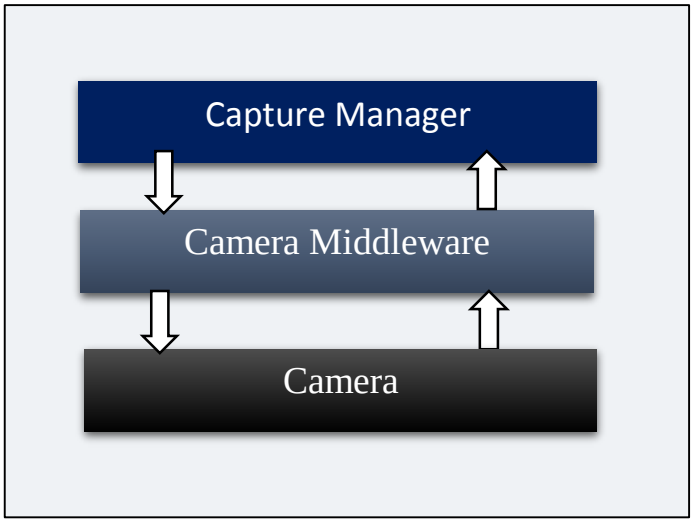


Figure 3. 2 Système de capture.

4.2. Système de tracking

Le système de tracking représente le cœur de ce travail. En effet, il permet de suivre dans le temps un objet sur les images acquises par la camera, l'objet en question est considéré comme un repère dans la scène nous permettant par la suite de localiser la camera par rapport à cette objet.

Concernant la méthode de suivi, nous avons optés pour une méthode à base de cibles naturelles. Nous avons abordé dans le chapitre précédent les différentes techniques existantes qui se basent sur les points d'intérêts, chaque méthode passe essentiellement par trois phases qui sont la détection des points d'intérêts, la description des points d'intérêt, et la mise en correspondance (ou le matching) :

- Pour la détection des points d'intérêts, nous avons choisi le détecteur FAST (Features from Accelerated Segment Test) (voir chapitre 2-Section 3.1.2 A.) connu pour sa rapidité,

- Pour ce qui est de la description des points d'intérêts, nous avons choisi MOBIL pour « MOments based BInary differences for Local description » (voir chapitre 2-Section 3.1.2-B-e) connu pour sa robustesse, son invariance au changement de luminosité, au changement d'échelle et du point de vu, il est aussi rapide en temps description, c'est un descripteur binaire qui se base sur les moments ce qui fait qu'il bénéficie de la robustesse des descripteurs à virgules flottantes et de la rapidité des descripteurs binaires.

- Enfin, pour le matching nous choisissons la méthode du plus proche voisin qui est la plus utilisée à cause de sa simplicité, suivi de l'algorithme RANSAC qui permet d'éliminer les erreurs de matching.

La figure ci-dessous décrit le processus de reconnaissance d'objet adopté. L'image de référence représente la connaissance à priori de l'environnement, la détection des points d'intérêts et leurs descriptions permet d'avoir des informations sur l'image de référence (qui représente l'objet à reconnaitre) et celle acquise par la camera, chaque vecteur descripteur contient des informations concernant le point d'intérêt au quel il est associé, l'étape du matching permet de comparer les vecteurs calculer à partir des deux images et de déterminer l'ensemble des points d'intérêts de l'image acquise par la camera qui correspondent à ceux de l'image de référence, ce qui nous permet d'avoir l'homographie « H » entre l'image de référence et l'image capturée, cette homographie permet de localiser l'objet de référence sur l'image en question. Le suivi se base sur ce processus de

reconnaissance qui est refait pour chaque image acquise par le système de capture et en temps réel.

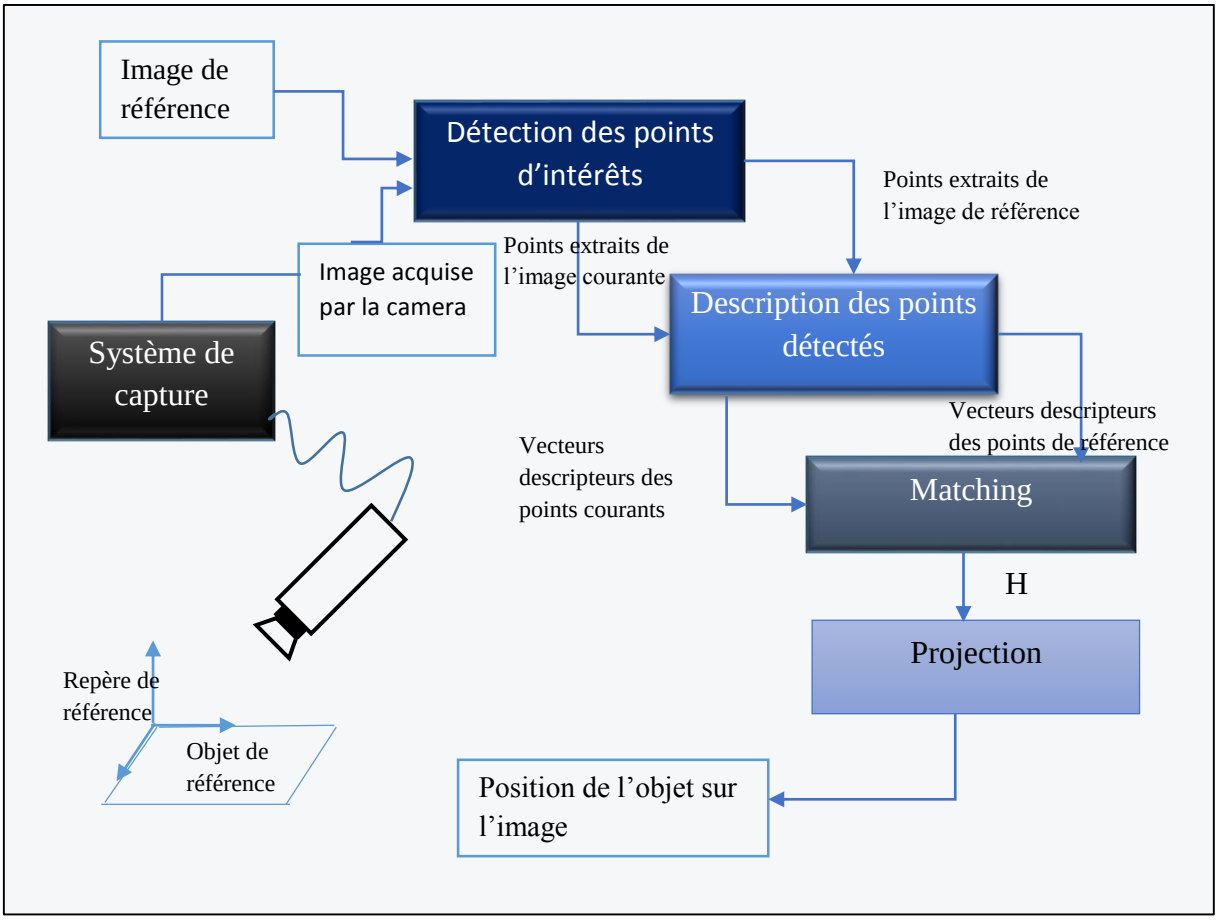


Figure 3. 3 Processus de reconnaissance d'objet.

4.2.1. Amélioration du système de tracking

Afin d'améliorer la robustesse du tracking d'objet nous proposons d'ajouter un calcul d'homographie récursive, dans le but de gagner en temps de calcul et d'avoir une meilleur stabilité et plus de mobilité de l'utilisateur dans la scène réelle.

Principe de l'homographie planaire :

Soit i, j, k trois plans tel que H_j^i est l'homographie qui lie les deux plans i et j, et H_k^j est l'homographie qui lie les deux plans j et k, ce qui se traduit comme suit :

$$\forall\, p \in i : H_j^i \times p = q \; avec \; q \in j \qquad \text{et} \qquad \forall\, q \in j : H_k^j \times q = g \; avec \; g \in k$$

L'homographie H_k^i qui lie les deux plans i et k est définie comme suit :

$$\forall\, p \in i\colon \underbrace{H_k^j \times H_j^i}_{H_K^i} \times p = g \; avec \; g \in k$$

Initialisation du système

Dans un premier temps, nous calculons l'homographie entre l'image acquise par la camera et l'image de référence (voir Figure 3.3), soit H_{init} l'homographie initiale.

Estimation d'homographie inter-image :

Il est à noter que les images issues d'un flux vidéo présentent une certaine cohérence temporelle, c'est-à-dire le mouvement de la camera n'étant pas très important entre deux images successives, ces dernières présentent de ce fait des similarités mais aussi des différences, en se servant de cette caractéristique nous gagnant en temps de matching vu qu'il n'y'a pas une grande différence entre les deux images, aussi même dans le cas où l'objet n'apparait pas complètement sur l'image, nous pouvons tout de même le localiser.

Pour l'extraction d'informations à partir des images nous gardons le même principe utilisé pour la reconnaissance d'objet (détection, description et matching) sauf que ces étapes s'appliquent sur les images successives et non l'image de référence.

Description : à l'instant t_i l'image (i) est capturée par la camera, après la détection et description des points d'intérêts de l'image (i) nous comparons cette description à celle de l'image acquise à l'instant t_{i-1}, ce qui nous permet d'obtenir l'homographie H_i^{i-1} qui représente la matrice de passage de l'image (i-1) à l'image (i) : $H_i^{i-1} \times p_{i-1} = p_i$, avec p_{i-1} (respectivement p_i) point de l'image (i-1) respectivement l'image (i). La figure 3.4 Schématise le processus de calcul d'homographie inter-image.

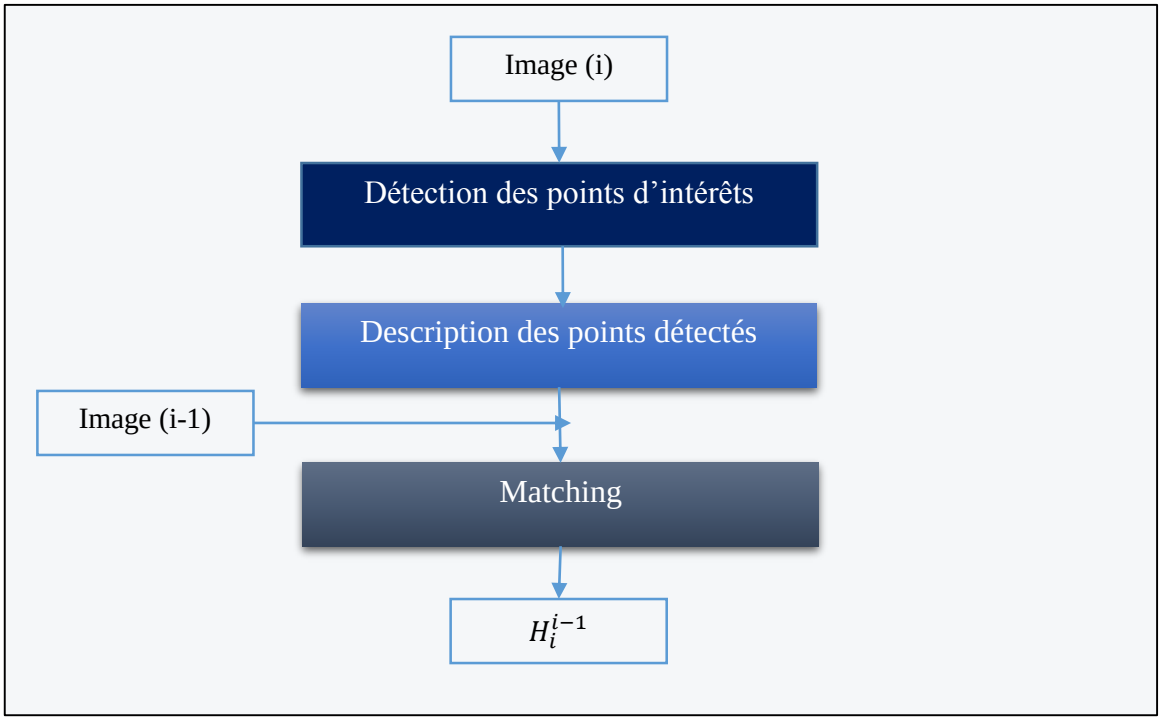


Figure 3. 4 Homographie inter-image. (La description de l'image (i-1) se fait à l'instant i-1).

Homographie récursive :

L'homographie H_i^{i-1} représente la transformation entre l'image capturée à l'instant (i-1) est l'image capturée à l'instant i. Nous devons à présent calculer l'homographie H_i qui est la transformation entre l'image de référence et l'image (i).

Soit :

$H_0 = H_{init}$ (Voir initialisation du système)

Selon le principe d'homographie planaire :

$$\begin{cases} H_1 = H_1^0 \times H_0 \\ H_2 = H_2^1 \times H_1 \\ \cdots, \\ H_i = H_i^{i-1} \times H_{i-1} \end{cases}$$

D'une manière générale, dans le cas d'images séquentielles :

$$H_n = \prod_{i=1}^{n} H_i^{i-1} \times H_0, avec\ H_0 = H_{init}\ et\ H_i^{i-1}\ l'homographie\ entre\ les\ deux\ images\ (i-1, i)$$

Le fait d'utiliser l'homographie inter-image nous permet d'assurer le suivi de l'objet même dans le cas où ce dernier n'apparait pas complétement ou n'apparait pas du tout sur l'image. Ainsi l'utilisateur peut bouger sans se soucier si l'objet est visible ou pas.

4.3 Système de localisation 3D

Ce que nous avons présenté jusqu'à présent concerne la partie imagerie, c'est-à-dire comment extraire des informations à partir des images capturées par la camera. A présent, comment utiliser ces informations pour déterminer le point de vue de la camera par rapport à l'objet suivi ?

Déterminer le point de vu de la camera revient à calculer la matrice de transformation perspective. Rappelons qu'un point $p_i(X_i, Y_i, Z_i)$ dans l'espace forme une image $q_i(u_i, v_i)$ sur l'image acquise par la camera, selon le modèle sténopé la transformation effectuée par la camera est donnée comme suit (voir chapitre 2- Section 2) :

$$s\begin{pmatrix} u_i \\ v_i \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{pmatrix}}_{M} \begin{pmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{pmatrix}$$

D'où :

$$\begin{pmatrix} u_i \\ v_i \end{pmatrix} = \begin{pmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -uX_i & -uY_i & -uZ_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -vX_i & -vY_i & -vZ_i \end{pmatrix} \times C$$

Avec : $C = (m_{11} \quad m_{12} \quad m_{13} \quad m_{14} \quad m_{21} \quad m_{22} \quad m_{23} \quad m_{24} \quad m_{31} \quad m_{32} \quad m_{33})^T$

Si nous avons un ensemble de points p_i de coordonnées 3D connues (X_i, Y_i, Z_i) et leurs correspondant sur l'image q_i de coordonnées (u_i, v_i) nous pouvons alors résoudre le système d'équation et déterminer le vecteur C qui compose la matrice de transformation M.

On choisit un point de l'objet suivi comme étant le repère de la scène réel, et on considère un ensemble de points sur l'objet comme étant nos p_i, quant aux points correspondant sur l'image : les q_i, on peut les déterminer grâce à la partie tracking en utilisant l'homographie calculée. La figure ci-dessous illustre l'idée :

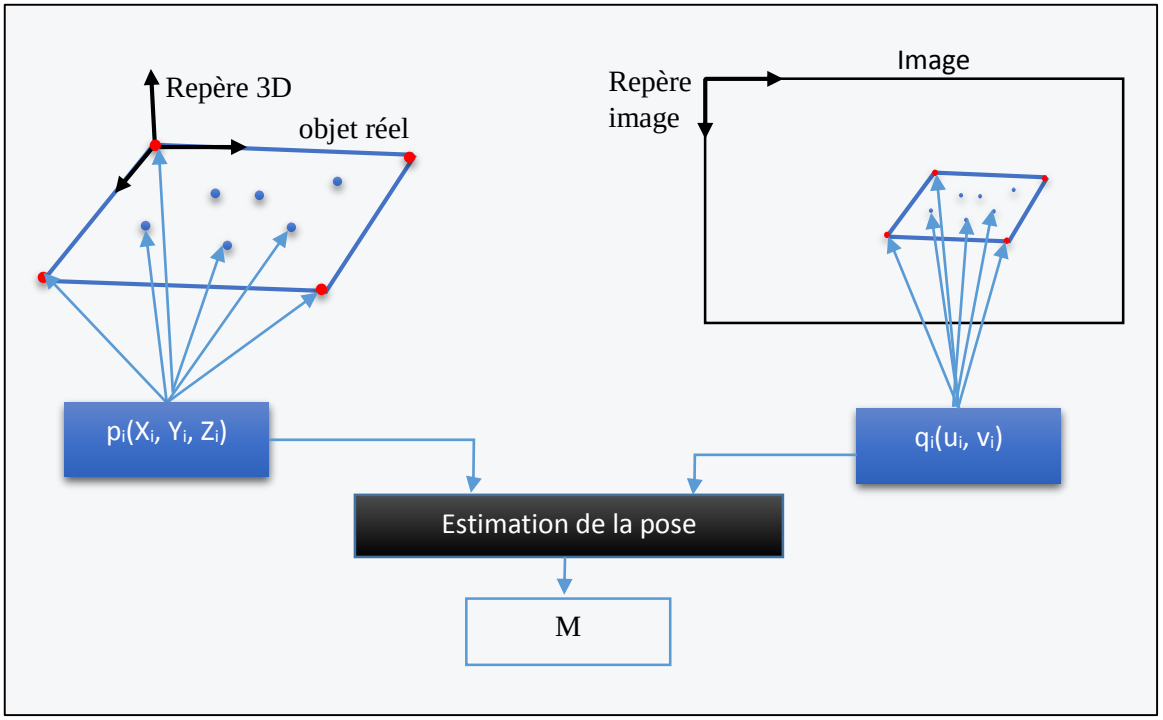


Figure 3. 5 Schéma d'estimation de la pose.

Dans un souci de simplification nous choisissons l'algorithme Coplanar-POSIT afin de calculer la matrice de transformation M à partir de laquelle nous pouvons calculer la matrice de rotation et le vecteur de translation correspondants à la position de la camera par rapport au repère de la scène, cette algorithme nécessite seulement 4 points coplanaires pour déterminer les paramètres extrinsèques de la camera, ce qui correspond à notre cas de figure, son principe de calcul itératif permet d'avoir la meilleur estimation possible tout en étant rapide, ce qui fait qu'il donne de meilleurs résultats que les algorithmes similaires utilisant des données coplanaires.

4.4 Composition du rendu graphique

Ce module permet d'insérer les objets virtuels dans la scène réelle de donner à la camera virtuelle les même propriétés que la camera réelle, afin de générer un environnement augmenté cohérent avec la réalité (Cette partie sera d'avantage détaillée dans le chapitre suivant).

La figure ci-dessous montre le recalage de l'objet virtuel sur l'objet réel en utilisant la matrice de transformation perspective.

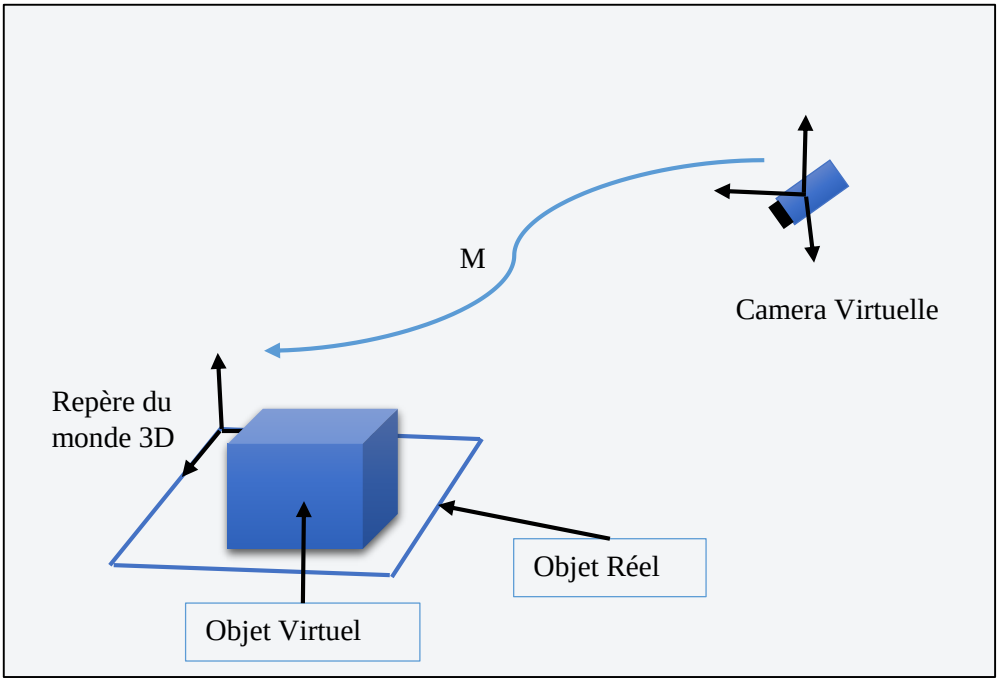


Figure 3. 6 Insertion de l'objet virtuel.

4.5 Déroulement du processus de suivi et d'augmentation

Nous présentons dans ce qui suit le déroulement des différentes étapes citées précédemment, le schéma ci-dessous montre ainsi l'enchainement de ces étapes : reconnaissance d'objet, appariement, estimation de pose…etc.

Lors de l'acquisition d'une image, la première étape à effectuer est la reconnaissance d'objet qui consiste en détection, description et matching des points avec ceux de l'image de référence. En parallèle on calcule l'homographie récursive en utilisant l'homographie inter-image, puis nous effectuons un test comparatif entre les deux homographies pour n'en choisir qu'une seule. L'homographie nous permet de positionner l'objet sur l'image.

A partir de là nous pouvons estimer la pose de la camera. Une fois la pose estimée, nous pouvons augmenter la scène réelle avec des informations virtuelles, le processus est refait pour chaque image acquise par la camera.

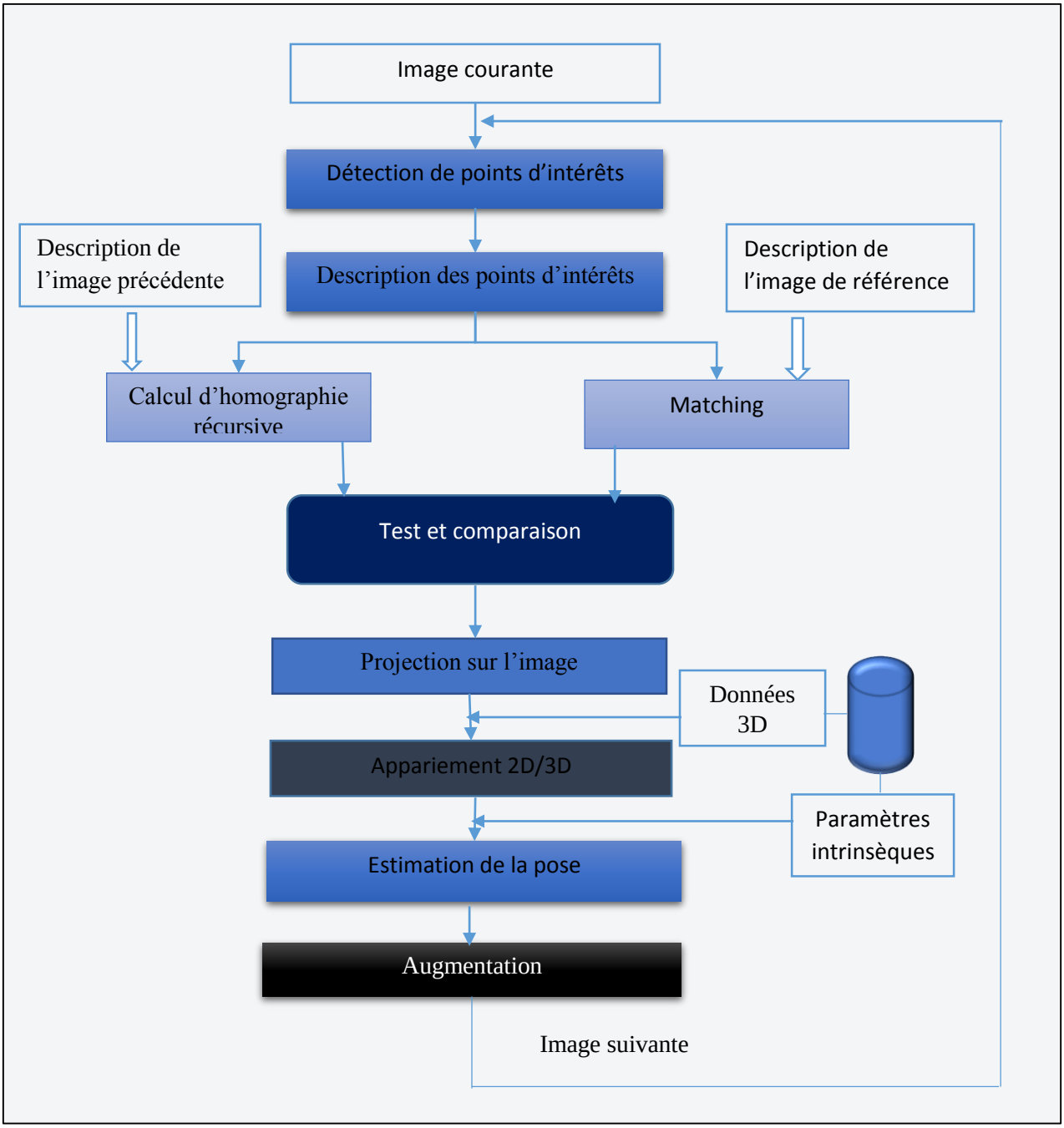


Figure 3. 7 Processus de suivi d'objet.

5. Diagramme de classe

La figure ci-dessous montre le diagramme de classe correspondant à l'application. La classe « Image » doit contenir les propriétés liées à l'image (données pixels, type…) et les fonctions qui permettent de manipuler l'image. La classe « Camera » doit contenir les propriétés de la camera (paramètres intrinsèques, paramètres extrinsèques…) et doit gérer l'acquisition de l'image. La classe « ObjetRef » doit contenir les informations concernant l'objet de référence (son image, position 3D…). Chaque image contient un ensemble de points d'intérêts permettant de la décrire, la classe « Points contient les informations descriptives liées au point d'intérêt, l'objet de référence est également défini par un ensemble de points. La classe « Objet3D » permet de définir l'objet virtuel est contient donc les informations concernant l'objet en question qui est positionné sur la scène par rapport à l'objet de référence, la camera est également localisée par rapport à cet objet.

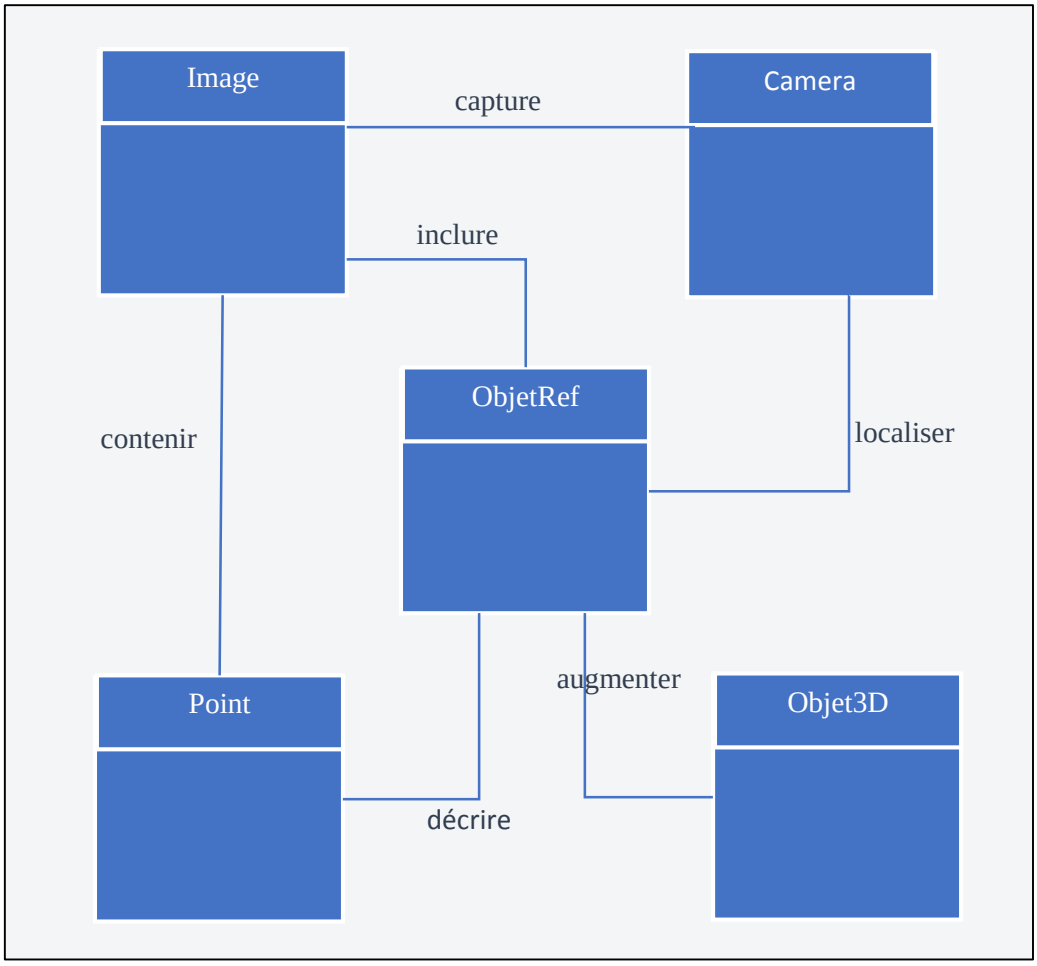


Figure 3. 8 Diagramme de classe.

6. Conclusion

Ce présent chapitre nous a permis d'aborder d'un point de vue mathématique et algorithmique, les différentes parties à implémenter en choisissant les méthodes à développer. Nous avons également proposé une approche de suivi basé sur un calcul d'homographie récursive en bénéficiant de la cohérence temporelle.

Ainsi, nous présentons dans ce chapitre une description de notre système et des modules qui le composent. La modélisation du système et le choix des algorithmes à implémenter nous permet d'aborder au mieux la prochaine étape qui est la réalisation.

De ce fait, le chapitre suivant présentera les différents outils matériels et logiciels utilisés pour le développement de notre application, ainsi que les résultats obtenus.

Chapitre 4 : Implémentation et Tests

IMPLEMENTATION ET TESTS

1. Introduction

Dans cette partie, nous exprimons l'implémentation de notre système qui est une phase au cours de laquelle les algorithmes définis dans le chapitre précèdent sont traduits dans un langage de programmation. Nous allons commencer par les outils matériels et logiciels utilisés pour la réalisation de ce travail, nous présentons ensuite le principe de la programmation 3D et le lien avec la partie théorique, puis nous passons à la partie résultat, on terminera enfin par une petite discussion autour des résultats obtenus et une conclusion.

2. Contexte matériel et logiciel

2.1. Equipement

Le matériel utilisé consiste en une caméra Logitech C920 et un pc de bureau HP avec un processeur intel® (i3 de 3.20 GHz) et 6GO de RAM (voir figure 4.1 a, b). L'application est également testée sur un casque HMD video Vuzix (voir figure 4.1 c).

Figure 4. 1 Matériel utilisé.

2.2 Software

Au cours de l'implémentation de notre projet, nous avons utilisé comme environnement de développement Microsoft Visual studio 2013. Nous avons implémenté les algorithmes et interfaces sous le langage C# et nous avons utilisé comme librairies externes de développement EmguCV, et DirectX. Dans ce qui suit nous expliquons le choix et l'intérêt de chaque Environnement et outil choisis.

2.2.1 Visual Studio 2013

Visual studio est un ensemble d'outils de développement d'applications de Microsoft. Il permet grâce à son environnement de développement intégré (IDE (Integrated Development Environment)) de proposer des solutions faisant appel à plusieurs langage comme Visual Basic, J#, C#, C++, etc. Par ailleurs le Framework .NET 4.0 qui est Intégré par défaut offre plusieurs fonctionnalités comme l'accès aux données, la prise en charge du protocole web, et la prise en charge de librairies dynamique (DLL (Dynamic Link Library)) de Windows, etc.

2.2.2 Langage C#

Le langage C# est un langage de programmation orienté objet. La syntaxe de ce langage est caractérisée par sa simplicité ce qui rend le processus de développement moins pénible. A l'aide d'un processus appelé « interopérabilité », C# offre la possibilité d'interagir avec d'autres logiciels et composants Windows tel que des objets COM et DLL Win 32 native. Grace à cette fonctionnalité C# est enrichi par tous les avantages qu'offre le C++.

2.2.3 Librairies de vision par ordinateur et 3D

Nous avons eu recours à des libraires supplémentaires, comme OpenCV, EmguCV utilisées dans le domaine de la vision par ordinateur et DirectX utilisée dans le domaine de la programmation 3D.

A. OpenCV

OpenCV est une bibliothèque open source développée par INTEL. Elle contient plus de 500 algorithmes optimisés, destinés pour le traitement d'images et de vidéos. OpenCV est conçu principalement pour le développement d'application temps réel. Elle prend avantage des performances des nouveaux processeurs multi-cores. Un des objectifs d'openCV est d'offrir une facilité d'utilisation des grandes technologies de traitement d'images à la communauté de vision par ordinateur. Dans sa version 2.3, OpenCV introduit le module OpenCV GPU. Ce module consiste en un ensemble de classes et fonctions

permettant d'accélérer les calculs en utilisant le GPU. Ces classes et fonctions ont été développées pour une utilisation sur la plateforme CUDA qui équipe les cartes graphiques NVIDIA récentes.

B. EmguCV

EmguCV est un WRAPPER de la bibliothèque OpenCV en C#. On y retrouve la quasi-totalité des fonctionnalités d'OpenCV permettant d'effectuer toutes les tâches basique d'analyse et de traitement d'image.

C. Direct X

C'est un simulateur virtuel qui fournit des outils permettant de créer des scènes virtuelles en 2D ou en 3D. Pour augmenter une scène réelle avec des objets virtuels, il faut essentiellement adapter la matrice des paramètres extrinsèques de la scène réelle avec la matrice "world" de DirectX, comme il faut aussi initialiser la matrice "projection" de DirectX avec la matrice des paramètres intrinsèques.

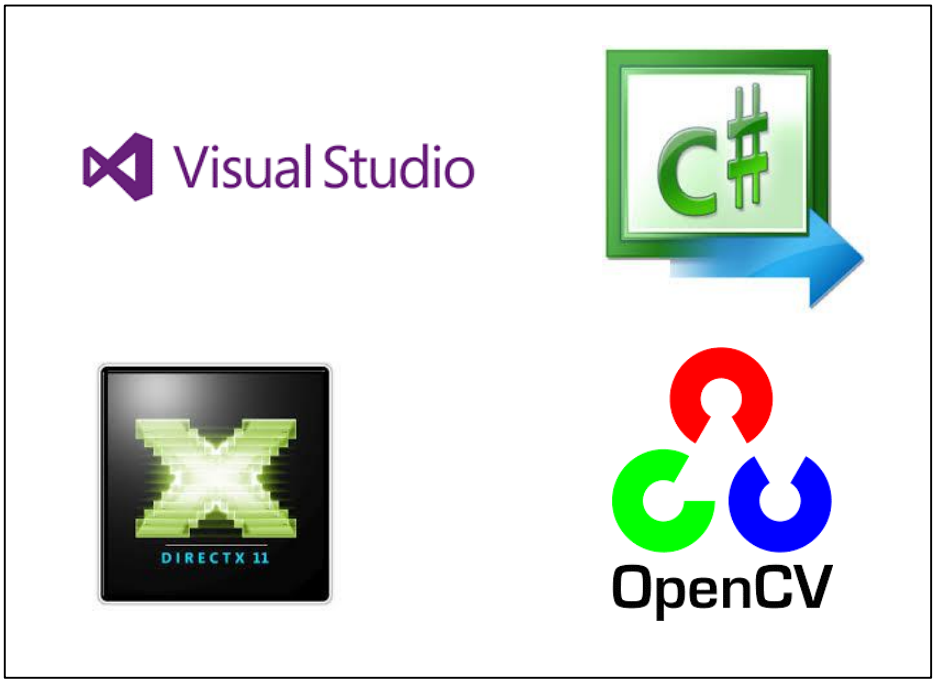


Figure 4. 2 Outils de développement.

3. Principe de la programmation 3D

Nous présentons dans cette partie les fondements de base de la programmation 3D que nous allons utiliser afin d'avoir une scène augmentée. La 3D est régit par un ensemble de concepts fondamentaux appelés espaces à savoir : monde (world), vue (view), projection, locale et espaces d'écran. Nous détaillons dans ce qui suit chacun de ces concepts.

3.1. Espace locale

Un espace local est un espace relatif à un objet. Lorsque nous créons un objet, nous le centrons en générale autour du point (0, 0, 0), ce qui rend beaucoup plus facile la création et la définition des sommets. L'espace local définit la position d'un sommet par rapport aux autres sommets dans cet objet. Les positions des sommets dans l'espace locale sont définies généralement dans un fichier contenant l'objet 3D, créé à partir d'un programme de modélisation 3D.

Prenons l'exemple d'une forêt, si nous devons faire une forêt. Ce serait un gaspillage de créer une tonne d'arbres qui définissent la forêt, quand tout ce que nous avons à faire c'est de créer l'arbre une fois et faire des copies repositionnées à des endroits différents, d'où l'intérêt de l'espace locale.

3.2. Espace du monde (world space)

L'espace du monde est utilisé pour positionner chaque objet par rapport à l'autre, dans la scène 3D. Chaque objet a sa propre matrice de l'espace monde. Cette matrice représente la position, la taille et la rotation de l'objet dans le monde 3D. Tous les objets seront position autour d'un seul point central (0, 0, 0), qui est le centre de l'espace world. L'espace du monde est défini donc par des transformations individuelles sur chaque objet, translations, Rotations et mise à l'échelle.

3.3. Espace de vue (view space)

L'espace view est essentiellement l'espace de la caméra. La caméra est positionnée au point (0, 0, 0) et regarde en direction de l'axe des z et s'élève sur l'axe des y (camera up). Le monde est translaté dans l'espace de caméra. Alors, quand nous faisons des transformations, nous avons l'impression que la caméra se déplace dans le monde 3D, alors qu'en fait, c'est ce monde qui est en mouvement, et la caméra reste fixe.

L'espace de Vue est défini par la création d'une matrice décrivant la position de la caméra, sa direction (cible), et le up qui représente l'axe y de la caméra. Nous pouvons donc facilement créer une matrice vue en utilisant 3 vecteurs, la position, la cible et l'up.

3.4. Espace de projection

L'espace de projection définit la zone de la scène 3D qui sera visible à partir du point de la caméra (Les objets qui seront affichées sur l'écran). D'un point de vu géométrique, l'espace de projection peut être représenté sous forme d'une pyramide dont la pointe est

coupée. La pointe de la pyramide serait la position de la caméra, et l'endroit où la pointe a été coupée serait le z-plan de proximité, et la base de la pyramide serait le plan z éloigné.

Ainsi l'espace de projection définit les objets ou parties d'objets qui seront visible sur l'écran, de sorte à ce qu'on ne voit que les objets qui sont dans l'espace de projection.

3.5. Espace d'ecran

Ce dernier espace représente fondamentalement les valeurs pixeliques (x, y) affichées sur l'écran. L'espace d'écran est en fait l'espace 2D qu'on affiche sur le moniteur.

Nous n'avons pas de définir cet espace, il s'agit plus d'une idée ou un concept de l'espace physique de notre moniteur. Cependant, nous utiliser cet espace dans le cas où nous souhaitons interagir avec nos objet, en prenant les coordonnées x et y de notre souris dans l'espace de l'écran pour voir si nous sommes en train de cliquer sur un objet 3D.

4. Principe d'une application de RA

Dans cette partie du chapitre nous allons montrer comment utiliser le principe de la programmation 3D pour avoir une application de réalité augmentée. En effet, nous pouvons utiliser les espaces présentés précédemment afin d'attribuer à notre monde 3D les même propriétés que notre monde réel.

Afin de créer l'environnement augmenté, l'image capturée par la camera est mise en arrière-plan du monde 3D, l'objet 3D est positionné par rapport au repère du monde 3D cette position représente l'espace du monde (world space) ici on attribue à l'objet la position que l'on souhaite qu'il est par rapport à notre repère réel, la position et l'orientation de la camera virtuelle par rapport au repère du monde 3D correspond proportionnellement à la position de la camera réelle par rapport à notre repère de la scène ce qui représente l'espace de vue (view space) (voir figure 4.3). Pour finir on attribue à l'espace de projection les mêmes propriétés que ceux de la camera réelle (paramètres intrinsèques), ce qui nous permet d'avoir sur l'écran le résultat présenté dans la figure 4.4.

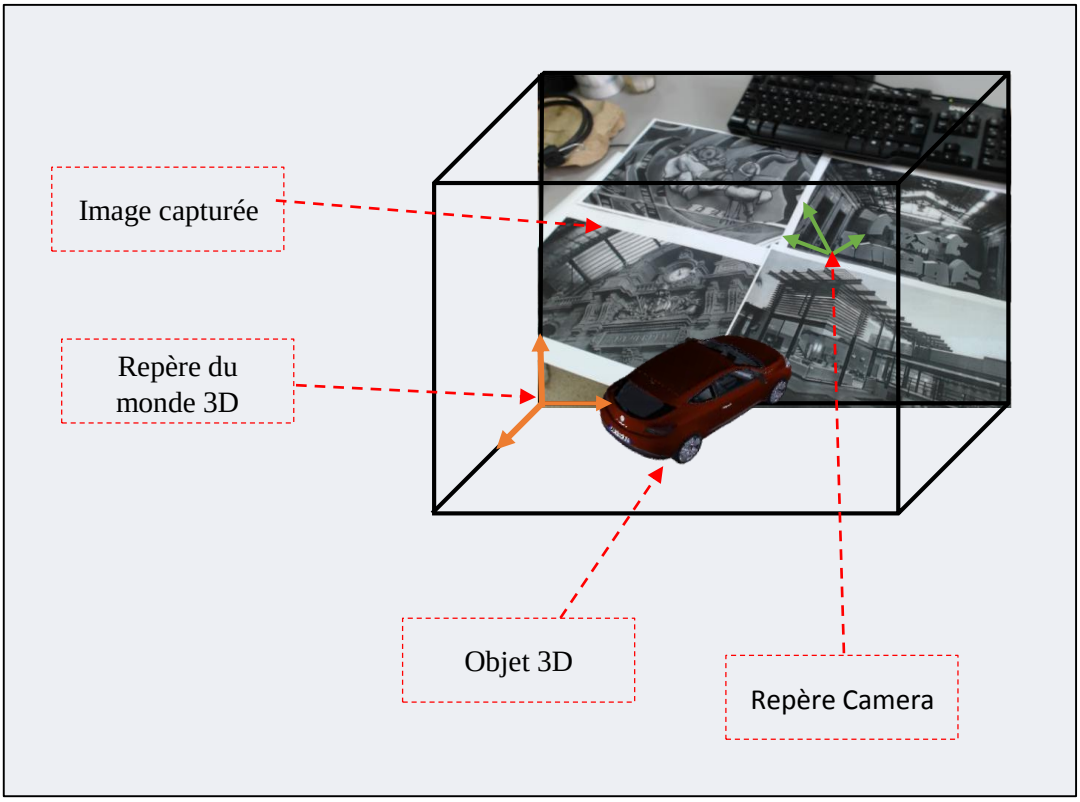


Figure 4. 3 Environnement augmenté.

Figure 4. 4 Vue d'écran d'une scène augmentée.

En résumé, pour réaliser une application de réalité augmentée nous devons avoir une technique de vision suffisamment robuste pour pouvoir reconnaitre une cible naturelle, cette dernière sera utilisée comme repère de la scène réelle. Nous devons également utiliser un algorithme de calcule de pose qui nous permettra de déterminer la position de

la camera par rapport à ce repère. Ces deux parties nous permettent de paramétrer notre monde 3D afin d'assurer un recalage stable et précis des objets virtuels et obtenir donc une scène augmentée (figure 4.4).

5. Reconnaissance d'un objet

Nous présentons dans cette partie les résultats en image de la détection des points d'intérêts, du matching et de la reconnaissance d'objet.

Les figures ci-dessous montrent chacune de ces étapes, les cercles présents sur la figure 4.5 représentent les zones d'intérêts dont les centres sont les points d'intérêts extraits de l'image capturée et de l'image de référence. La figure 4.6 représente la mise en correspondance entres les points d'intérêts (matching). La figure 4.7 représente la reconnaissance de l'objet.

Comme défini dans le chapitre précèdent, la détection des points d'intérêts ce fait en utilisant l'algorithme FAST, la description ce fait en utilisant MOBIL, le matching ou mise en correspondance ce fait en utilisant l'algorithme du plus proche voisin et RANSAC pour éliminer les erreurs de matching.

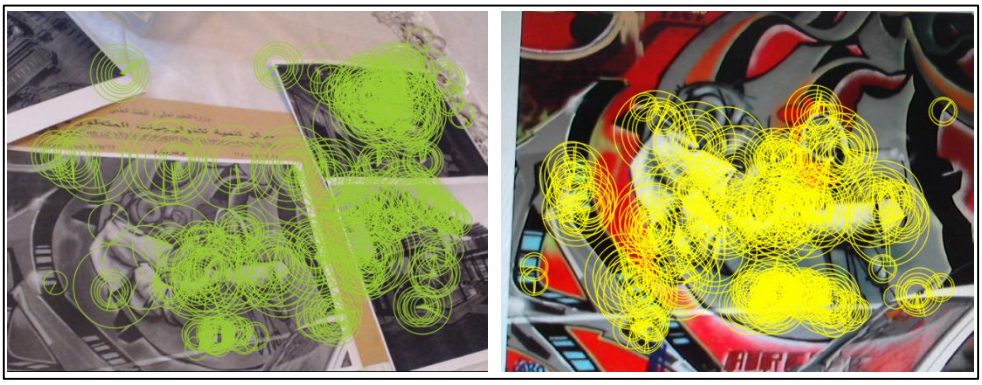


Figure 4. 5 Détection des points d'intérêts (gauche : image capturée, droite : image de référence).

Figure 4. 6 Mise en correspondance.

Figure 4. 7 Reconnaissance de l'objet.

6. Vers l'immersion mobile en réalité augmentée

Ce que nous avons présenté jusque-là, nous permet d'avoir une application de réalité augmentée en présence d'une cible naturelle. Cependant, étant lié à cette cible, nous ne pouvons pas assurer une immersion mobile de l'utilisateur dans son monde réel, car l'absence de la cible naturelle (marqueur naturel) du champ de vision de l'utilisateur cause la disparition de l'objet virtuel inséré. Ce problème apparait fréquemment dans le cas d'un objet virtuel en mouvement.

Comme nous pouvons le constater dans la figure ci-dessous, le champ de mouvement de l'utilisateur est limiter car il doit garder en vue la cible naturelle.

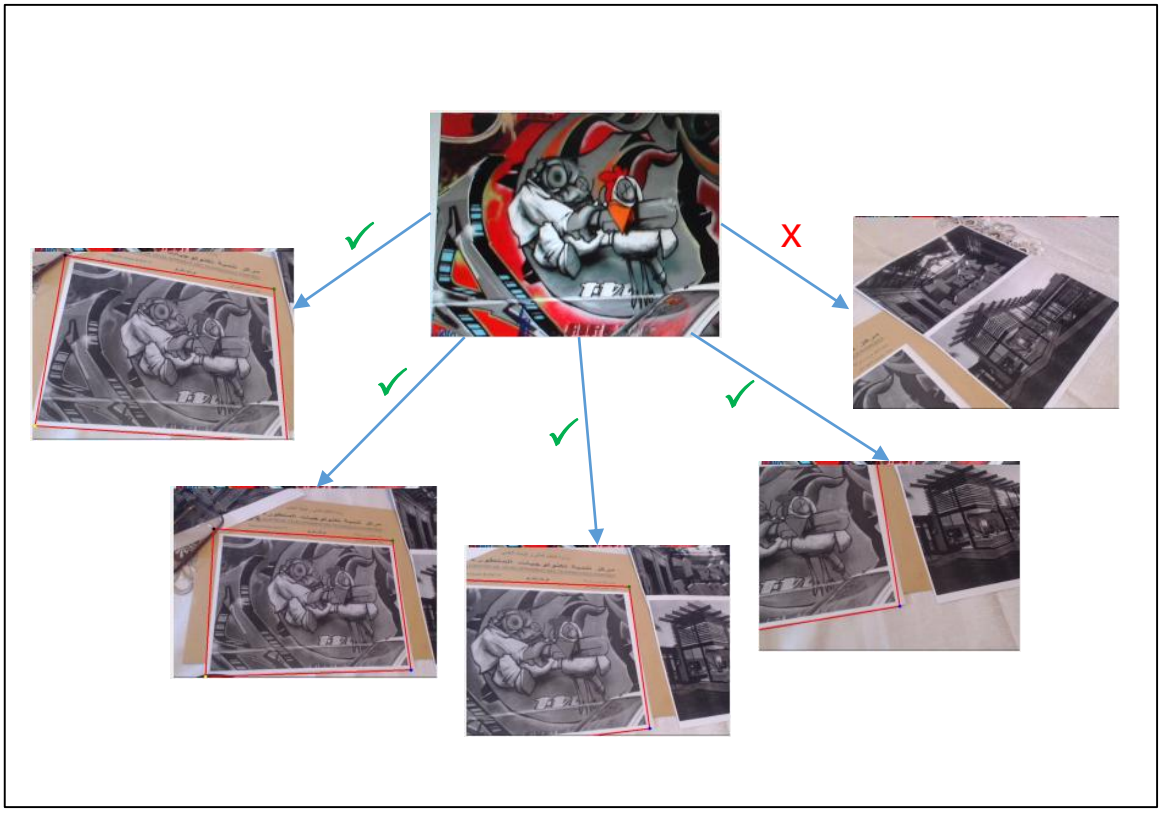


Figure 4. 8 Reconnaissance d'une cible naturelle (localisation limitée => mouvement de l'utilisateur réduit).

Une des solutions à ce problème est la reconnaissance multi-cibles qui permet d'élargir un peu le champ de mouvement de l'utilisateur dans la limite des cibles, dans cette posture, les objets virtuels pourront être omniprésents. Cependant, cette configuration de multi-marqueurs demande une pré-calibration, et dans le cas où une des cibles est déplacée, cela nécessite une re-calibration. Aussi, pour pouvoir se repérer dans la scène nous devons comparer l'image capturée avec toutes les cibles ce qui prends beaucoup de temps lors de l'exécution.

Dans cette optique, nous avons proposé dans le chapitre 3 (section 4.2.1) une approche de calcul d'homographie récursive qui se base sur le descripteur MOBIL afin d'assurer la mobilité de l'utilisateur dans la scène. Cette approche de localisation est donc basée sur une hybridation du descripteur MOBIL avec la mise à jour de la matrice d'homographie à chaque frame capturé [Belghit et al, 2015]. De ce fait, l'utilisateur peut bouger librement dans la scène réelle tout en assurant l'insertion des objets virtuels.

Le principe de fonctionnement de cette approche est donné comme suit :

Dans un premier temps, l'utilisateur doit commencer d'un endroit ou la position est déjà connue, sinon il utilise une cible naturelle pour la première frame pour pouvoir calculer sa position initiale. Une fois que la première frame est capturée et la position initiale est calculée, nous estimons l'homographie planaire entre les images capturée.

Ainsi la position de la cible naturelle est estimée même quand celle-ci n'apparait pas sur l'image capturée (voir figure 4.9).

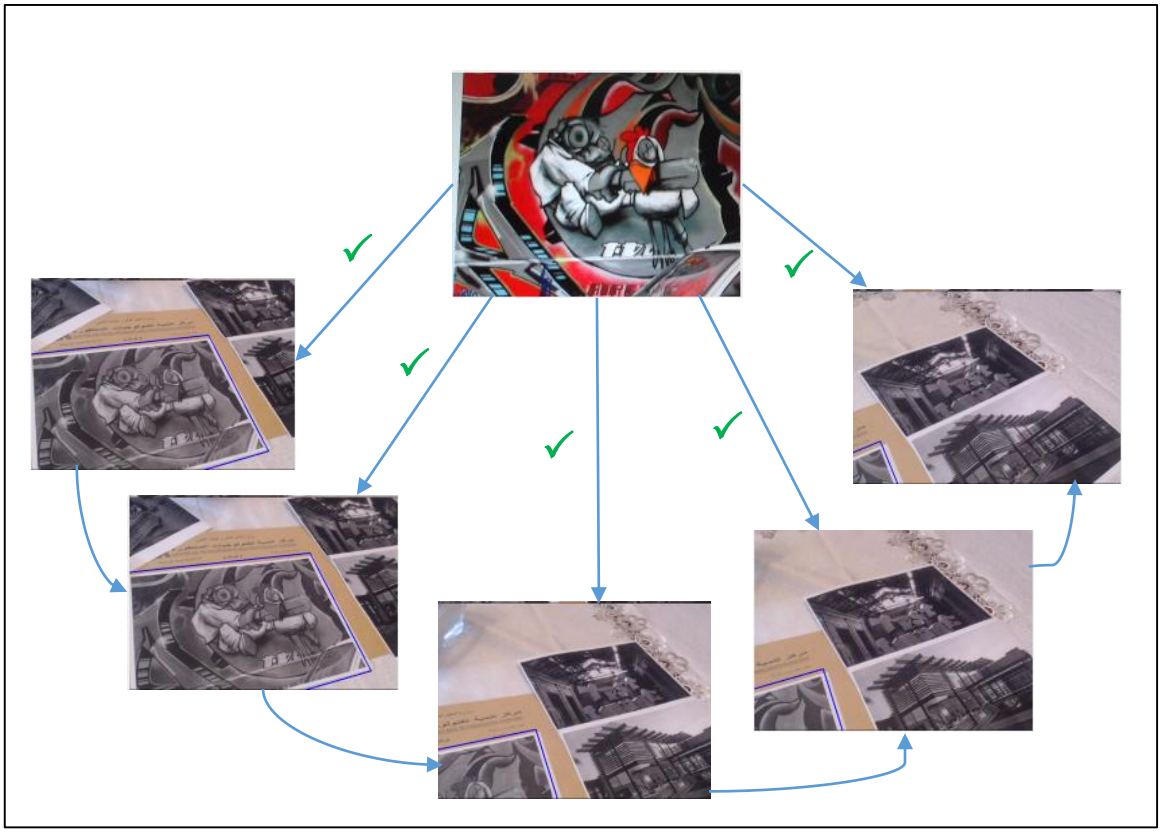


Figure 4. 9 Application de l'homographie récursive.

7. Localisation 3D et augmentation

A partir des résultats obtenus précédemment nous estimons la pose de la caméra à l'aide de l'algorithme récursive (CoplanarPosit), une fois la pose estimée, nous avons toutes les données nécessaires pour pouvoir insérer des objets virtuels.

Nous présentons dans cette partie quelques points de vue d'scène augmentée (voir figure 4.10).

Figure 4. 10 Scène augmentée.

La figure ci-dessous présente une séquence d'images d'un objet virtuel en mouvement :

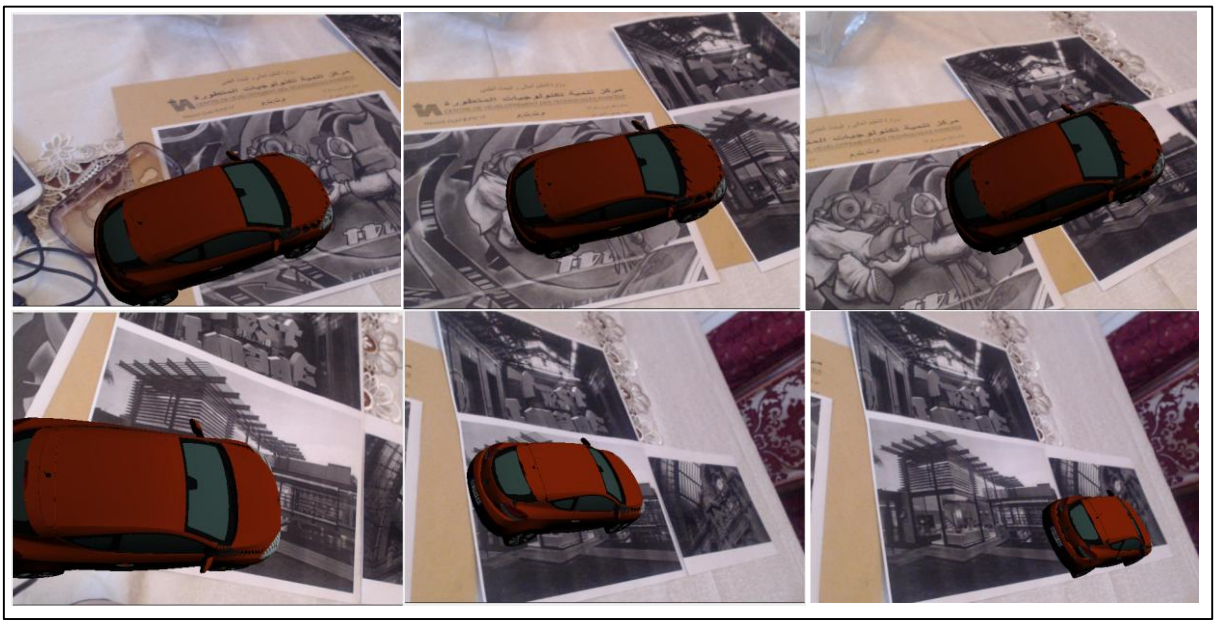


Figure 4. 11 Voiture en mouvement.

8. Evaluation et Tests

Afin d'évaluer notre système nous avons utilisé la même base de données d'images que celle utilisée pour MOBIL_2B à savoir la base de Mikolajczyk et Schmid [Mikolajczyk et al., 2005]. Nous nous sommes basés sur deux critères de tests le temps d'exécution et la précision de l'homographie calculée, le nombre de points est fixé à 800 points d'intérêts.

Le tableau suivant montre les résultats sur le calcul du temps d'exécution

Table 2 Temps d'exécution du système.

Partie testée	Temps d'exécution par frame ms
Detection des points (800 points)	7
Description des points (800 points)	22
Matching/ image de référence (800 points)	8-70
Matching/ image précédente (800 points)	5

La première valeur du tableau indique le temps de détection des points d'intérêt, la seconde le temps de description des points d'intérêts en utilisant MOBIL, la troisième valeur représente le temps du matching des points d'une image i de la base de donnée avec l'image de référence ce dernier varie entre 8ms et 70ms, la dernière valeur montre le temps d'exécution du matching entre deux images successives qui est en moyenne de 5ms. En comparant les deux dernières lignes du tableau nous constatons qu'il y'a un gain dans le temps de matching celui-ci est dû au fait qu'il n y'a une grande différence entre les images successives dans les différentes transformations (échelle, rotation et translation), nous pouvons donc conclure que la méthode proposée a permis de réduire le temps de matching.

A présent comment évaluer la précision ? Il est à noter que la base de données utilisée pour les tests contient une série d'images successives avec des fichiers contenant l'homographie exacte entre chaque image et la première image de la base qui est considéré comme étant l'image de référence, ces différentes homographies sont calculées à l'aide d'un dispositif mécanique pour avoir les résultats exact.

Pour évaluer la précision du système nous avons procédé comme suit :

1- Calcul de l'homographie pour chaque image de la base de données en utilisant notre approche,

2- Détection des points d'intérêts sur l'image de référence,

3- Projection des points détectés sur chaque image i en utilisant l'homographie correspondante $H_{i,}$ nous obtenons un vecteur de points projetés qui représente la position de ces points sur l'image i,

4- Projection des points détectés sur l'image i en utilisant l'homographie exacte H_{i_exact}. Nous obtenons un vecteur de points projetés avec les coordonnées exactes,

5- Calcul de la distance euclidienne entre les éléments des deux vecteurs respectifs, nous obtenons un vecteur de distances.

6- Calcul de la moyenne du vecteur.

Pour évaluer la précision du système de reconnaissance nous avons pratiquement utilisé le même principe :

1- A partir des homographies exactes de la base de données, nous calculons les homographies inter-image $H_{i\_i+1\_exact}$,

2- En utilisant notre système de reconnaissance d'objet, nous calculons également les homographies inter-image H_{i_i-1},

3- A partir de ces deux séries d'homographies nous exécutons les mêmes étapes citées précédemment (projections, calcul de distances, calcul de moyenne).

Le tableau ci-dessous présente l'erreur moyenne de notre système, l'erreur qui est dûe au système de reconnaissance et l'erreur dûe à l'approche récursive utilisée pour le tracking.

Table 3 Erreurs moyennes

Erreur globale (pixel)	Erreur du système de reconnaissance (pixel)	Erreur de l'approche récursive (pixel)
1,0012370	**0,6550013**	**0,3462357**

Nous avons constaté que l'erreur globale varie entre 0.65 et 1.40 pixel ce qui est satisfaisant en termes de précision.

Nous avons également testé MOBIL pour la reconnaissance directe de l'image de référence sur les séquences d'images de la base, la reconnaissance s'est arrêtée au bout de la 4éme image, par contre en utilisant le calcul récursif de l'homographie nous pouvons suivre l'objet de référence sur toutes les séquences d'images malgré les différentes transformations appliquées à l'objet.

9. Discussion des résultats

Ce présent travail nous permet d'insérer des objets virtuels dans une scène réelle. L'utilisation de l'homographie récursive permet d'assurer le tracking en offrant à l'utilisateur plus de mobilité.

L'immersion mobile est un nouveau concept qui est apparue avec l'évolution technologique et le besoin de mobilité. Les recherches actuelles ne sont pas encore arrivées à offrir un système de tracking mobile stable et performant.

Aujourd'hui, les systèmes de réalité augmentée existants sont faits pour un environnement particulier étudié au préalable (lumière, contraste…) ou bien une cible bien choisi et étudiée.

Nous avons proposé une nouvelle approche pour l'immersion mobile, les résultats obtenus sont satisfaisant, toute fois la méthode utilisée requière une scène texturée.

Aussi, l'homographie récursive augmente la stabilité de la reconnaissance de cible quand cette dernière est visible dans l'image. Par contre, lorsque l'utilisateur bouge et la cible n'est pas visible, nous arrivons à localiser l'utilisateur mais avec moins de précision. Ceci est dû d'une part au degré de robustesse du descripteur et d'autre part à la technique de matching.

10. Conclusion

Nous avons abordé dans ce chapitre le principe de réalisation d'une application de réalité augmentée d'un point de vue pratique. Puis nous avons présenté les résultats obtenus pour ce qui est de la reconnaissance d'objet, nous avons également introduit le concept d'immersion mobile avec une approche récursive. Pour finir nous avons présenté les résultats de scènes augmentées.

La RA tend à révolutionner les interfaces homme machine, avec le concept d'immersion mobile, les applications de RA deviennent plus simple et à la portée du grand public.

Dans ce sens et à l'issus des résultats obtenus, comme perspective d'amélioration, nous pouvons envisager d'utiliser une autre méthode de matching qui permet d'avoir plus de robustesse, nous pouvons également penser à améliorer le descripteur de points.

CONCLUSION GENERALE

La Réalité Augmentée (RA) est une technologie qui enrichit le monde réel avec des données et médias digitaux, comme des modèles 3D et des vidéos, qui apparaissent en temps réel en surimpression du flux caméra de votre smartphone, tablette, PC ou lunettes connectées.

Ce phénomène émergeant a pris un grand élan avec le développement et la démocratisation des nouvelles technologies de l'information et de la communication (NTIC).

La vision par ordinateur se trouve au cœur des applications de réalité augmentée. En effet, la RA se base essentiellement sur la vision par ordinateur pour pouvoir assurer un bon recalage des objets virtuels dans la scène réelle.

Nous avons présenté à travers ce document les concepts de base liés à la réalité augmentée ainsi que les méthodes de localisation en RA en mettant l'accent sur les méthodes de reconnaissance d'objets basées sur les points d'intérêts.

Nous avons présenté par la suite l'approche que nous avons adoptée avec une nouvelle technique de tracking qui tend vers l'immersion mobile. Les résultats sont présentés dans le dernier chapitre avec une discussion autour de l'approche proposée.

Nous avons essayé d'aborder ce mémoire d'une manière simple en englobant les concepts nécessaires à la réalisation d'une application de réalité augmentée. Ainsi, une application de RA est composée de deux parties essentielles : la partie imagerie nous permet d'extraire des informations à partir des images capturées par la camera et de suivre un objet dans la scène qui est utilisé comme repère dans la seconde partie qui consiste à déterminer le point de vue de l'utilisateur dans la scène et ceux par rapport à l'objet suivi. Une fois que nous avons déterminé le point de vue de l'utilisateur qui correspond en général au point de vue de la camera nous pouvons alors insérer l'objet virtuel dans la scène réelle afin d'avoir une scène augmentée.

Le fait d'être obligé de se référer à un objet dans la scène réduit le champ de mouvement de l'utilisateur, ce qui limite les applications de RA, aussi au jour d'aujourd'hui la RA demande une scène plus au moins particulière et nécessite des scènes texturées ou avec des contours bien tracé cela est dû aux limites des techniques de vision utilisées. Dans ce sens, la recherche actuelle se concentre sur l'immersion mobile et sur la RA centrée utilisateur. On entend par RA centrée utilisateur des applications simples et moins

exigeantes en ce qui concerne la scène réelle avec plus de liberté de mouvement autrement dit « Réalité Augmentée à la portée de tous ».

Nous avons proposé dans ce travail une technique de calcul d'homographie récursive qui permet d'avoir une certaine mobilité, aussi nous avons constaté que cette technique ajoute de la stabilité à la méthode de reconnaissance d'objet utilisée.

Afin d'améliorer ce travail, en plus de modifier la méthode de matching et d'améliorer le descripteur de point d'intérêts nous pouvons envisager d'introduire une caméra de profondeur pour avoir plus d'information sur la scène et proposer une description des points qui prends en compte leurs position 3D.

Nous pouvons également envisager un modèle de scène à suivre au lieu d'un objet et pourquoi pas un modèle dynamique qu'on modifie au cours du temps.

Pour conclure, la RA connais un grand succès au niveau international et touches de plus en plus les secteurs socio-économiques. Toutefois, ce domaine ne connaît malheureusement pas le même succès en Algérie, notre ambition est d'évoluer d'avantage dans ce domaine et de vulgariser la RA au niveau des universités et pourquoi pas dans l'avenir de tisser des liens avec les différents secteurs socio-économiques en proposant des produit RA qui peuvent aider dans l'accomplissement de certaines tâches.

REFERENCES

[Alahi et al., 2012] Alahi, A., Ortiz, R., et Vandergheynst, P. (2012, June). Freak: Fast retina keypoint. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on (pp. 510-517). Ieee.

[Azuma, 1997] Azuma.R, (1997). A survey of augmented reality. In Presence: Tele-operators and Virtual Environments, 6(4):355_385.

[Bay et al., 2006] Bay, H., Tuytelaars, T., et Van Gool, L. (2006). Surf: Speeded up robust features. In Computer vision–ECCV 2006 (pp. 404-417). Springer Berlin Heidelberg.

[Bellarbi et al., 2014] Bellarbi, A., Otmane, S., Zenati, N., et Benbelkacem, S. (2014, September). [Poster] MOBIL: A moments based local binary descriptor. In Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on (pp. 251-252). IEEE.

[Belghit et al, 2015] Belghit, H., Bellarbi, A., Zenati, N., et Otmane, S. (2015, Avril). Vision based Mobile Collaborative Augmented Reality. The 17th ACM Virtual Reality International Conference (VRIC'15).

[Benbelkacem et al., 2011] Benbelkacem, S., Bellarbi, A., Zerarga, F., Belhocine, M., Tadjine, M., Zenati-Henda, N., et Malek, S. (2011). Augmented reality platform for collaborative E-maintenance systems. INTECH Open Access Publisher.

[Benhimane et al., 2004] Benhimane, S., et Malis, E. (2004, September). Real-time image-based tracking of planes using efficient second-order minimization. In Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on (Vol. 1, pp. 943-948). IEEE.

[Calonder et al., 2012] Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., et Fua, P. (2012). BRIEF: Computing a local binary descriptor very fast. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 34(7), 1281-1298.

[Chekhlov et al., 2007] Chekhlov, D., Gee, A. P., Calway, A., et Mayol-Cuevas, W. (2007, November). Ninja on a plane: Automatic discovery of physical planes for augmented reality using visual slam. In Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality (pp. 1-4). IEEE Computer Society.

[Comport et al., 2006] Comport, A. I., Marchand, E., Pressigout, M., et Chaumette, F. (2006). Real-time markerless tracking for augmented reality: the virtual visual servoing framework. Visualization and Computer Graphics, IEEE Transactions on, 12(4), 615-628. [Ellis et al., 1994] Ellis, C., et Wainer, J. (1994, October). A conceptual model of groupware. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work* (pp. 79-88). ACM.

[Dementhon et al., 1992] DeMenthon, D. F., et Davis, L. S. (1992, January). Model-based object pose in 25 lines of code. In Computer Vision—ECCV'92 (pp. 335-343). Springer Berlin Heidelberg.

[Didier, 2005] Didier, J. Y.(2005).Contributions à la dextérité d'un système de réalité augmentée mobile appliqué à la maintenance industrielle (Doctoral dissertation, Université d'Evry-Val d'Essonne).

[Fiala, 2005] Fiala, M. (2005, June). ARTag, a fiducial marker system using digital techniques. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on (Vol. 2, pp. 590-596). IEEE.

[Fuchs et al., 2006] Fuchs, P., et Moreau, G. (2006). *Le traité de la réalité virtuelle: Volume 2, interfaçage, immersion et interaction en environnement virtuel* (Vol. 2). Presses des MINES.

[Hamidia et al., 2014] Hamidia, M. Zenati-Henda N. Belghit H. et Belhocine,M. (2014, April). Markerless tracking using interest window for augmented reality applications. In Multimedia Computing and Systems (ICMCS), 2014 International Conference on (pp. 20-25). IEEE.

[Haouchine et al., 2013] Haouchine, N., Dequidt, J., Berger, M. O., et Cotin, S. (2013, February). Deformation-based augmented reality for hepatic surgery. In Medicine Meets Virtual Reality, MMVR 20.

[Harris et al., 1988] Harris C and Stephens M, "A Combined Corner and Edge Detector", Proceedings of The Fourth Alvey Vision Conference, Univ. Manchester, pp. 147-151, 1988.

[Herling et al., 2012] Herling, J., et Broll, W. (2012, December). Random model variation for universal feature tracking. In Proceedings of the 18th ACM symposium on Virtual reality software and technology (pp. 169-176). ACM.

[Hugues, 2011] Hugues, O. (2011). Réalité augmentée pour l'aide à la navigation. SIGMA: Système d'information Géographique Maritime Augmentée (Doctoral dissertation, Université Sciences et Technologies-Bordeaux I).

[IKEA, 2015] IKEA, catalogue (2015). La réalité augmentée au service du client.
URL1: http://info.ikea-usa.com/Catalog/
URL2: http://www.marketingtechnologybrief.com/three-practical-examples-of-augmented-reality-applications-for-marketing/

[Imran et al., 2011] Imran, S. A., et Aouf, N. (2011). A recursive least squares solution for recovering robust planar homographies. In Towards Autonomous Robotic Systems (pp. 36-45). Springer Berlin Heidelberg.

[Kan et al., 2009] Kan, T. W., Teng, C. H., et Chou, W. S. (2009, December). Applying QR code in augmented reality applications. In Proceedings of the 8th International Conference on Virtual Reality Continuum and its Applications in Industry (pp. 253-257). ACM.

[Kato et al., 1999] Kato, H., et Billinghurst, M. (1999). Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on (pp. 85-94). IEEE.

[Kitchen et al., 1982] Kitchen L and Rosenfeld A, "Gray level corner detection", Pattern Recognition Letters, vol. 1, no. 2, pp.95-102, 1982.

[Kounavis et al., 2012] Kounavis, C. D., Kasimati, A. E., Zamani, E. D., et Giaglis, G. M. (2012). Enhancing the tourism experience through mobile augmented reality: Challenges and prospects. International Journal of Engineering Business Management, 4(10), 1-6.

[Larue et al., 2012] Larue, M., B. Michaut, et al. (2012). "La Réalité Augmentée: Avancées scientifiques et Réalisations techniques." Retrieved 31/01/2015, URL: http://tpe-la-realite-augmentee.e-monsite.com/pages/iv-applications-possibles/a-domaine-militaire.html.

[Lepetit, 2001] Lepetit, V.. Gestion des occultations on Réalité Augmentée. Thèse de Doctorat. Université Henri Poincaré, Nancy 1. 23 Mai 2001.

[Lepetit et al., 2006] Lepetit, V., et Fua, P.(2006).Keypoint recognition using randomized trees. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 28(9), 1465-1479.

[Leutenegger et al., 2011] Leutenegger, S., Chli, M., et Siegwart, R. Y. (2011, November). BRISK: Binary robust invariant scalable keypoints. In Computer Vision (ICCV), 2011 IEEE International Conference on (pp. 2548-2555). IEEE.

[Livingston et al., 2011] Livingston, M. A., Rosenblum, L. J., Brown, D. G., Schmidt, G. S., Julier, S. J., Baillot, Y., et Maassel, P. (2011). Military applications of augmented reality. In Handbook of augmented reality (pp. 671-706). Springer New York.

[Lowe, 1999] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In Computer vision, 1999. The proceedings of the seventh IEEE international conference on (Vol. 2, pp. 1150-1157). IEEE.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), 91-110.

[Mei et al., 2007] Mei, C. et Rives, P.(2007).Cartographie et localisation simultanée avec un capteur de vision. Journées Nationales de la Recherche en Robotique.

[Milgram et al., 1995] Milgram, P., Takemura, H., Utsumi, A., and Kishino, F., Augmented reality: A class of displays on the reality-virtuality continuum. In Photonics for Industrial Applications, pages 282–292. International Society for Optics and Photonics, 1995.

[Mikolajczyk et al., 2005] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 10, pp. 1615–1630, 2005.

[Mokhtarian et al., 1998] Mokhtarian F and Suomela R, “Robust image corner detection through curvature scale space”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 12, pp.1376-1381, 1998.

[Moravec, 1980] Moravec, H., (1980). "Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover". Tech Report CMU-RI-TR-3 Carnegie-Mellon University, Robotics Institute.

[Mouragnon et al., 2006] Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., et Sayd, P. (2006, August). Monocular vision based SLAM for mobile robots. In Pattern Recognition, 2006. ICPR 2006. 18th International Conference on (Vol. 3, pp. 1027-1031). IEEE.

[Naimark et al., 2002] Naimark L. and Foxlin E.. Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker. In ISMAR 2002: IEEE /ACM International Symposiumon Mixed and AugMented Reality, Darmstadt, Germany, Sept. 2002.

[Nasman et al., 2012] Nasman, J. D., et Cutler, B. (2012, March). Evaluation of a tangible interface for architectural daylighting analysis. In Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (pp. 207-207). ACM.

[Nistér, 2005] Nistér, D.(2005).Preemptive RANSAC for live structure and motion estimation. Machine Vision and Applications, 16(5), 321-329.

[Noguchi et al., 2012] Noguchi, A., et Yanai, K. (2012). A surf-based spatio-temporal feature for feature-fusion-based action recognition. In Trends and Topics in Computer Vision (pp. 153-167). Springer Berlin Heidelberg.

[Oberkampf et al., 1993] Oberkampf, D., DeMenthon, D. F., et Davis, L. S. (1993, June). Iterative pose estimation using coplanar points. In Computer Vision and Pattern Recognition,

1993. Proceedings CVPR'93., 1993 IEEE Computer Society Conference on (pp. 626-627). IEEE.

[Rabbi et al., 2013] Rabbi, I., et Ullah, S. (2013). A Survey on Augmented Reality Challenges and Tracking. Acta Graphica znanstveni časopis za tiskarstvo i grafičke komunikacije, 24(1-2), 29-46.

[Rekimoto et al., 2000] Rekimoto, J., et Ayatsuka, Y. (2000, April). CyberCode : designing augmented reality environments with visual tags. In Proceedings of DARE 2000 on Designing augmented reality environments (pp. 1-10). ACM.

[Rice et al., 2006] Rice, A. C., Beresford, A. R., et Harle, R. K. (2006, March). Cantag: an open source software toolkit for designing and deploying marker-based vision systems. In Pervasive Computing and Communications, 2006. PerCom 2006. Fourth Annual IEEE International Conference on (pp. 10-pp). IEEE.

[Rosten et al., 2006] Rosten, E., et Drummond, T. (2006). Machine learning for high-speed corner detection. In Computer Vision–ECCV 2006 (pp. 430-443). Springer Berlin Heidelberg.

[Rublee et al., 2011] Rublee, E., Rabaud, V., Konolige, K., et Bradski, G. (2011, November). ORB: an efficient alternative to SIFT or SURF. In Computer Vision (ICCV), 2011 IEEE International Conference on (pp. 2564-2571). IEEE.

[Simon et al., 2006] Simon, G., et Decollogne, J. (2006). Intégrer images réelles et images 3D-Post-production et réalité augmentée (p. 224). Dunod.

[Smith et al., 1997] Smith, S.M. and Brady, J. M., (May 1997). "SUSAN - a new approach to low level image processing". International Journal of Computer Vision (The SUSAN corner detector).

[Sutherland, 1965] Sutherland.I, (1965). The ultimate display. In In IFIPS Congress, volume 2, pages 506_508, New York, NY, USA.

[Sutherland, 1968] Sutherland.I, (1968). A head-mounted three dimensional display. Proceedings of the December 9-11, 1968, fall joint computer conference, part I, pages 757_764.

[Ta et al., 2009] Ta, D. N., Chen, W. C., Gelfand, N., et Pulli, K. (2009, June). Surftrac : Efficient tracking and continuous object recognition using local feature descriptors. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on (pp. 2937-2944). IEEE.

[Tola et al., 2010] Tola, E., Lepetit, V., et Fua, P. (2010). Daisy: An efficient dense descriptor applied to wide-baseline stereo. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 32(5), 815-830.

[Toscani, 1987] Toscani, G. (1987). Systemes de calibration et perception du mouvement en Vision Artificielle (Doctoral dissertation, Paris 11).

[Trajkovic et al., 1998] Trajkovic M and Hedley M, “Fast Corner Detection”, Image and Vision Computing, vol. 16, no. 2, pp.75-87, 1998.

[Tuytelaars et al., 2008] Tuytelaars, T., et Mikolajczyk, K.(2008). Local invariant feature detectors: a survey. Foundations and Trends® in Computer Graphics and Vision, 3(3), 177-280.

[Vigueras, 2007] Vigueras-Gomez, F. (2007). Système de réalité augmentée basé sur l'observation de structures planes: conception et évaluation (Doctoral dissertation, Université Henri Poincaré-Nancy I).

[Wagner et al., 2007] Wagner, D., et Schmalstieg, D. (2007). Artoolkitplus for pose tracking on mobile devices (pp. 139-146). na.

[Wagner et al., 2008] Wagner, D., Reitmayr, G., Mulloni, A., Drummond, T., et Schmalstieg, D. (2008, September). Pose tracking from natural features on mobile phones. In Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality (pp. 125-134). IEEE Computer Society.

[Zendjebil, 2010] Zendjebil, M. I. (2010). Localisation 3D basée sur une approche de suppléance multi-capteurs pour la réalité augmentée mobile en milieu extérieur (Doctoral dissertation, Evry-Val d'Essonne).

[Zhang, 2000] Zhang, Z. (2000). A flexible new technique for camera calibration. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 22(11), 1330-1334.