

THESE

Présentée à l'**Université de la Méditerranée**

Pour obtenir le grade de **Docteur**

Spécialité : **Informatique**

# Suivi d'objets à partir d'images issues de caméras mobiles non calibrées

---

Application à la reconstruction 3D de scènes sportives

Par : **Ludovic Lucia**

Date de soutenance de thèse : 14 février 2011, devant le jury composé de :

Dominique Faudot	Professeur à l'Université de Bourgogne	Rapporteur
Laurent Lucas	Professeur à l'Université de Reims	Rapporteur
Samir Akkouche	Professeur à l'Université Claude Bernard Lyon 1	Examineur
Marc Daniel	Professeur de l'Université de la Méditerranée	Examineur
Sébastien Mavromatis	Maître de conférences à l'Université de la Méditerranée	Codirecteur
Jean Sequeira	Professeur à l'Université de la Méditerranée	Codirecteur

*Doctorat Aix-Marseille Université*

*délivré par l'Université de la Méditerranée*

Je souhaite remercier le laboratoire LSIS qui m'a accueilli le temps de ma thèse et plus particulièrement : mes directeurs J. Sequeira et S. Mavromatis pour leur disponibilité, leurs encouragements et la justesse de leurs conseils. Aussi, je remercie H. Ripoll et B. Zoudji qui ont contribué à l'avancée de ma réflexion. Je n'oublie pas mes camarades J-C. Chambelland, G. Thibault, G. Operto, A. Nisan, M. Sérayet, C. Palmann et F. Graglia qui m'ont supporté tout le temps de ma thèse.

J'adresse également mes remerciements à l'agence SII d'Aix-en-Provence qui a financé mes travaux de recherche et notamment : J-P. Chevée et F. Ponsard pour la confiance qu'ils m'ont accordée ainsi que les collaborateurs S. Perrotte, E. Pillant, S. Bronner, J. Achard et B. Poulard qui ont contribué à mon intégration et au bon déroulement du projet SimulFoot.

Aussi, je remercie L. Baudoux pour son soutien permanent et ses précieuses relectures.

Enfin, je suis très reconnaissant envers D. Faudot, L. Lucas, S. Akkouche et M. Daniel pour l'honneur qu'ils m'accordent en acceptant d'évaluer mon travail de thèse.

# Tables des matières

Tables des figures .....	7
Introduction.....	9
Partie A. Etude et recherche .....	10
Chapitre I. Le suivi d'objets dans le cadre de la reconstruction 3D .....	11
1.1    La reconstruction 3D .....	11
1.2    Les techniques de suivi.....	15
1.2.1    Détection du mouvement .....	16
1.2.2    Conservation de l'identité .....	22
1.2.3    Conclusion .....	24
1.3    Cadre applicatif.....	25
1.3.1    Approche sépare et rassemble.....	25
1.3.2    Approche directe.....	27
1.3.3    Approche multicaméra.....	29
1.4    Conclusion .....	31
Chapitre II. Un référentiel commun aux images .....	32
2.1    Le processus de recalage.....	33
2.1.1    Extraction d'éléments caractéristiques.....	34
2.1.2    Mise en correspondance d'éléments caractéristiques .....	35
2.1.3    Conclusion .....	36
2.2    Mise en correspondance .....	37
2.2.1    Cas général .....	37
2.2.2    Mise en œuvre dans notre cadre applicatif .....	39
2.2.3    Résultats .....	45
2.3    Classification de droites en fonction de leurs points de fuite.....	45
2.3.1    Détection de points de fuite.....	47
2.3.2    Détection des faisceaux de droites .....	48
2.3.3    Résultats .....	53
2.4    Calcul des paramètres de la caméra .....	56
2.4.1    Calibrage de la caméra .....	56
2.4.2    Calcul des paramètres simplifiés de la caméra .....	56
2.4.3    Résultats .....	64
2.5    Protocole de test .....	66

2.5.1	Description et hypothèses.....	66
2.5.2	Résultats .....	68
2.5.3	Perspectives.....	71
2.6	Conclusion .....	71
Partie B. Développements et utilisation industrielle .....		73
Chapitre I. Valorisation de la recherche .....		74
1.1	Contexte .....	74
1.1.1	Historique, Laboratoire LSIS .....	74
1.1.2	SII, agence d'Aix-en-Provence .....	75
1.2	Transfert technologique .....	78
1.2.1	Etude de l'existant.....	78
1.2.2	Réalisation d'un prototype.....	80
Chapitre II. Maturation.....		85
2.1	Veille technologique.....	85
2.1.1	Logiciels orientés formation.....	85
2.1.2	Logiciels orientés « média » .....	87
2.2	Etude ergonomique.....	90
2.2.1	Introduction.....	90
2.2.2	De la représentation imagée à l'action .....	90
2.2.3	L'apprentissage par l'image.....	91
2.2.4	Conclusion .....	92
2.3	SimulFoot Entraînement .....	93
2.3.1	Scénarios dynamiques.....	94
2.3.2	Un outil adapté à l'entraîneur .....	95
Conclusion et perspectives.....		97
Annexe I. Soustraction de fond .....		98
1.1	Filtre médian temporel.....	98
1.2	Moyenne gaussienne .....	98
1.3	Mélange gaussien.....	99
Annexe II. Calcul du flot optique .....		101
2.1	Méthodes différentielles.....	101
2.1.1	Contrainte globale.....	102
2.1.2	Contrainte locale .....	103
2.1.3	Contrainte de contour et de surface.....	103

2.1.4	Contrainte multiple .....	104
2.2	Méthode par corrélation .....	105
2.3	Affinement du flot optique.....	106
2.3.1	Procédé hiérarchique .....	106
2.3.2	Raffinement temporel.....	107
	Bibliographie.....	109

## Tables des figures

Figure 1 – Reconstruction 3D en architecture .....	11
Figure 2 – Reconstruction 3D en médecine .....	12
Figure 3 – Reconstruction 3D dans le divertissement.....	13
Figure 4 – Reconstruction d'une scène de football.....	14
Figure 5 – Les facteurs impliqués dans la reconstruction 3D.....	14
Figure 6 – Modèle géométrique de caméra.....	16
Figure 7 – Représentation du mouvement apparent par un vecteur de déplacement.....	17
Figure 8 – Soustraction d'images et difficultés du choix du seuil .....	19
Figure 9 – Exemple de flot optique : image au temps t-1 (a), au temps t (b) et le flot optique résultant .....	19
Figure 10 – Hypothèses utilisées pour calculer le flot optique .....	20
Figure 11 – Représentation d'un "blob".....	23
Figure 12 – Modélisation du comportement des "blobs" lors d'une occultation (approche sépare & rassemble).....	24
Figure 13 – Modélisation du comportement des "blobs" lors d'une occultation (approche directe)..	24
Figure 14 – Représentation en graphe d'une séquence d'images .....	26
Figure 15 – Deux groupes formés de nœuds connexes .....	26
Figure 16 – Exemple de <b>ROI</b> et de ses différents <b>PI</b> .....	28
Figure 17 – Description du processus de suivi dans le cas d'occultations .....	29
Figure 18 – Le référentiel utilisé pour comparer deux images correspond au terrain de football.....	33
Figure 19 – Décomposition du processus de recalage .....	34
Figure 20 – Le recalage manuel consiste à sélectionner quatre amers dans le plan image (a) et leurs correspondants dans le plan modèle (b) pour obtenir la transformation liant ces deux plans (c).....	34
Figure 21 – Détection de droites dans une image .....	35
Figure 22 – Exemple de recalage : (a) plan image, (b) plan modèle .....	39
Figure 23 – Condition nécessaire à la mise en correspondance .....	40
Figure 24 – Détection de droites qui correspondent à des droites du modèle .....	41
Figure 25 – Recalage plan à plan.....	42
Figure 26 – Symétrie d'axe Ox.....	43
Figure 27 – Symétrie d'axe Oy.....	43
Figure 28 – Cas ambigu .....	44
Figure 29 – Points de fuite à distance finie et infinie .....	46
Figure 30 – La sphère de Gauss comme espace d'accumulation .....	47
Figure 31 – Application du théorème de Thalès .....	49
Figure 32 – Théorème de Ptolémée.....	50
Figure 33 – Exemple de faisceaux partageant une droite commune.....	51
Figure 34 – Justification de la propriété 2.....	52
Figure 35 – Détection de faisceaux de droites.....	54
Figure 36 – Exemple de détection de faisceaux de droites.....	55
Figure 37 – Paramètres d'observation choisis : <b>S</b> la position de la caméra, <b>M</b> le projeté de <b>C</b> dans le plan <b>xOy</b> et <b>θ</b> l'ouverture de la caméra .....	58
Figure 38 – Repère associé à l'écran .....	59
Figure 39 – Expression de <b>α</b> , l'angle <b>xM</b> , <b>Ms</b> , à partir de <b>M1</b> et <b>M3</b> .....	59

Figure 40 – Illustration des points $M_i$ après une rotation d'angle $-\alpha$ dans le repère $xMz$ .....	60
Figure 41 – Calibrage de caméra .....	64
Figure 42 – Reconstruction 3D d'une scène de football .....	65
Figure 43 – Exemple d'images utilisées par un entraîneur pour la formation. Ces dernières vérifient les hypothèses liées aux images (prise de vue, recouvrement conséquent et nombre de droites suffisant).....	67
Figure 44 – Exemple d'images de notre base de données : (a) (b) et (c) sont des images extraites de séquences sélectionnées par un expert, (a) et (b) représentent la même action de deux points de vue différents, (d) est une image choisie pour son point de vue pertinent .....	67
Figure 45 – Illustration des résultats de la mise en correspondance : (a) (c) sont des images sources et (b) (d) le modèle .....	69
Figure 46 – Illustration du critère de cohérence et de son utilisation .....	70
Figure 47 – Le groupe SII .....	76
Figure 48 – Secteurs d'activité et métiers des agences SII.....	76
Figure 49 – SII et la simulation .....	77
Figure 50 – Recalage du terrain.....	79
Figure 51 – Placement des joueurs .....	79
Figure 52 – Interface Homme-Machine du prototype de validation .....	82
Figure 53 – Illustration de l'assistant.....	83
Figure 54 – Résultats .....	83
Figure 55 – Illustration du logiciel Winner .....	85
Figure 56 – Illustration des outils TechFoot & TactFoot .....	86
Figure 57 – Illustration de l'IHM du logiciel Soccer-Trainer .....	86
Figure 58 – Illustration du logiciel X-Simul .....	87
Figure 59 – Illustration du logiciel Virtual Replay.....	88
Figure 60 – Illustration du logiciel 3DReplay.....	88
Figure 61 – Illustration du logiciel PIERO .....	89
Figure 62 – Reconstruction et visualisation 3D (sous Ogre3D) d'une séquence de jeu.....	90
Figure 63 – Situation imaginée.....	92
Figure 64 – Situation réelle .....	93
Figure 65 – Un scénario composé d'une étape initiale $t_0$ (a) et de deux autres étapes $t_1$ (b) et $t_2$ (c) .....	94
Figure 66 – Illustration des différentes animations (de gauche à droite) : marche, course rapide et course .....	95
Figure 67 – Illustration du logiciel SFE.....	96
Figure 68 – Extraction de l'arrière-plan par un filtre médian temporel.....	98
Figure 69 – Soustraction d'arrière-plan à l'aide d'une approche de type moyenne gaussienne.....	99
Figure 70 – Ligne de contrainte .....	102
Figure 71 – Problème de l'ouverture .....	102
Figure 72 – Vitesse estimée à l'aide d'une contrainte de contour .....	104
Figure 73 – Décalage détecté par une méthode par corrélation .....	105
Figure 74 – Voisinage de recherche .....	106
Figure 75 – Exemple 1D du problème lié aux déplacements inférieurs au pixel .....	106
Figure 76 – Procédé hiérarchique .....	107

## Introduction

Le travail présenté dans ce mémoire en vue d'être soutenu dans le cadre d'une thèse de doctorat correspond à la fois à une activité de recherche (avec des résultats originaux, et des publications les présentant), mais aussi à des développements importants (qui ont permis la réalisation d'un prototype), car il a été réalisé dans un environnement industriel (convention CIFRE avec la société SII). De plus, même si le domaine dans lequel les travaux de recherche ont été réalisés est celui de l'imagerie numérique, partie intégrante de l'informatique dans le sens où il est pris ici, il a fallu néanmoins aborder d'autres thèmes de manière assez approfondie tant sur les plans théorique (ergonomie cognitive) que technique (plates-formes de développement) et applicatif (finalisation du prototype), voire commercial (relations avec les utilisateurs du prototype développé). Ce contexte à la fois riche et pluridisciplinaire a engendré quelques contraintes quant à la rédaction de ce document, ne permettant pas une rédaction « linéaire » classique sous la forme « Etat de l'Art – Contribution théorique – Résultats ». Nous avons donc opté pour une structure un peu différente, en deux parties, la première s'inspirant du schéma classique évoqué précédemment et la deuxième faisant état des développements et de l'utilisation industrielle des résultats obtenus.

La finalité industrielle de ce travail est la réalisation d'un simulateur qui permette aux entraîneurs de football de mieux interpréter certaines situations de jeu à partir de séquences vidéo en proposant une vision 3D, mais aussi de présenter de la manière la plus ergonomique qui soit des schémas de jeux sur la base de ces situations réelles. De plus, contrairement à ce qui se pratique assez couramment dans ce cadre-là, nous avons souhaité – pour des raisons de souplesse d'utilisation – rester dans un cadre assez large et ne pas profiter d'une phase préalable (fastidieuse) de calibrage, ce qui a été à l'origine de verrous scientifiques et technologiques que nous avons cherché à résoudre, et pour lesquels nous proposons des solutions dans la première partie.

Les problèmes que nous avons eu à traiter sont de deux natures, informatiques d'une part et ergonomiques d'autre part. Il s'est agi, par exemple, de caractériser la transformation homographique permettant de passer de l'image au modèle, de retrouver la position de la caméra ou de déterminer automatiquement à quelle zone du terrain correspond l'image acquise. Et, au niveau ergonomique, il s'est agi de reproduire le plus fidèlement possible les modes d'appréhension de la réalité terrain, ainsi que de modélisation et de communication des entraîneurs de football.

La première partie est structurée en deux chapitres : le premier présente un état de l'art des méthodes permettant le suivi d'objets dans le cadre de la reconstruction 3D, et ce dans un contexte plus large que l'application proposée ; le second aborde, dans chacune de ses sections, les différents verrous rencontrés et les solutions proposées.

La deuxième partie, plus courte que la première, est également structurée en deux chapitres : le premier situe le travail dans son contexte industriel et le second présente la maturation du prototype développé, i.e. la manière dont la communication avec les utilisateurs a pu le faire évoluer (aspects ergonomiques).

# PARTIE A.

## Etude et recherche

---

### *Suivi d'objets appliqué à la reconstruction 3D de scènes sportives*

Dans cette partie nous détaillons le processus de suivi d'objets, appliqué à la reconstruction 3D, dans des séquences d'images. Après avoir exposé différentes techniques de suivi, nous montrons que celles-ci sont mal adaptées aux images issues de caméras dynamiques. Nous proposons donc une approche originale qui consiste à effectuer le suivi dans un contexte classique (de type caméras fixes) après avoir « plongé » ces images dans un référentiel commun (le modèle obtenu à partir de la connaissance sur la scène). Cette opération consiste à recalculer une image avec le modèle et, après avoir précisé les difficultés liées à l'automatisation de ce processus, nous présentons un schéma générique qui répond à cette problématique. De plus, nous détaillons son application dans notre contexte et notamment les trois verrous technologiques qui sont : l'extraction d'informations dans l'image (i.e la détection et la classification de droites en faisceaux), la mise en correspondance de ces droites avec celles du modèle et enfin le calibrage de la caméra à partir du calcul de la transformation entre l'image et le modèle. Nous proposons par la suite un protocole de test pour valider l'ensemble du processus de recalage dans notre cadre applicatif.

## Chapitre I. Le suivi d'objets dans le cadre de la reconstruction 3D

La reconstruction 3D est un processus qui consiste à reproduire virtuellement, via une représentation tridimensionnelle, une scène réelle. De nombreuses études ont été menées en vue de reconstruire des scènes statiques dans les domaines de l'architecture, de la médecine ou encore du divertissement. Mais, ce processus est plus complexe dans le cas de scènes dynamiques, i.e contenant des objets en mouvement. En effet, celui-ci implique la localisation des objets au cours du temps. Par exemple, déterminer la position des joueurs est une étape indispensable à la reconstruction 3D de scènes sportives. Aussi, décrire le mouvement de l'ensemble des objets d'une scène nécessite l'application de techniques de suivi ; opération qui a pour objectif la localisation spatio-temporelle d'objets à partir de séquences d'images. Le suivi est un problème ouvert, source de nombreux travaux. Les techniques de suivi récemment proposées dans la littérature sont peu ou prou inadaptées à la reconstruction 3D. En effet, ces dernières permettent d'obtenir la position des objets non pas dans le « repère monde », i.e le repère de la scène virtuelle, mais dans le « repère image ». Par conséquent, la reconstruction 3D de scènes dynamiques est, dans la pratique, fortement supervisée.

Dans la partie suivante, nous présentons plus en détail le processus de reconstruction 3D dans le cas de scènes statiques et dynamiques. Nous détaillons ensuite les techniques de suivi proposées dans la littérature ainsi que les difficultés majeures liées à la mise en œuvre de ces techniques dans notre cadre applicatif.

### 1.1 La reconstruction 3D

La reconstruction 3D est un axe de recherche en plein essor dans les domaines d'application aussi variés que l'architecture, la médecine ou encore dans le divertissement. En architecture l'objectif est principalement de créer à moindre coût une construction en lieu et place d'une maquette. La reconstruction 3D est alors utilisée lorsqu'un bâtiment déjà existant nécessite des réaménagements ou, dans un domaine plus proche de l'archéologie, pour visualiser ce à quoi ressemblait un vestige (Debevec, 2005) (Debevec, 1996) (Debevec, et al., 1996) (Debevec, et al., 1998).

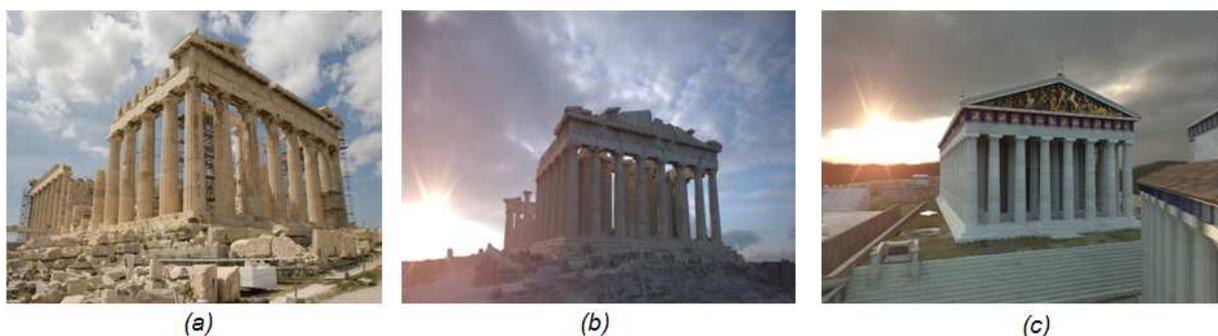


Figure 1 – Reconstruction 3D en architecture

La Figure 1 est composée d'une photographie du Parthénon<sup>1</sup> (a) ainsi que sa reconstruction 3D partielle, i.e « en l'état » (b), et totale (c). Ces images sont extraites des travaux réalisés par « USC Institute for Creative Technologies<sup>2</sup> ».

De même, la reconstruction 3D est fortement sollicitée dans le domaine de la médecine, notamment dans l'imagerie médicale. En effet, celle-ci a pour objectif de proposer au praticien (chirurgien, cardiologue, etc.) une représentation visuelle la plus « simple » possible et qui « condense » une information pertinente issue de mesures ; ces dernières étant acquises en fonction d'une modalité donnée (scanner, radiographie, IRM, etc.). Ainsi, la reconstruction 3D (d'un organe ou d'un tissu par exemple) est une représentation visuelle particulièrement pertinente. Un autre type d'application dans ce domaine consiste à créer un environnement virtuel pour simuler les conditions d'une opération (réalité virtuelle) et ainsi permettre la formation des futurs praticiens (Kühnapfel, et al., 1993) (Kühnapfel, et al., 1999) (Kühnapfel, et al., 2000).

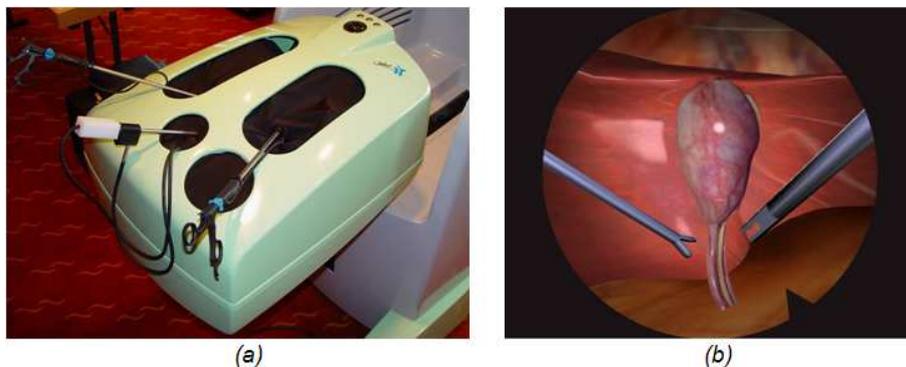


Figure 2 – Reconstruction 3D en médecine

La Figure 2 représente un système de réalité virtuelle appliqué à la médecine viscérale (endoscopie) qui se compose d'une interface haptique<sup>3</sup>, une boîte à instruments (a), et d'une visualisation de la scène (b). Ce système a été proposé par le groupe de recherche VR-Systems/Realtime-Simulation (KISMET<sup>4</sup>) et la société<sup>5</sup> « Select IT VEST Systems AG » / Bremen.

Les métiers du divertissement (notamment la télévision et le cinéma) utilisent également des techniques de reconstruction, principalement dans le but d'enrichir les images des flux vidéo qu'ils produisent (réalité augmentée). On peut citer comme exemple le sport à la télévision (Demeris, et al., 2001), des films ou encore des parcs d'attractions.

---

<sup>1</sup> Un édifice situé sur l'acropole d'Athéna

<sup>2</sup> <http://gl.ict.usc.edu/>

<sup>3</sup> Interfaces qui donnent des sensations par le toucher

<sup>4</sup> <http://www-kismet.iai.fzk.de/>

<sup>5</sup> <http://www.select-it.de/>



Figure 3 – Reconstruction 3D dans le divertissement

La Figure 3 représente différentes utilisations de techniques de reconstruction appliquées aux métiers du divertissement. L'image (a) est tirée du film « Matrix Reloaded » ; film dont les effets visuels ont été réalisés et supervisés en partie par S. Ceretti (BUF Compagnie<sup>6</sup>) qui a déclaré : « *Il fallait désormais être capable de filmer une séquence, de la reconstruire en 3D dans tous ses détails et de la transformer en code par la suite. Cette approche permettait aux réalisateurs de faire bouger leur caméra comme ils le souhaitent dans un monde vu entièrement en code* ». L'image (b) est une illustration de l'attraction « les animaux du futur<sup>7</sup> » qui propose de restituer un environnement mi-réel, mi-virtuel à partir de jumelles. Ces dernières embarquent une caméra qui filme le décor réel observé par le spectateur. Un ordinateur incruste ensuite des animaux virtuels en 3D sur les images capturées par cette caméra.

Les classes de problèmes soulevés par la reconstruction 3D dépendent grandement du type d'application et des objectifs fixés par celle-ci (visualisation réaliste, mesures, etc.). En effet, ces difficultés sont fonction de la connaissance liée à l'environnement reconstruit – la scène est elle « fixe » ou bien dynamique ? Existe-t-il un modèle de la scène et de ses éléments ? La prise de vue est-elle une donnée connue ? – mais aussi en fonction de la qualité du rendu souhaitée et du degré d'automatisation attendu.

Les difficultés liées à la reconstruction 3D de scènes sportives sont principalement le dynamisme de la scène et les conditions de prise de vue (Grau, et al., 2005) (Grau, et al., 2006). En effet, une scène sportive est composée d'un ensemble d'objets (les joueurs) qui sont en mouvement au cours du temps. Ainsi, déterminer le mouvement de ces joueurs est une étape indispensable à cette reconstruction. Par exemple, Le logiciel édité par SportsTec propose la reconstruction (2D) d'une séquence de basketball à partir de « données terrain ». Un ou plusieurs utilisateurs ont à disposition une « tablette » qui représente un terrain de basketball et ils indiquent en temps réel les déplacements des joueurs. Aussi, ceux-ci sont saisis manuellement et aucun processus de suivi de détection des mouvements des joueurs n'est utilisé. Ce procédé entraîne inévitablement un manque de précisions concernant les positions des joueurs, cette dernière étant liée à l'appréciation de l'utilisateur. D'autres logiciels, par exemple 3DReplay<sup>8</sup> ou encore PIERO<sup>9</sup>, proposent la reconstruction 3D d'un moment clef d'une scène de football ; reconstruction réalisée à partir de séquences

<sup>6</sup> <http://www.buf.com/main.php>

<sup>7</sup> <http://www.lesanimauxdufutur.com/>

<sup>8</sup> <http://www.orad.tv/>

<sup>9</sup> <http://www.redbeemedia.com/piero/>

d'images. Or, celle-ci est possible uniquement si les paramètres de la caméra sont connus ou estimés. De plus, si la caméra est mobile, il est indispensable de recalculer ses paramètres au cours du temps.

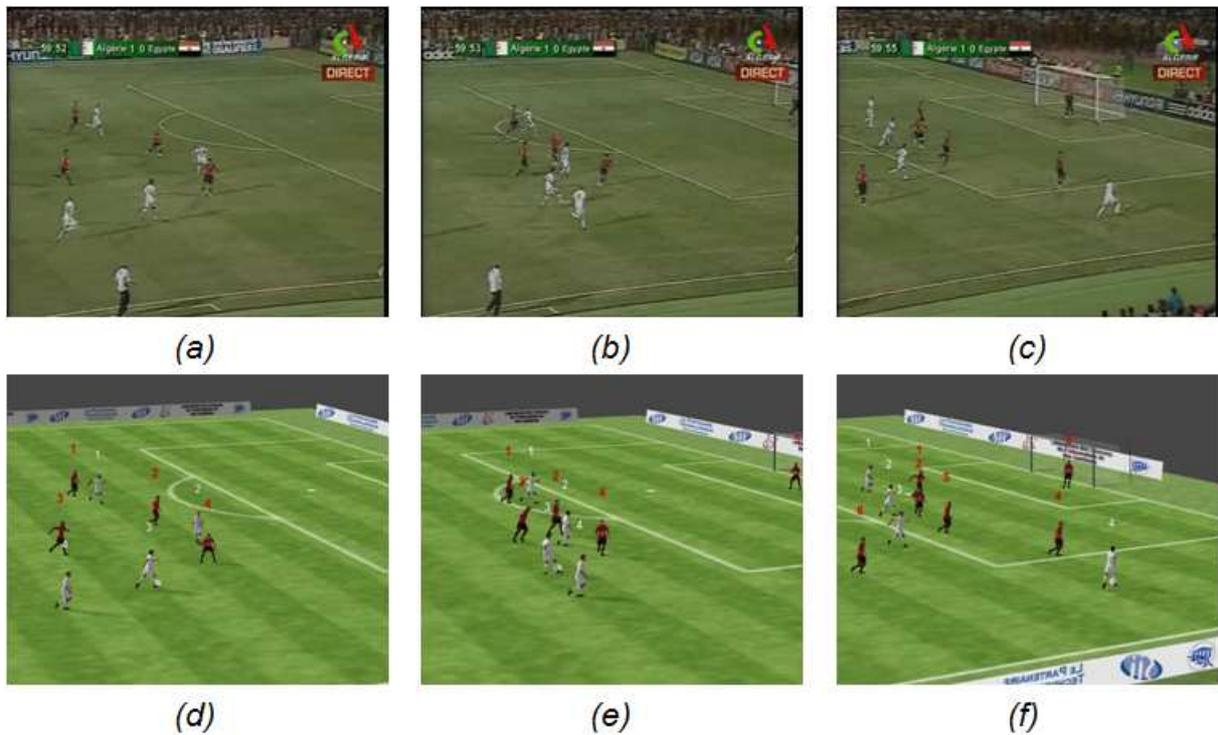


Figure 4 – Reconstruction d'une scène de football

La Figure 4 représente trois images (a) (b) et (c) qui correspondent à des moments clés d'une séquence de football. Les images (d) (e) et (f) sont les reconstructions 3D respectives de ces images. On remarque que les objets de la scène (les joueurs) sont en mouvements et il en est de même pour le point de vue.

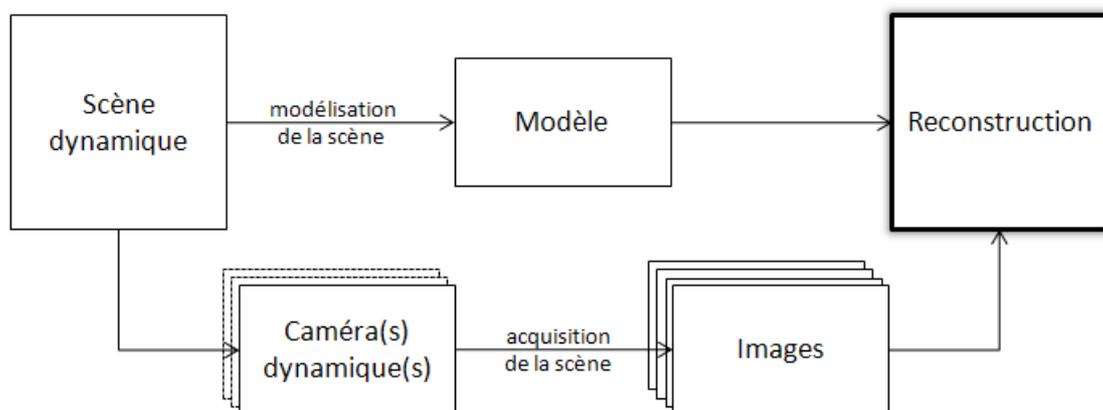


Figure 5 – Les facteurs impliqués dans la reconstruction 3D

La reconstruction 3D de scènes sportives est un processus complexe (Figure 5) qui implique la résolution de nombreux problèmes tels que la détection du mouvement dans une image, le calcul des paramètres d'une caméra, etc. Néanmoins, la connaissance a priori d'une scène sportive peut être modélisée et mise à contribution pour guider la reconstruction. Par exemple, une scène de

tennis est principalement composée d'un court et ce dernier à certaines propriétés invariantes telles que ses dimensions ou encore ses tracés au sol (lignes de fonds, lignes de services, etc.). Par conséquent, une image de cette scène contient ces informations et leur extraction simplifie le processus de reconstruction : un joueur qui n'est entouré d'aucune ligne est en dehors du court.

La problématique liée à la reconstruction 3D de scènes sportives peut alors s'énoncer sous la forme suivante : détecter et suivre les changements qui s'opèrent dans une scène dynamique à partir d'images issues de caméras faisant l'acquisition de cette scène ; les paramètres de ces caméras étant inconnus et pouvant varier dans le temps.

La détection et le suivi du mouvement sont des problèmes cruciaux en traitement d'images qui ont donné lieu à de nombreuses publications. Ainsi, après avoir détaillé ce processus et les techniques de suivi proposées dans la littérature, nous montrons que la mise en œuvre de ces techniques est particulièrement délicate, notamment dans le cas d'images issues de caméras mobiles non calibrées.

## 1.2 Les techniques de suivi

Le processus de suivi a pour but la localisation dans le temps et l'espace des objets en mouvement à partir de séquences d'images issues d'une ou de plusieurs caméras. Il est particulièrement utilisé dans le domaine de la vision assistée par ordinateur (étude du comportement, aide à la prise de décisions, reconstruction 3D, etc.). Les problèmes soulevés par les méthodes de suivi sont divers selon le domaine d'application et plus précisément selon la scène et le type d'acquisition. En effet, les techniques de suivi sont nécessairement plus ou moins complexes selon les caractéristiques des objets de la scène (rigide ou non par exemple) et de leurs « comportements » (occultations, etc.), mais aussi selon les images capturées (type de caméra, paramètres connus ou non, caméra fixe ou mobile, etc.).

Notre cadre applicatif, la reconstruction 3D de scènes sportives (et plus particulièrement de football), nécessite le suivi d'objets non rigides (des joueurs de football) à partir de séquences d'images issues d'une ou de plusieurs caméras mobiles dont les paramètres ne sont pas connus. Le suivi humain est un domaine de recherche qui a récemment donné lieu à de nombreux articles. Cet engouement s'explique principalement par le nombre grandissant d'applications commerciales (la surveillance routière, l'aide à la prise de médicaments, etc.) qui nécessitent notamment du suivi « temps réel » ; opération rendue possible par l'augmentation de la puissance de calcul des ordinateurs. Ainsi, de nombreux auteurs (Aggarwal, et al., 1999) (Gavrila, 1999) (Moeslund, et al., 2001) (Moeslund, et al., 2006) (Wang, et al., 2003) (Rowe, 2008) se sont intéressés aux différentes approches traitant du processus de suivi humain. Ces études ont permis l'émergence d'une méthodologie ; ce processus est alors le plus souvent accompli en deux temps : la détection du mouvement puis la conservation de l'identité. La première étape consiste à extraire de l'image les objets et leurs mouvements et correspond à une étape de segmentation<sup>10</sup>. La seconde, vise à conserver l'identité des objets d'une image sur l'autre. Une dernière étape peut également être ajoutée pour vérifier la cohérence des

---

<sup>10</sup> Opération qui consiste à extraire d'une image des primitives géométriques, le plus souvent des segments (contours) ou les surfaces (régions).

résultats obtenus. Celle-ci est guidée par la connaissance inhérente au type d'objet suivi et correspond à la compréhension du comportement humain.

## 1.2.1 Détection du mouvement

### 1.2.1.1 Prérequis

La compréhension du processus de formation d'une image, et plus particulièrement d'une séquence d'images acquise par une caméra, est essentielle à l'étude du mouvement.

#### Modèle géométrique de caméra

Le modèle géométrique de caméra le plus couramment utilisé pour décrire le processus de formation d'une image (Davies, 1997) est le modèle sténopé (*pin-hole*) ou modèle « trou d'aiguille » (Figure 6). Il existe de nombreux autres systèmes tels que les objectifs grands angles (par exemple de type *fisheye*) ou encore les objectifs à décentrement. Néanmoins, le modèle sténopé est le dispositif le plus simple et permet d'explicitier le processus d'acquisition et les différents mécanismes mis en œuvre.

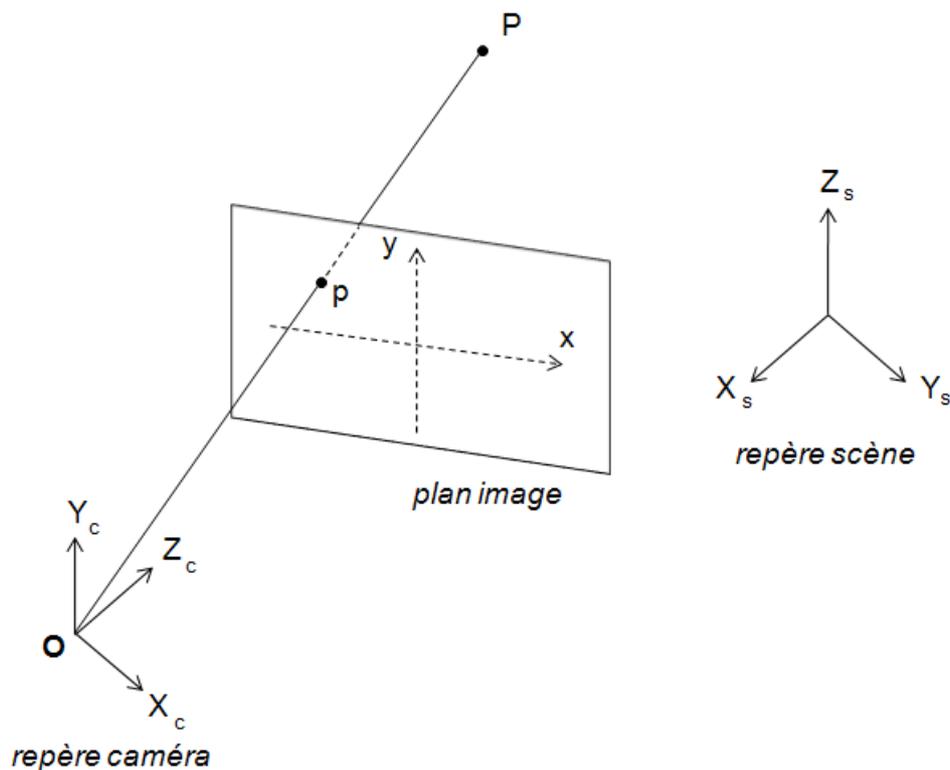


Figure 6 – Modèle géométrique de caméra

Le modèle sténopé est constitué d'un plan image et du centre optique  $O$ . Un point  $P$  de la scène, de coordonnée  $(X, Y, Z)$ , se projette sur le plan en un point  $p$ , de coordonnée  $(x, y)$ , qui est l'intersection de la droite  $(OP)$  avec le plan image. Cette projection est représentée, en coordonnées homogènes, par l'équation suivante :

$$\begin{pmatrix} x & y & 1 \end{pmatrix}^T = \mathbf{M} \begin{pmatrix} X & Y & Z & 1 \end{pmatrix}^T$$

avec  $M$  la matrice de projection perspective liée aux paramètres (interne et externe) de la caméra.

### Modélisation du mouvement

Évaluer le mouvement d'un objet à partir d'une séquence d'images revient à décrire la transformation que subit cet objet d'une image sur l'autre. Or, cette estimation correspond à un problème « mal posé ». En effet, le mouvement d'un objet dans une scène, i.e le « mouvement réel », est un déplacement 3D alors que les images sont des projections d'une scène 3D dans un plan 2D. Le mouvement détecté dans l'image, i.e le « mouvement apparent » correspond donc au mouvement réel observé via un système projectif. Il est le plus souvent modélisé par une translation et représenté par un vecteur de déplacement (Figure 7).

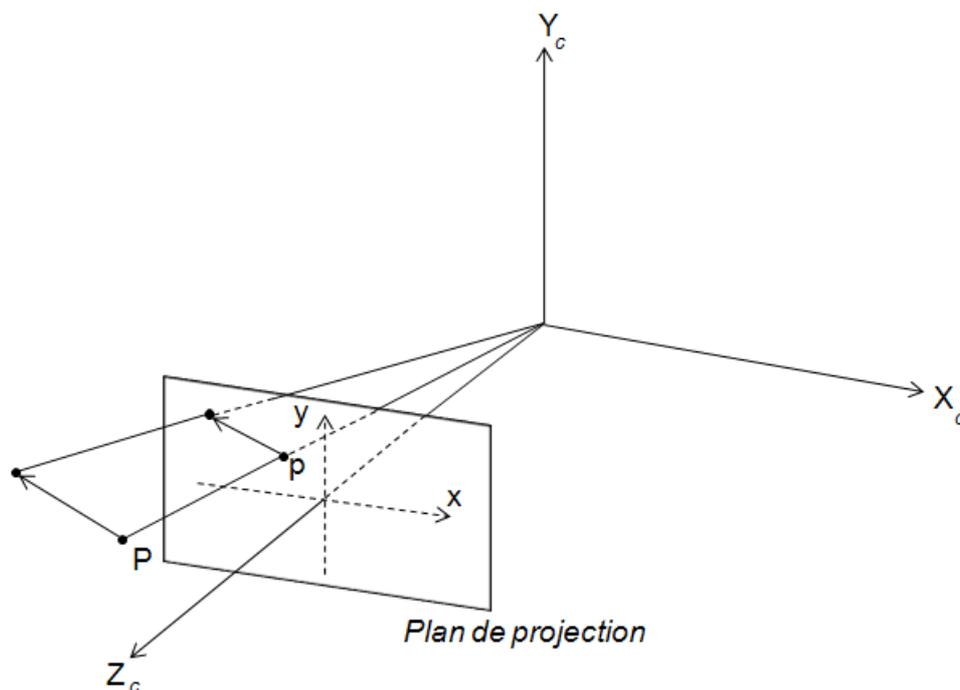


Figure 7 – Représentation du mouvement apparent par un vecteur de déplacement

La détection du mouvement correspond alors non pas à la détection du mouvement réel, mais à la détection du mouvement apparent. De plus, ce dernier décrit simultanément le mouvement des objets de la scène et celui de la caméra. De ce fait, il est indispensable de tenir compte du mouvement de la caméra afin de retrouver le mouvement réel des objets de la scène.

Le mouvement apparent est calculé le plus souvent via l'application de méthodes directes (Irani, et al., 1999) ou indirectes (Torr, et al., 1999). Les premières utilisent les valeurs pixels des images d'une même séquence pour calculer le mouvement apparent alors que les secondes s'appuient sur l'extraction et la mise en correspondance d'éléments caractéristiques des objets (points, droites, contours, etc.).

#### 1.2.1.2 Méthode directe

Les méthodes directes utilisent les valeurs des pixels des images d'une même séquence pour extraire le mouvement (Irani, et al., 1999). Cette opération est le plus souvent réalisée à l'aide d'approches

de type soustraction de fond (Piccardi, 2004) (Wren, et al., 1997) (Stauffer, et al., 1999) (Lo, et al., 2001) ou encore par le calcul du « flot optique » (Black, 1992) (Barron, et al., 1994) (Beauchemin, et al., 1995).

### Soustraction de fond

Dans le cas de caméras statiques, le mouvement apparent correspond seulement au mouvement des objets d'une scène. Les éléments contenus dans une image sont alors classés en deux catégories : l'arrière-plan (les objets immobiles) et le premier plan (les objets en mouvement).

Le problème de détection de mouvement peut alors être ramené au problème suivant : étant donné une séquence d'images acquise par une caméra statique, quels sont les éléments qui appartiennent au premier plan ? Une approche classique pour déterminer cette appartenance consiste à définir une « image de référence » (qui est alors la représentation de l'arrière-plan). Ainsi, les éléments qui appartiennent au premier plan sont détectés par soustraction entre l'image courante – l'image de travail – et l'image de référence :

$$|I(x, t) - I_r(x, t)| > \theta$$

Avec  $I(x, t)$  et  $I_r(x, t)$  respectivement l'intensité de l'image courante et de celle de référence à la coordonnée  $x = (x, y)$  et  $\theta$  un seuil fixé au préalable.

Les problèmes soulevés par ce type d'approche sont donc principalement la détermination de l'image de référence ainsi que le calcul de la valeur du seuil. Une approche intuitive pour déterminer l'image de référence consiste à calculer la différence entre des images successives :

$$|I(x, t) - I(x, t - 1)| > \theta$$

Dans ce cas, l'image de référence correspond à l'image précédente. Cette solution est très sensible à la valeur choisie pour le seuil (Figure 8).

Différentes approches ont alors été proposées dans la littérature pour éviter les problèmes soulevés par la détermination de l'image de référence et par le choix du seuil (cf. Annexe I. Soustraction de fond) ; approches détaillées par M. Piccardi (Piccardi, 2004). Ces dernières permettent la détermination du mouvement des objets contenus dans des séquences d'images issues de caméras fixes. Mais, ces méthodes sont difficiles à mettre en œuvre dans le cas d'images issues de caméras mobiles ; difficultés qui résultent de la détection de l'arrière-plan.

Néanmoins, caractériser les objets immobiles contenus dans une image comme appartenant à l'arrière-plan et ceux mobiles au premier plan est une idée intéressante de par son intuitivité. De plus, les arrières plans d'images issues de caméras filmant une scène sportive correspondent en majorité à un terrain de jeu. Aussi, cette connaissance peut être mise à contribution pour rendre possible l'extraction des arrières plans, notamment dans le cas d'images issues de caméras mobiles. Mais, ces arrières plans sont alors « en mouvement » et l'hypothèse que les objets en mouvement appartiennent uniquement au premier plan est invalidée. Dans ce cas, « plonger » une séquence d'image dans un référentiel commun – et immobile – est une solution envisageable ; ce référentiel tenant alors un rôle similaire à l'arrière-plan.

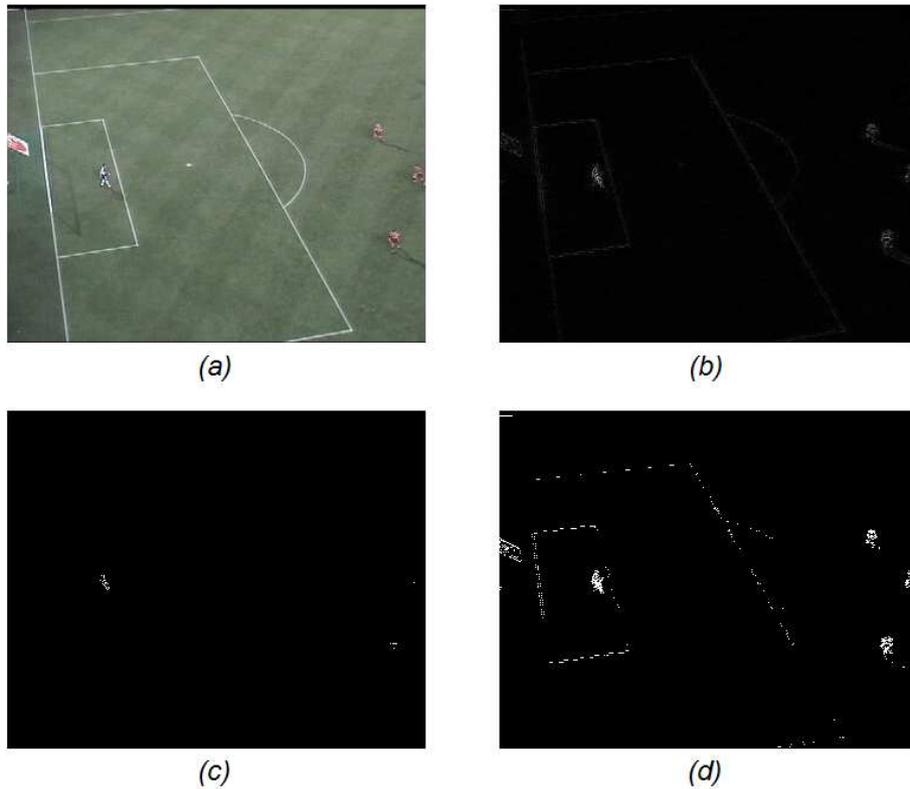


Figure 8 – Soustraction d’images et difficultés du choix du seuil

La Figure 8 représente une image de travail (a), la différence entre cette image et l’image précédente (b), le résultat avec un seuillage trop élevé (c) et trop bas (d).

### Flot optique

Pour déterminer le mouvement des objets – à partir d’une séquence d’image – une approche fortement plébiscitée dans la littérature consiste à calculer les flots optiques de chacune de ces images (Horn, et al., 1981) (Lucas, et al., 1981) (Lucas, 1985) (Black, 1992) (Barron, et al., 1994) (Beauchemin, et al., 1995).

**Définition (flot optique)** – Le flot optique d’une image est défini comme le champ de vecteurs qui décrit le mouvement apparent de cette image

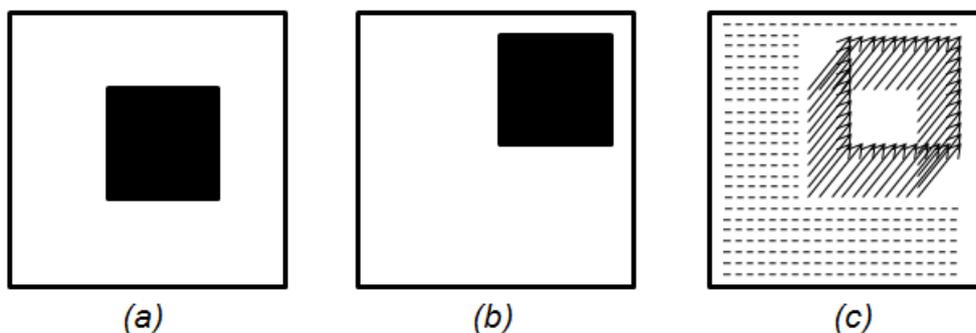


Figure 9 – Exemple de flot optique : image au temps t-1 (a), au temps t (b) et le flot optique résultant

Le calcul du flot optique d’une image permet d’observer le mouvement des objets d’une scène. En effet, le flot optique d’une image décrit le mouvement apparent, i.e le mouvement des objets de la

scène, mais aussi le mouvement de la caméra elle-même. Ainsi, le mouvement réel des objets est déduit du mouvement apparent et de celui de la caméra (si les paramètres de calibrage de la caméra sont connus, ou bien estimés au cours du temps). Néanmoins, le calcul du flot optique nécessite un certain nombre de suppositions ; ces dernières découlent directement des propriétés de la scène observée. Les trois hypothèses principalement utilisées ont été décrites par M. J. Black (Black, 1992) et correspondent à : la conservation des données, la cohérence spatiale et la continuité temporelle.

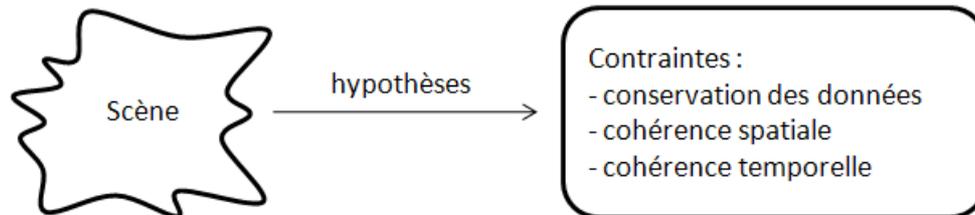


Figure 10 – Hypothèses utilisées pour calculer le flot optique

La conservation des données est une hypothèse fondamentale sur laquelle s'appuient la plupart des méthodes de calcul du flot optique (cf. Annexe II. Calcul du flot optique) et peut être formulée par :

**Définition (conservation des données)** – Les mesures (telle que l'intensité) faites sur une partie de l'image restent identiques dans l'espace et le temps.

L'hypothèse de cohérence des données, utilisée seule, ne permet pas la détermination exacte du flot optique. Le plus souvent, une contrainte de cohérence spatiale est conjointement utilisée et cette dernière est définie par :

**Définition (cohérence spatiale)** – Les points voisins dans une scène appartiennent typiquement à la même surface et par conséquent ont des vitesses similaires. Les points voisins dans une scène étant le plus souvent projetés en points voisins sur le plan image, le flot optique peut alors être considéré comme peu variant.

Les deux précédentes hypothèses sont habituellement employées pour calculer le flot optique entre deux images consécutives (cf. Annexe II. Calcul du flot optique). Une hypothèse également exploitée concerne la persistance temporelle et s'exprime sous la forme suivante :

**Définition (continuité temporelle)** – Le mouvement d'une surface, observée dans une séquence d'images, change graduellement dans le temps.

Les hypothèses formulées ci-dessus sont nécessaires au calcul du flot optique, or, ces dernières s'appuient sur une représentation de la scène qui est par nature une idéalisation. Par conséquent, ces hypothèses ne sont pas « entièrement » respectées dans la pratique. En effet, certains phénomènes – tels que l'apparition ou encore la disparition – peuvent invalider ces hypothèses.

**Définition (disparition)** – la disparition, ou encore l'occultation, est un phénomène de recouvrement apparent d'un objet par un autre.

**Définition (apparition)** – l'apparition est le phénomène inverse de la disparition, i.e. un objet, précédemment « invisible » devient visible.

L'hypothèse de conservation des données peut ne pas être vérifiée, notamment pour un certain nombre de situations d'illumination. On peut citer comme exemple les réflexions spéculaires, qui ne « bougent pas » nécessairement avec la surface, les ombres qui peuvent changer l'apparence d'une région ou encore la variation d'illumination (notamment pour les scènes d'extérieur). De plus, le bruit lié à l'acquisition de la séquence d'images peut également être une source d'erreurs. De même, le type d'objets observés dans la scène peut invalider l'hypothèse de conservation des données. En effet, cette dernière s'appuie sur le fait que la région d'une image est localement invariante aux déformations, ce qui peut être faux, notamment dans le cas d'objets non rigides. L'observation de multiples mouvements dans la séquence d'images peut également entraîner un certain nombre de cas problématiques tels que ceux décrits précédemment (occultation, disparition, etc.). Une autre source d'ambiguïté est soulevée quand la valeur d'intensité d'un point n'est pas déterminée par une surface unique, mais par un ensemble de surfaces – translucides ou réfléchives – comme une scène observée au travers d'une fenêtre sale.

La cohérence spatiale s'exprime sous différentes formes et la plus commune est de supposer que le flot optique est constant dans une région donnée. Toutefois, cette hypothèse est fautive dans le cas de discontinuités, notamment lors d'occultations (partielles ou totales).

L'hypothèse de continuité temporelle est une contrainte puissante, mais qui ne peut être valide que sur de courtes durées. De plus, le mouvement d'un objet ne peut être complètement prédit, il peut s'arrêter, changer brusquement de direction, etc. L'application stricte de cette hypothèse entraîne donc un certain nombre d'erreurs, notamment lorsque le mouvement décrit par un objet change rapidement. Cette hypothèse est également mise en défaut lors d'occultations. En effet, une surface persistante dans le temps disparaît subitement lors d'une occultation. L'apparition pose un problème similaire puisqu'une nouvelle surface devient visible et qu'aucune information n'était précédemment connue quant à son mouvement.

Il est nécessaire, pour garantir la stabilité du calcul du flot optique, de s'assurer que ces différentes hypothèses soient vérifiées dans leur contexte d'application. Par exemple, l'hypothèse de la continuité temporelle – supposition implicite au calcul du flot optique – est une source d'erreurs non négligeable dans notre cadre applicatif ; notamment lorsqu'un joueur se déplace rapidement alors que généralement les images sont acquises « plus lentement » que ce mouvement. De plus, le flot optique résulte du mouvement de la caméra et du mouvement des objets ; ce qui est problématique dans le cas d'images issues de caméras mobiles non calibrées.

### ***1.2.1.3 Méthode indirecte***

Pour estimer le mouvement des objets contenus dans une image, les méthodes indirectes (Tomasi, et al., 1991) (Shi, et al., 1994) (Rowe, 1996) (Torr, et al., 1999) utilisent les éléments caractéristiques de ces objets ; mouvement alors obtenu par la mise en correspondance de ces éléments d'une image sur l'autre. Ces méthodes procèdent en deux étapes :

- l'extraction d'éléments caractéristiques dans l'image,
- la mise en correspondance de ces éléments d'une image sur l'autre.

Ainsi, les éléments caractéristiques de l'image doivent respecter un certain nombre de propriétés, principalement d'invariance (Beaudet, 1978) à la transformation géométrique (liant deux images consécutives), pour être ensuite mis en correspondance.

Le plus souvent, les éléments extraits dans l'image sont des points et correspondent alors à des « coins ». Pour plus de détails sur les méthodes d'extraction de points, on se référera aux travaux de R. Deriche et G. Giraudon (Deriche, et al., 1990). Ces derniers ont classé les méthodes de détection de « points anguleux » en deux catégories principales:

- les méthodes basées sur les segments (Asada, et al., 1986) (Mokhtarian, et al., 1986) (Medioni, et al., 1987) (Deriche, et al., 1990),
- celles basées sur l'intensité (Morevec, 1977) (Kitchen, et al., 1982) (Dreschler, et al., 1982) (Zuniga, et al., 1983) (Nagel, 1983) (Shah, et al., 1984) (Noble, 1988) (Harris, et al., 1988) (Montesinos, et al., 1998) (Gouet, et al., 2002).

D'autres éléments peuvent également être recherchés dans l'image tels que les segments et les droites ou encore les contours d'un objet. L'extraction de tels éléments nécessite alors des traitements dédiés. On peut citer par exemple la détection de droites de type Hough (Duda, et al., 1972) ou encore l'extraction de contours par des approches de type contours actifs (Kass, et al., 1987).

Déterminer le mouvement des objets dans une séquence d'images par une méthode indirecte nécessite l'extraction d'éléments caractéristiques – et donc la résolution des problèmes soulevés par cette opération – mais aussi la mise en correspondance de ces éléments d'une image sur l'autre. Cette opération entraîne inévitablement un certain nombre de difficultés supplémentaires. Le plus souvent, ces méthodes utilisent un ou plusieurs critères pour mesurer la similarité entre deux éléments, tels que :

- une distance (de Manhattan, euclidienne, ou encore la distance de Mahalanobis) ;
- une corrélation ; classée en cinq familles par S. Chambon (Chambon, 2005) : la famille croisée, classique, dérivée, non paramétrique et la famille robuste) ;
- une méthode statistique : Espérance-Maximisation (Dempster, et al., 1997), K-Means (Hartigan, et al., 1979), Ransac (Fischler, et al., 1981), etc.

Les méthodes indirectes de détection du mouvement supposent l'extraction et la mise en correspondance d'éléments caractéristiques ; processus délicat, notamment lorsque deux images consécutives – et les éléments qui les composent – sont hétérogènes. En effet, ces méthodes nécessitent divers « réglages » (taille du voisinage, seuils, etc.) et ces derniers peuvent être très variables d'une image à l'autre. Par conséquent, déterminer automatiquement ces réglages est fortement souhaitable, mais représente une difficulté supplémentaire.

### **1.2.2 Conservation de l'identité**

Le suivi des objets le long d'une séquence d'images est réalisé en deux étapes : l'extraction du mouvement de ces objets et la conservation de leur identité. Cette dernière étape correspond à un

problème de classification pouvant être synthétisé par la question suivante : *comment caractériser et identifier un objet d'une image sur l'autre, notamment dans le cas d'occultations*<sup>11</sup> ?

Le plus souvent, les objets sont caractérisés par une série d'attributs telle que la position, la vitesse ou encore l'apparence. Ainsi, P. Gabriel *et al.* (Gabriel, et al., 2003) définissent un « blob » comme un ensemble d'objets (Figure 11). La nature de ces objets est peu importante (elle englobe des personnes, des voitures, etc.) et un blob peut également être un ensemble de blobs.

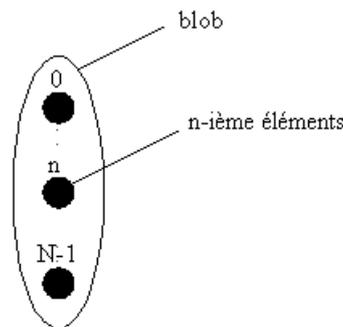


Figure 11 – Représentation d'un "blob"

P. Gabriel *et al.* (Gabriel, et al., 2003) ont alors déclinés deux types d'approches pour appréhender le problème de conservation de l'identité dans des cas complexes : les approches « sépare et rassemble » (*split & merge*) et les approches directes (*straight & through*).

De plus, l'utilisation de caméras multiples (filmant la même scène de différents points de vue) aide à résoudre le suivi d'objets dans le cas d'occultations. En effet, il suffit qu'un objet soit encore visible par une des caméras pour qu'il puisse être suivi. En général, le suivi d'objets est réalisé indépendamment pour chacune des caméras. Ensuite, les résultats obtenus sont fusionnés ; fusion possible uniquement si la relation entre chacune de ces caméras est connue. Or, cette donnée ne l'est pas nécessairement et il est alors indispensable de retrouver la relation entre ces images ; via le calcul des paramètres des caméras par exemple.

### 1.2.2.1 Approche sépare et rassemble

Dans le cas d'occultations, les objets qui s'intersectent sont fusionnés en groupe et ils forment alors une seule « entité ». Cette entité est ensuite divisée en plusieurs objets (correspondant aux objets initiaux) dès que ceux-ci se séparent (Figure 12). La principale difficulté soulevée par cette approche est donc de rétablir l'identité des objets lorsqu'ils se divisent.

---

<sup>11</sup> Un objet subit une occultation lorsqu'il est « recouvert » par un autre objet de façon partielle ou totale

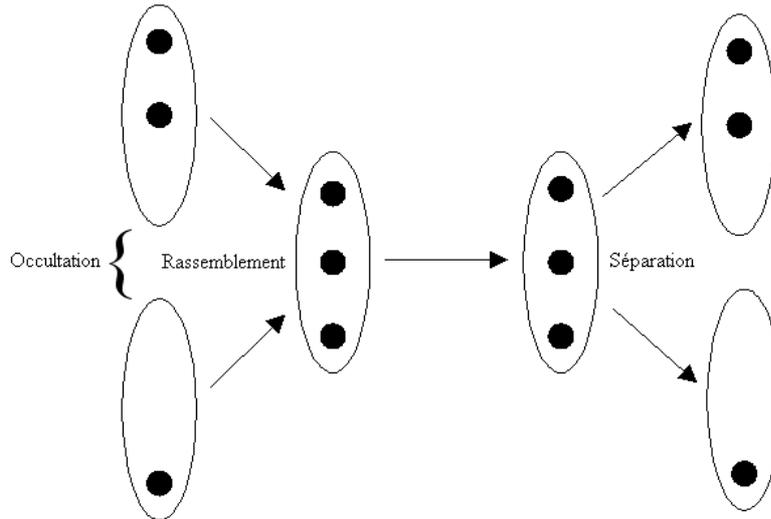


Figure 12 – Modélisation du comportement des "blobs" lors d'une occultation (approche sépare & rassemble)

### 1.2.2.2 Approche directe

Les approches directes ont pour principe d'effectuer le suivi d'objets de façon individuelle sans aucune fusion (Figure 13). Ces techniques s'appuient le plus souvent sur les caractéristiques des pixels qui représentent un objet de la scène. Ces caractéristiques peuvent être multiples selon les approches envisagées (le plus souvent le calcul de la profondeur relative de l'objet suivi). Ce type d'approche a pour avantage de maintenir l'identité de chaque objet tout au long du suivi. Mais, les problèmes qu'elle soulève concernent principalement l'attribution des « pixels partagés » entre plusieurs objets.

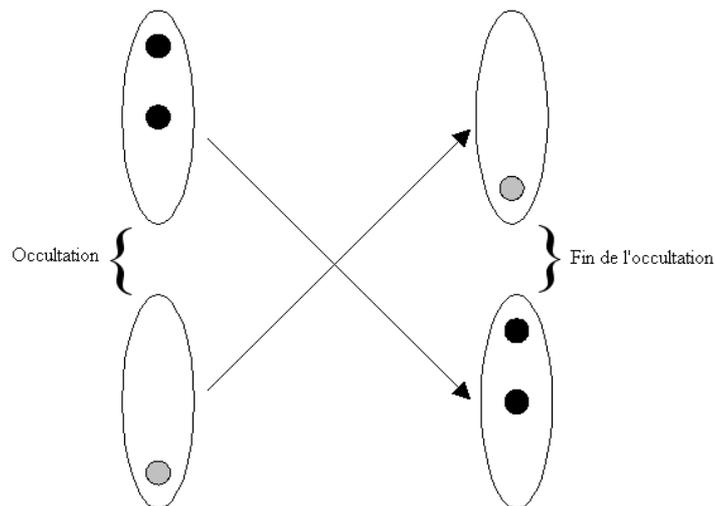


Figure 13 – Modélisation du comportement des "blobs" lors d'une occultation (approche directe)

### 1.2.3 Conclusion

Le suivi d'objets dans une séquence d'image est généralement réalisé en deux étapes : l'extraction de leurs mouvements et la conservation de leurs identités au cours du temps. L'identité de ces objets

est généralement conservée d'une image sur l'autre via l'application de méthodes directes, de type « sépare & rassemble » ou encore multicaméra. L'étape d'extraction du mouvement est quant à elle réalisée via l'application de méthodes directes ou indirectes. Or, ces méthodes détectent le mouvement apparent de l'image – le mouvement conjugué de la caméra et des objets – ce qui est problématique dans le cas d'images issues de caméras mobiles. Néanmoins, de nombreux travaux s'appuient sur ces méthodes pour proposer des solutions partielles au problème de suivi. Elles abordent par exemple le suivi dans des cas bien spécifiques (caméra fixe, paramètres de calibrage connus, etc.). Dans la partie suivante nous présentons ces approches et notamment celles appliquées dans un contexte proche du nôtre (scènes de football).

## 1.3 Cadre applicatif

Des travaux sur le suivi d'objets ont récemment été proposés (Iwase, et al., 2002) (Iwase, et al., 2003) (Figueroa, et al., 2004) (Gabriel, et al., 2005) dans un cadre d'application proche du nôtre (scène sportives). Ces derniers s'intéressent aux différents problèmes impliqués par le suivi d'objets (joueurs), notamment dans le cas d'images issues de caméras fixes (Figueroa, et al., 2004) (Gabriel, et al., 2005) ou lorsque les paramètres de la caméra sont connus (Iwase, et al., 2002) (Iwase, et al., 2003).

### 1.3.1 Approche sépare et rassemble

P. Figueroa *et al.* (Figueroa, et al., 2004) cherchent à déterminer la position des joueurs sur un terrain de football à partir de séquences d'images issues de plusieurs caméras statiques calibrées. Ils se sont particulièrement intéressés aux problèmes des occultations et ont proposé pour cela une méthode de type « sépare et rassemble ». Ils modélisent dans l'image les joueurs comme étant des « blobs » et effectuent le suivi de ces derniers à l'aide d'une représentation en graphes<sup>12</sup>.

#### Méthodologie

Pour effectuer le suivi des joueurs, P. Figueroa *et al.* (Figueroa, et al., 2004) procèdent en deux temps : la segmentation des joueurs et de leurs mouvements, puis le suivi des joueurs.

Ils réalisent la segmentation des joueurs en deux étapes. Dans un premier temps, ils extraient le mouvement des joueurs dans l'image via une approche de type soustraction d'arrière-plan. Ils utilisent ensuite des outils de morphologie mathématique pour filtrer l'image obtenue. Ainsi, cette image (binaire) contient différentes régions (formées par des pixels connexes), ou encore des « blobs » qui correspondent aux joueurs.

Les principales difficultés pour effectuer le suivi des joueurs sont engendrées par les occultations et les auteurs utilisent des informations spatio-temporelles telles que : la taille, la forme et la couleur des blobs, mais aussi la relation entre deux blobs – d'une image sur l'autre – par le biais d'une représentation en graphe (Figure 14) qui modélise alors la dépendance entre eux.

---

<sup>12</sup> Cf. Théorie des graphes

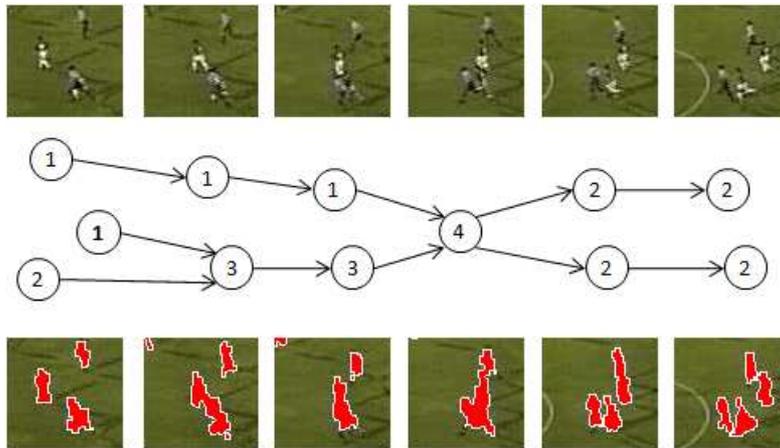


Figure 14 – Représentation en graphe d'une séquence d'images

Le graphe est obtenu à partir des blobs détectés dans l'étape précédente (segmentation des joueurs). Les nœuds correspondent aux blobs et les segments à la distance entre eux, d'une image sur l'autre. Le suivi des joueurs est alors obtenu par la recherche du chemin optimal dans le graph. Pour cela, P. Figueroa *et al.* (Figueroa, et al., 2004) ont dû résoudre les problèmes suivants :

- déterminer le nombre de composantes d'un blob,
- caractériser et séparer les blobs,
- mettre en correspondance chacun de ces blobs d'une image sur l'autre.

La cohérence temporelle du graphe est utilisée pour déterminer le nombre de composantes d'un blob. En effet, deux nœuds  $u_1$  et  $u_2$  dans une image au temps  $t$  appartiennent au même groupe  $v_1$  si et seulement s'il existe un nœud  $u$  de telle sorte qu'il existe des segments entre  $(v_1, u)$  et  $(v_2, u)$  dans le graphe.

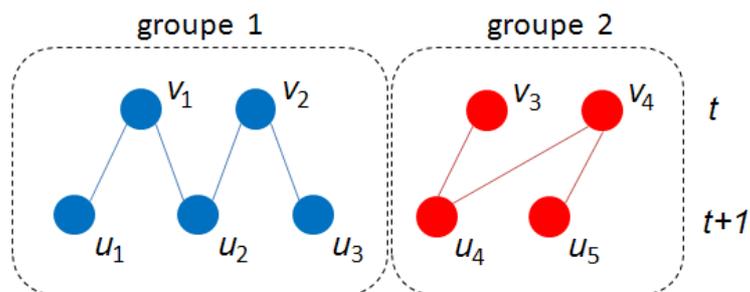


Figure 15 – Deux groupes formés de nœuds connexes

Les blobs sont identifiés à l'aide de critères colorimétriques, les auteurs définissent un modèle de joueurs (zones « uniformes » : t-shirt, short, chaussette, etc.) et utilisent deux filtres verticaux basés sur la distribution de leurs intensités. De plus, ils modélisent chaque blob (avant occultation) et lorsqu'ils détectent une séparation (horizontale ou verticale) ils cherchent à mettre en correspondance les « nouveaux » blobs avec les modèles précédents.

Enfin, dans le but de suivre un joueur, les auteurs supposent qu'un blob est défini (ainsi que le nœud associé) et il suffit alors, à chaque étape, de parcourir le graphe en considérant le chemin minimal (distance fonction des informations entre les blobs). Ainsi, le suivi des joueurs « isolés » est quasiment immédiat ; un seul segment est à déterminer pour chacune des étapes. Mais, le suivi des

joueurs « en contact » ou subissant des occultations est plus délicat et nécessite, en plus des informations couleurs des blobs et des distances, l'extraction d'informations sur la direction des joueurs pour choisir le « bon chemin » dans le graphe.

### Limitations

P. Figueroa *et al.* (Figueroa, et al., 2004) supposent qu'au plus trois joueurs se séparent en même temps ce qui est limitant dans le cas de fortes occultations (plus de trois joueurs s'occultant mutuellement). De plus, si la taille du blob n'est pas suffisante pour une séparation horizontale (ou verticale) alors les objets sont considérés comme étant en totale occultation et le suivi de ces objets est alors peu ou prou un échec. Aussi, les auteurs utilisent dans la pratique plusieurs caméras pour couvrir la totalité du terrain de football, mais ils n'abordent pas les problèmes liés à ce genre de méthodologie, notamment le calcul de la transformation liant ces différentes caméras. De même, ils n'abordent pas le problème du suivi appliqué à la reconstruction 3D. En effet, leur méthode de suivi permet de déterminer le mouvement des joueurs dans une image – ce dernier correspond au mouvement des joueurs conjugué au mouvement de la caméra – ce qui est problématique lorsque la caméra est mobile et que ses paramètres sont inconnus.

### 1.3.2 Approche directe

P. Gabriel *et al.* (Gabriel, et al., 2005) ont proposé une méthode de suivi dans le cas d'occultations. Leur approche consiste à caractériser un objet  $O$  comme étant une région d'intérêt ( $ROI$ ), une « zone » rectangulaire avec un taille donnée, composée par un ensemble de points d'intérêts ( $IP$ ). Ces derniers sont extraits de l'image par l'application du détecteur de Harris (étendue aux images couleurs). Ainsi, les auteurs réalisent le suivi d'une  $ROI$  par la mise en correspondance des  $IP$  d'une image sur l'autre. Pour cela, ils caractérisent les  $IP$  en fonction de paramètres locaux (couleurs, gradients, etc.) et de paramètres géométriques (distribution des  $IP$  par rapport au centre de gravité de la  $ROI$ ). Ils utilisent ensuite la distance de Mahalanobis pour déterminer la concordance entre ces  $IP$ .

### Notations

Soit  $ROI(i)$  la région d'intérêt d'un objet  $O$  de l'image  $i$  et  $IP_{ROI}(I)$  l'ensemble des points caractéristiques de cette  $ROI(i)$ . Ainsi, P. Gabriel *et al.* définissent pour chaque élément  $a \in IP_{ROI}(i)$  le vecteur suivant :

$$v_a(i) = (r, g, b, r_x, g_x, b_x, r_y, g_y, b_y, R, x_c, y_c)$$

avec  $(r, g, b)$  les valeurs couleurs,  $(r_x, g_x, b_x)$  les valeurs gradients,  $R$  la valeur de Harris (*cornerness*) et  $(x_c, y_c)$  le centre de gravité de la  $ROI(i)$ . La  $ROI(i)$  est alors caractérisée par l'ensemble des vecteurs :

$$V_{ROI}(i) = \{v_a(i) \mid a \in IP_{ROI}(i)\}$$

Ainsi, effectuer le suivi d'un objet  $O$  consiste à déterminer la région d'intérêt  $ROI(i + 1)$  de l'image  $i + 1$  à partir de la  $ROI(i)$  de l'image  $i$ .

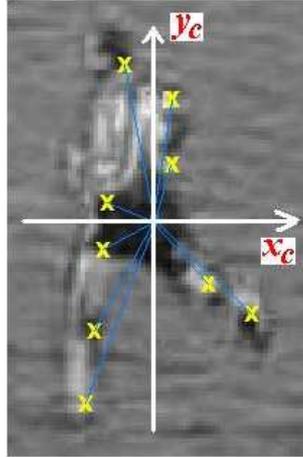


Figure 16 – Exemple de *ROI* et de ses différents *PI*

### Méthodologie

P. Gabriel *et al.* (Gabriel, et al., 2005) déterminent la région d'intérêt  $ROI(i + 1)$  et notamment son centre de gravité via l'utilisation d'un filtre de Kalman. Ils extraient ensuite de  $ROI(i + 1)$  l'ensemble des points caractéristiques  $IP_{ROI}(i + 1)$  par l'application du détecteur de Harris couleur. Enfin, ils appariant chacun des éléments de  $IP_{ROI}(i + 1)$  avec ceux correspondant de  $IP_{ROI}(i)$ . Pour cela, ils comparent – via la distance de Mahalanobis – les différents vecteurs qui caractérisent les points d'intérêts. Ainsi, le point d'intérêt  $a \in IP_{ROI}(i)$  a pour correspondant le point  $b \in IP_{ROI}(i + 1)$  qui a la distance de Mahalanobis minimale (inférieure à un seuil  $\theta$  donné) :

$$d_M^2(i, j) = (v_a(i) - v_b(i + 1))^T C^{-1} (v_a(i) - v_b(i + 1))$$

Avec  $C$  la matrice de covariance mesurée dans un premier temps dans une « séquence d'entraînement ». Les éléments de  $IP_{ROI}(i + 1)$  qui n'ont pas de correspondants sont alors écartés et le centre de gravité de la  $ROI(i + 1)$  est recalculé en fonction.

Cette approche permet alors de résoudre en partie le problème d'occultation. En effet, lorsque un  $IP$  est contenu par plusieurs  $ROI$  d'une même image, les auteurs parlent alors de « point disputé », il est possible de déterminer l'objet  $O$  qui contient « réellement » cet  $IP$ , un point disputé  $k$  appartenant alors à l'objet  $O$  qui a la distance  $d_M^O(i, k)$  minimale. Aussi, cette méthode est peu adaptée lorsque l'objet  $O$  est totalement occulté car d'une image sur l'autre aucun  $IP$  ne peut être relié à cet objet.

### Limitations

La méthode proposée par P. Gabriel *et al.* (Gabriel, et al., 2005) soulève d'autres complications. La première est liée à l'étape d'initialisation ; étape qui implique un certain nombre de contraintes fortes telles que la détection des objets (séparés entre eux). La seconde concerne l'étape d'appariement et plus particulièrement la distance de Mahalanobis. Cette dernière nécessite une étape « d'apprentissage » pour fournir une estimation robuste de la matrice de covariance  $C$ . Dans une moindre mesure, le seuil  $\theta$  est source d'erreur et doit être choisi avec attention (connaissance des images traitées, apprentissage, etc.). De même, la représentation d'un objet choisie par les auteurs peut également entraîner des erreurs. En effet, un objet est représenté par une

*ROI* rectangulaire et de taille fixe ce qui est problématique par exemple lorsque la caméra réalise un agrandissement (l'objet contenu par la *ROI* devient plus grand que cette dernière). Enfin, le suivi de joueurs est réalisé dans « l'espace image » et correspond alors aux mouvements des joueurs combinés au mouvement de la caméra ; ce qui est problématique dans le cadre de la reconstruction 3D d'une scène sportive.

### 1.3.3 Approche multicaméra

S. Iwase et H. Saito (Iwase, et al., 2002) (Iwase, et al., 2003) ont eux aussi proposé un système de suivi de joueurs de football dans le cas d'occultations. Leur système se compose de huit caméras fixes. Ils utilisent vingt points pour mettre en correspondance ces caméras et, pour effectuer le suivi, ils distinguent deux types d'opérations : les opérations intra-caméras et inter-caméras (Figure 17).

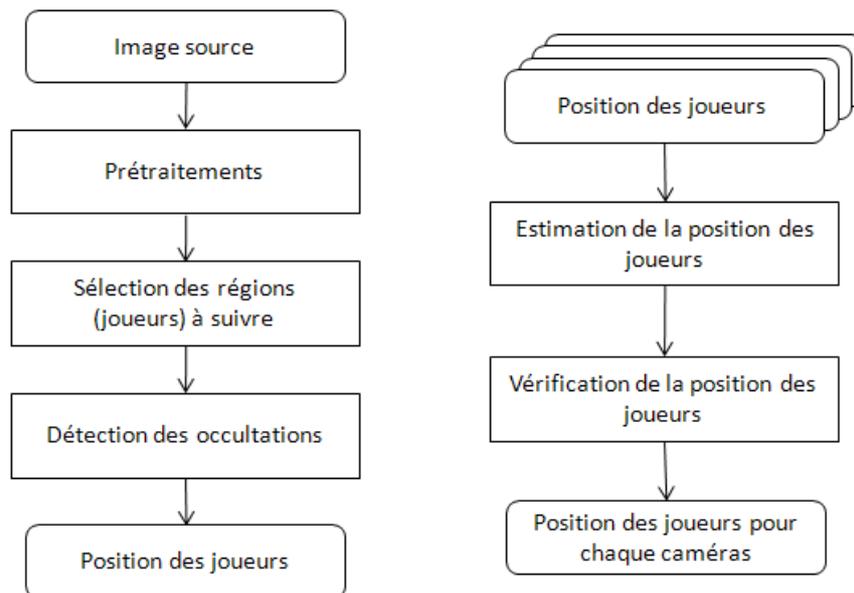


Figure 17 – Description du processus de suivi dans le cas d'occultations

La Figure 17 représente le processus de suivi dans le cas d'occultations et plus particulièrement les opérations intra-caméras (a) et inter-caméras (b).

#### Méthodologie

Les opérations intra-caméras se décomposent en trois étapes et sont effectuées indépendamment pour chacune des images obtenues par les huit caméras :

- prétraitements,
- sélection des régions qui représentent les joueurs suivis,
- détection des occultations.

La première étape consiste à effectuer un prétraitement sur l'image source dans le but d'obtenir les régions qui représentent un (ou plusieurs) joueur(s). Pour cela, S. Iwase et H. Saito (Iwase, et al., 2002) (Iwase, et al., 2003) utilisent une méthode de « soustraction de fond ». Ils obtiennent une image binaire qui, après traitement (réduction de bruit, des ombres, etc.) est utilisée pour extraire

les régions-joueurs. Ils calculent ensuite le centre, les coordonnées des pieds, la longueur, la largeur, la boîte englobante<sup>13</sup> de ces joueurs et leur attribuent une « étiquette ». L'étape suivante concerne la mise en correspondance de chacune des régions-joueurs, extraites et étiquetées dans l'image précédente, avec celles détectées dans l'image courante. Pour cela, ils calculent la similitude entre deux joueurs en fonction de la distance qui les sépare ainsi que de leur colorimétrie. La mise en correspondance peut ne pas être effectuée, notamment lorsqu'un joueur est en dehors du point de vue, ou que la région-joueur n'est pas détectée, ou encore lorsque le joueur est occulté. La dernière étape détermine si un joueur suivi est occulté : les auteurs considèrent que c'est le cas si la boîte englobante d'un joueur augmente considérablement d'une image sur l'autre. Par conséquent, le suivi de ce joueur ne peut être effectué par une opération intra-caméra.

Les positions des joueurs (i.e le suivi des joueurs) sont ensuite estimées via un ensemble d'opérations inter-caméras :

- l'estimation des positions des joueurs,
- la vérification de ces positions.

S. Iwase et H. Saito (Iwase, et al., 2002) (Iwase, et al., 2003) font l'hypothèse que les joueurs se situent sur un plan (le terrain de jeu) et ils utilisent alors la géométrie épipolaire<sup>14</sup> pour calculer la correspondance entre deux points de deux images issues de différentes caméras. De ce fait, si la position d'un joueur a pu être estimée dans une image (processus intra-caméras) alors cette position peut être retrouvée dans chacune des autres images. Dans le cas contraire, le suivi de ce joueur est un échec. De plus, les auteurs vérifient la cohérence des positions estimées pour les joueurs dans chacune des images.

### Limitations

L'approche proposée par S. Iwase et H. Saito (Iwase, et al., 2002) (Iwase, et al., 2003) nécessite la mise en œuvre d'un dispositif composé de huit caméras fixes. Par conséquent, leur approche ne permet pas de traiter le suivi dans le cas d'images issues de caméras « quelconques ». De plus, ce dispositif nécessite une mise en œuvre « lourde » (la mise en place de huit caméras, l'entretien de leurs réglages, etc.) et entraîne par conséquent un coût non négligeable. De même, la relation entre ces caméras doit être estimée (opérations inter-caméras), cette opération implique le calcul des transformations liant les images entre elles. Or, le calcul de ces dernières est réalisé par la mise en correspondance entre quatre points caractéristiques de chacune de ces images. Dans le cas d'images issues de caméras fixes il suffit de réaliser cette opération une unique fois, mais dans le cas d'images issues de caméras mobiles elle doit être réitérée. Il est alors indispensable de déterminer la relation entre ces images de façon automatique via le calcul de la transformation liant chacune de ces images ou via le calcul des paramètres des caméras.

---

<sup>13</sup> Le plus petit parallélépipède qui contient entièrement un objet

<sup>14</sup> Géométrie caractérisant les liens entre deux images stéréoscopiques.

## 1.4 Conclusion

Les méthodes de suivi existantes procèdent généralement en deux temps :

- l'étape d'extraction du mouvement, réalisée à partir de méthodes directes (soustraction d'arrière-plan, flot optique) ou indirectes (extraction d'éléments caractéristiques et mise en correspondance de ces éléments d'une image sur l'autre) ;
- l'étape de conservation de l'identité des objets le long de la séquence, accomplie par des approches sépare & rassemble, directes ou encore multicaméra.

Les méthodes de suivi appliquées dans un contexte proche du nôtre s'intéressent à des « sous-classes » de problèmes ; les images traitées sont issues de caméras statiques ou de caméras mobiles, mais dont les paramètres sont connus. Aussi, ces méthodes apportent des solutions partielles au problème de suivi. En effet, si les paramètres de prise de vue sont inconnus, l'étape de détection du mouvement est problématique (le mouvement extrait d'une séquence d'image correspond aux mouvements des objets et de la caméra) ; difficulté d'autant plus accrue lorsque la prise de vue est dynamique.

Dans notre cadre applicatif, la prise de vue n'est pas une donnée connue et peut varier dans le temps. Il est donc indispensable de proposer une méthode d'extraction du mouvement robuste aux variations de prise de vue. Nous proposons pour cela de définir un référentiel commun à chacune des images pour les comparer entre elles et ainsi s'affranchir de ces variations.

Dans le Chapitre II, nous montrons que la modélisation d'une scène de football permet de définir un référentiel pour comparer deux images consécutives et ainsi résoudre le problème du suivi appliqué à la reconstruction 3D dans le cas d'images issues de caméras mobiles non calibrées. Par conséquent, il est indispensable de déterminer la relation entre une image et ce référentiel ; opération qui correspond à un problème de recalage. Aussi, après avoir détaillé le processus de recalage, nous proposons une méthode entièrement automatique de mise en correspondance pour calculer la transformation liant l'image et le modèle (le référentiel). Nous montrons également comment mettre en œuvre cette méthode dans notre cadre applicatif qui implique dans un premier temps l'extraction d'une information suffisamment pertinente pour garantir la robustesse de notre mise en correspondance (via la détection des faisceaux de droites de l'image). Et dans un second temps la vérification de la cohérence de nos résultats (via le calcul des paramètres de la caméra).

## Chapitre II. Un référentiel commun aux images

Dans la partie précédente, nous avons fait état des techniques de suivi appliquées à la reconstruction 3D. Celles-ci s'emploient à extraire et à suivre les objets en mouvement le long d'une séquence d'images. Or, ce mouvement correspond à celui des objets, conjugué à celui de la caméra. Ainsi, l'emploi de ces techniques pour traiter des séquences d'images issues de caméras mobiles est problématique.

Le projet SimulFoot a pour objectif la reconstruction 3D de scènes de football à partir de séquences d'images (flux vidéo) issues d'une ou de plusieurs caméras dont les paramètres sont inconnus et variables dans le temps. Les problématiques soulevées par ce cadre applicatif sont essentiellement : l'analyse d'images (suivi d'entités, recalage d'images, extraction d'informations sur la prise de vue, etc.), la modélisation géométrique et la visualisation scientifique (représentation 3D et ergonomie cognitive). Comme nous l'avons précisé dans le chapitre précédent, la reconstruction 3D d'une scène de football implique le suivi d'entités (joueurs) dans une séquence d'images. Or, les techniques de suivi impliquent l'extraction du mouvement ; étape problématique dans le cas d'images issues de caméras mobiles non calibrées. Cette difficulté est liée à l'étape d'extraction du mouvement qui suppose le plus souvent la comparaison d'images consécutives. En effet, cette dernière est particulièrement délicate si ces images ne partagent pas un « repère » similaire. Une solution intuitive consiste à utiliser un référentiel commun pour établir la relation entre deux images consécutives et ainsi extraire le mouvement des objets contenus dans une séquence images.

Pour définir ce référentiel dans notre cadre applicatif (SimulFoot) la connaissance liée à la scène (filmée) est mise à contribution. En effet, cette scène est modélisée et correspond alors à :

- un terrain de football, et plus précisément un plan contenant un ensemble d'éléments caractéristiques connus (points, droites, cercles, etc.) défini par les divers tracés du terrain (surface de réparation, point de pénalty, etc.),
- un ensemble d'objets 3D mobiles disposés sur ce terrain tels que des joueurs de football, des arbitres ou encore le ballon.

De plus, un certain nombre d'hypothèses sont émises sur les images et la scène telle que :

- toute ou partie d'une image est une représentation totale ou partielle du modèle, i.e une image correspond à la représentation visuelle d'une scène de football ;
- le modèle et les éléments qui le composent sont entièrement plans (par exemple les droites qui correspondent aux tracés du terrain) ou ont au moins une composante plan (par exemple les joueurs sont disposés sur le sol).

Aussi, le terrain de football est un référentiel pertinent pour la comparaison de ces images ; référentiel désigné dans la suite de ce mémoire comme étant le plan modèle (Figure 18). Il est alors indispensable de déterminer la transformation géométrique permettant de passer du plan image – la partie de l'image qui correspond au terrain de football – au plan modèle ; opération qui correspond à un problème de recalage.

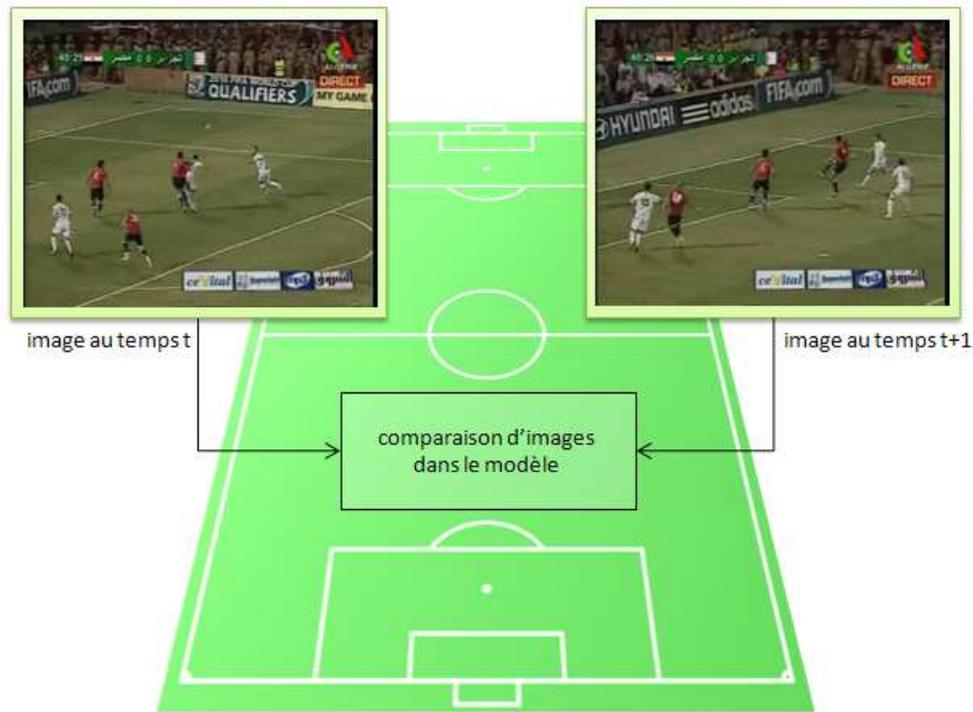


Figure 18 – Le référentiel utilisé pour comparer deux images correspond au terrain de football

## 2.1 Le processus de recalage

Le recalage entre une image et un modèle est une opération incontournable dans de nombreux domaines tels que l'imagerie satellitaire (Huseby, et al., 2005) (Joshi, et al., 2006) (Aanaes, et al., 2008), l'imagerie médicale (Thomas, 1998) (Keysers, et al., 2003) (Makela, 2003) ou plus généralement la vision assistée par ordinateur (Chin, et al., 1986) (Lucas, et al., 1981) (Gohlke, 2009).

La transformation recherchée liant une image et un modèle est plus ou moins complexe (translation, rotation, homographie, similitude, etc.) en fonction des caractéristiques de la scène considérée et du type d'acquisition de l'image. Le plus souvent, les méthodes de recalage basées sur un modèle se décomposent en deux étapes (Brown, 1992) (Maintz, et al., 1998) (Zitova, et al., 2003) : l'extraction d'éléments caractéristiques dans l'image, puis la mise en correspondance entre ces éléments et ceux du modèle (Figure 19).

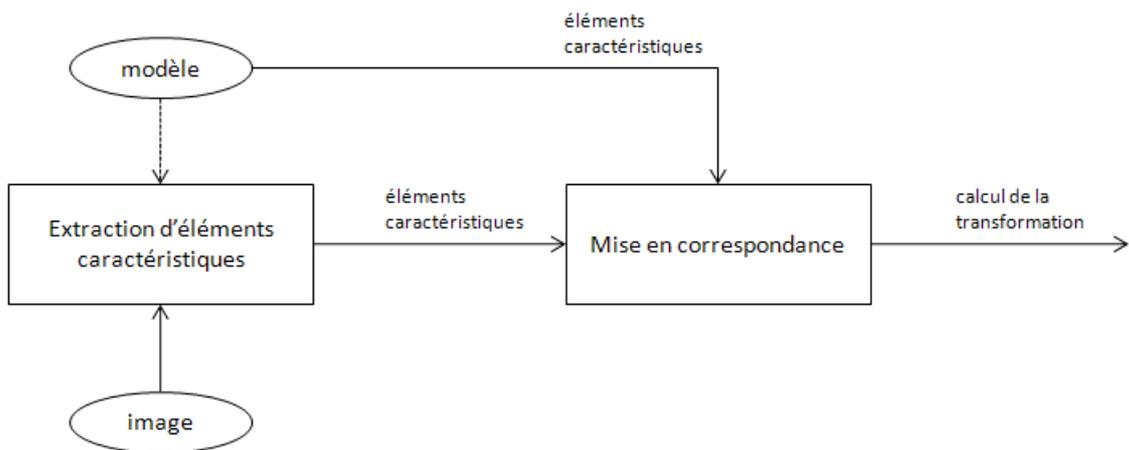


Figure 19 – Décomposition du processus de recalage

La transformation recherchée dans notre cadre applicatif est une transformation plan à plan, celle liant le plan image et le plan modèle (le terrain de jeu), et correspond alors à une homographie. J. Mundy et A. Zisserman (Mundy, et al., 1992) ont montré que l'appariement d'un quadruplet de points (sans triplets de points alignés) permet le calcul de l'homographie qui associe deux plans. Ainsi, le recalage du plan image avec le plan modèle est réalisé via l'appariement de quatre amers<sup>15</sup> du plan image avec ceux correspondant du plan modèle. Cette opération a d'ailleurs été effectuée par A. Le Troter (Le Troter, 2006), ce dernier calcule l'homographie à partir de l'appariement manuel entre quatre amers des deux plans (Figure 20).

Dans le but de traiter une séquence d'images (flux vidéo) il est impératif d'automatiser entièrement le processus de recalage, i.e d'automatiser les étapes d'extraction et de mise en correspondance d'éléments caractéristiques.

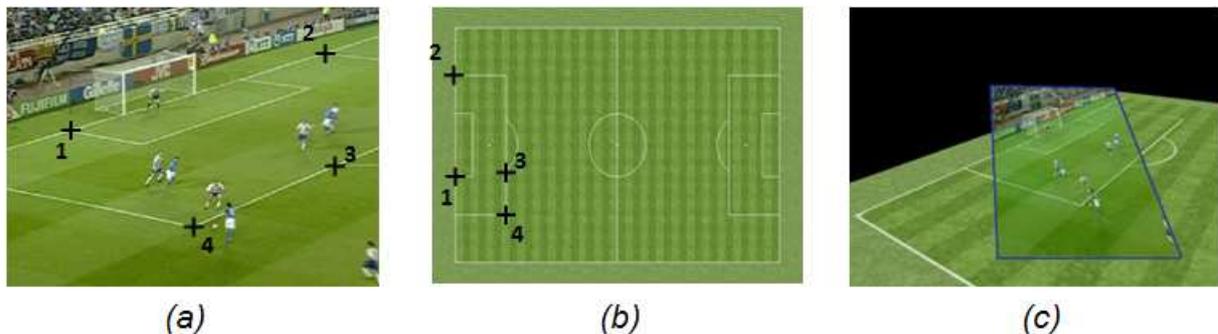


Figure 20 – Le recalage manuel consiste à sélectionner quatre amers dans le plan image (a) et leurs correspondants dans le plan modèle (b) pour obtenir la transformation liant ces deux plans (c)

### 2.1.1 Extraction d'éléments caractéristiques

La première étape du processus de recalage consiste à extraire dans l'image un ensemble d'éléments caractéristiques tels que des points, des droites ou encore des contours. Cette étape est généralement guidée par la connaissance du modèle. L'extraction d'éléments caractéristiques dans une image est un problème fondamental qui a donné lieu à de nombreux travaux tels que ceux de J.

<sup>15</sup> Intersections de droites

Canny (Canny, 1986) et de R. Deriche (Deriche, 1987) sur l'extraction de segments ou encore ceux de C. Harris et M. Stephens (Harris, et al., 1988) sur l'extraction de points caractéristiques. Pour plus de détails nous pourrions nous référer à la revue de B. Zitova et J. Flusser (Zitova, et al., 2003).

Dans notre cadre applicatif, Arnaud Le Troter (Le Troter, 2006) a proposé une méthode d'extraction qui procède en deux temps : il détecte à l'aide d'attributs colorimétriques la région d'intérêt de l'image (la région qui correspond au terrain de jeu) puis il utilise une approche de type transformée de Hough pour extraire un ensemble d'éléments caractéristiques (droites et ellipses) dans cette région (Figure 21).

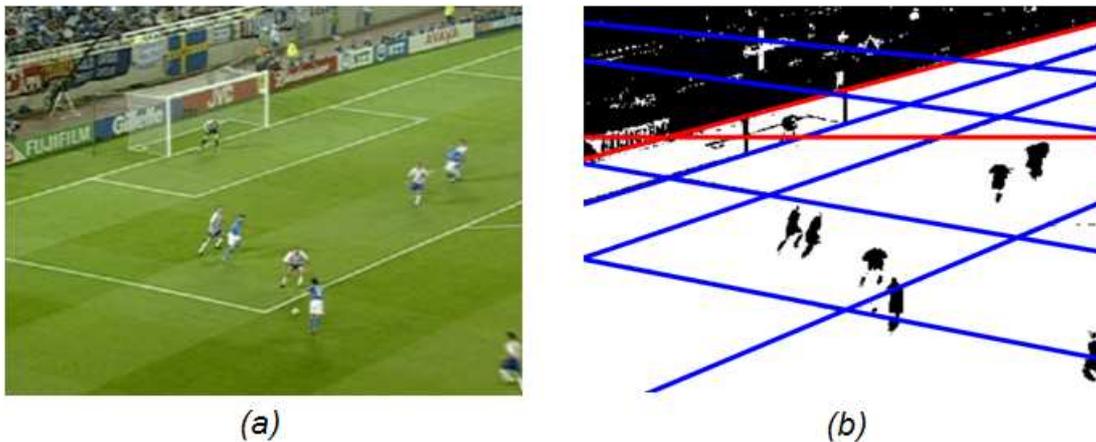


Figure 21 – Détection de droites dans une image

La Figure 21 est composée d'une image source (a) et une d'une image binaire (b), i.e le résultat de l'extraction de la ROI de l'image source. Un certain nombre d'éléments caractéristiques (des droites) ont été extraits de cette image binaire. Ces derniers correspondent alors à des éléments existant dans le modèle (en bleu) ou à des droites qui n'ont aucun lien avec le modèle (en rouge) et qui sont le plus souvent liées à des erreurs de détection.

L'étape d'extraction d'éléments caractéristiques a pour objectif la détection des éléments de telle sorte que ces derniers correspondent à ceux du modèle. Aussi, il est crucial d'éviter de prendre en compte des éléments qui n'ont pas de correspondants dans le modèle. Or, ces derniers sont le plus souvent issus d'erreurs de détection et pour les identifier on utilise la connaissance a priori du modèle. Par exemple, le terrain de jeu (le modèle) est composé de tracés (des droites) qui ont pour propriété de former deux groupes de droites parallèles. Par conséquent, les droites de l'image forment également deux faisceaux de droites et les déterminer aide à invalider les droites issues d'erreurs de détection.

### 2.1.2 Mise en correspondance d'éléments caractéristiques

La mise en correspondance d'éléments caractéristiques entre une image et un modèle est un processus particulièrement délicat car il implique la comparaison entre des éléments de différentes natures (les pixels d'une image et les éléments d'un modèle) et exclut alors l'utilisation de mesures classiques de similarité (distance, corrélation, etc.). Néanmoins, ce problème a été abordé par J. B. Hayet *et al.* (Hayet, et al., 2005) et K. Okuma *et al.* (Okuma, et al., 2004) dans un cadre applicatif proche du nôtre.

En effet, J. B. Hayet *et al.* (Hayet, et al., 2005) ont proposé une méthode de mise en correspondance appliquée au recalage de scènes sportives. Dans un premier temps, ils extraient de l'image un ensemble de segments qu'ils attribuent à différents groupes en fonction de leurs points de fuite. Ils cherchent ensuite à mettre en correspondance deux amers<sup>16</sup> formés par l'intersection de deux droites de deux groupes distincts avec ceux du modèle. Pour cela, ils définissent une mesure de similarité basée sur la connaissance de leur modèle et plus particulièrement sur dix-sept caractéristiques (intrinsèques aux droites et à leurs positions relatives, à leurs intersections et à leurs points de fuite). Enfin, ils valident cet appariement en s'appuyant sur la propriété de conservation des birapports de la projection perspective. Toutefois, la mise en correspondance proposée par ces auteurs est fortement liée aux caractéristiques de leur modèle et ne peut donc être utilisée dans un cadre plus général.

Pour éviter les problèmes soulevés par l'étape de mise en correspondance, K. Okuma *et al.* (Okuma, et al., 2004) ont proposé une méthode appliquée aux images issues d'un flux vidéo. Celle-ci se déroule en deux temps : une étape d'initialisation qui consiste à mettre en correspondance manuellement un ensemble d'éléments de l'image avec ceux du modèle, puis une étape de suivi qui s'emploie à extraire et à suivre ces éléments dans les différentes images du flux vidéo. Ainsi, l'étape de mise en correspondance d'éléments est « remplacée » par une étape de suivi, la mise en correspondance étant effectuée une première fois au préalable. Cependant, ce type d'approche entraîne inévitablement des complications le plus souvent liées à la méthode de suivi employée. Parmi elles, l'étape d'initialisation qui doit être répétée quand, d'une image sur l'autre, un élément caractéristique n'est plus visible ou lorsqu'une tout autre partie de la scène est acquise. L'approche proposée par K. Okuma *et al.* (Okuma, et al., 2004) permet le recalage de séquences d'images issues d'une vidéo, mais les problèmes soulevés par l'étape d'initialisation et par les techniques de suivi ne garantissent pas un recalage entièrement automatique.

### 2.1.3 Conclusion

L'extraction automatique d'éléments caractéristiques dans une image est une opération éprouvée, déjà traitée dans notre cadre applicatif (Le Troter, 2006) et que l'on considère comme suffisamment robuste. Aussi, le problème de recalage peut être réduit au problème de mise en correspondance. Or, cette opération est particulièrement délicate et mérite toute notre attention, il n'existe en effet aucune solution appropriée à notre cadre applicatif.

Dans la partie suivante, nous proposons une méthode originale de mise en correspondance qui apporte une solution complète au problème de recalage automatique entre une image et un modèle. Aussi, nous exposons la mise en œuvre de cette méthode dans notre cadre applicatif. Pour cela nous nous appuyons sur la connaissance liée à notre modèle ; via l'extraction d'informations dans l'image et plus précisément l'extraction de faisceaux de droites. De par les spécificités du modèle (symétrie, rapport d'échelle, etc.), nous montrons qu'il est nécessaire d'extraire les paramètres de la caméra pour valider nos résultats.

---

<sup>16</sup> Un amer est un point formé par l'intersection de deux droites

## 2.2 Mise en correspondance

### 2.2.1 Cas général

Dans cette partie, nous exposons un schéma général de mise en correspondance automatique entre une image et un modèle. Nous présentons par la suite son application dans notre cadre applicatif.

#### 2.2.1.1 Présentation

Soit  $M$  le modèle d'une scène et  $I$  une image de cette scène acquise par une caméra. L'image  $I$  est alors une représentation partielle de ce modèle et toute ou partie de cette image correspond au modèle ayant subi une transformation  $T$ . Le recalage entre cette image et ce modèle revient alors à calculer cette transformation.

Tout d'abord, il est primordial de déterminer le type de la transformation  $T$  recherchée (translation, rotation, similitude, etc.). Le calcul de cette transformation est ensuite réalisé par la mise en correspondance entre les éléments caractéristiques de l'image et ceux du modèle. Pour calculer la transformation  $T$  nous procédons de la façon suivante : dans un premier temps, nous déterminons un sous-ensemble d'éléments caractéristiques dans l'image de telle sorte que l'appariement de ces éléments avec ceux correspondant du modèle permette le calcul de la transformation recherchée, avec des contraintes minimales. Ces dernières sont alors conditionnées au type de transformation et au type d'éléments mis en correspondance. Dans un second temps, nous cherchons le sous-ensemble d'éléments caractéristiques du modèle qui correspond au sous-ensemble fixé dans l'image. Par exemple, si la transformation recherchée est une translation et que les éléments caractéristiques sont des points, il suffit alors de fixer un point dans l'image et de chercher ensuite son correspondant dans le modèle.

#### 2.2.1.2 Formalisme et hypothèses

Soit  $M_{elem}$  l'ensemble des éléments caractéristiques connus de notre modèle et  $I_{elem}$  l'ensemble des éléments extraits dans une image. Ainsi, une partie ou la totalité des éléments de  $I_{elem}$  correspondent aux éléments de  $M_{elem}$  ayant subi la transformation  $T$  :

$$\forall x \in I_{elem} - \{J_{elem}\}, T(x) \in M_{elem}$$

avec  $J_{elem} \subset I_{elem}$  l'ensemble des éléments de l'image qui ne correspondent à aucun des éléments de  $M_{elem}$ .

Soit l'opération  $corr$  qui fait correspondre un à un et de façon unique chaque élément de deux ensembles. Nous supposons alors qu'il existe  $I'_{elem}$  et  $M'_{elem}$  respectivement deux sous-ensembles de  $I_{elem}$  et  $M_{elem}$  de telle sorte que l'opération  $corr$  permette le calcul de la transformation  $T$  :

$$\exists I'_{elem}, \exists M'_{elem} \mid corr(I'_{elem}, M'_{elem}) \Rightarrow T \quad (1)$$

Ainsi, recalibrer une image et son modèle, i.e calculer la transformation entre cette image et son modèle, revient à déterminer les sous-ensembles  $I'_{elem}$  et  $M'_{elem}$  qui vérifient l'équation (1). Pour cela, nous faisons l'hypothèse suivante : si chaque élément de  $I_{elem}$  transformé par  $t$  correspond à un unique élément de  $M_{elem}$  alors  $t$  est la transformation  $T$  recherchée :

$$\forall x \in I_{elem} - \{J_{elem}\}, t(x) \in M_{elem} \Rightarrow t = T \quad (2)$$

### 2.2.1.3 Méthode

Notre méthode de mise en correspondance d'éléments caractéristiques entre une image et son modèle procède alors en deux temps :

- une étape d'initialisation : fixer un sous-ensemble d'éléments caractéristiques  $I'_{elem}$  de l'image de telle sorte qu'il vérifie les contraintes minimales pour le calcul de la transformation  $T$ ,
- une étape de recherche : déterminer le sous-ensemble  $M'_{elem}$  qui correspond au sous-ensemble  $I'_{elem}$  de l'image, i.e déterminer  $M'_{elem}$  qui vérifie l'équation (1).

#### Etape d'initialisation

L'étape d'initialisation consiste à fixer  $I'_{elem}$  vérifiant les contraintes minimales pour le calcul de la transformation  $T$ . Pour cela il est nécessaire de poser un certain nombre de conditions en fonction des éléments caractéristiques extraits dans l'image et de la transformation recherchée. De plus, pour chaque sous-ensemble du modèle équivalent<sup>17</sup> à  $I'_{elem}$  il est possible de calculer une transformation de même type que la transformation  $T$  :

$$\exists M'_{elem} \mid corr(I'_{elem}, M'_{elem}) \Rightarrow t$$

Ainsi, La transformation  $t$  fait correspondre à chaque élément de  $I'_{elem}$  un élément de  $M'_{elem}$ . Mais, cette transformation n'est pas nécessairement celle recherchée. Il est alors nécessaire de déterminer quel est le sous-ensemble  $M'_{elem}$  qui, mis en correspondance avec  $I'_{elem}$ , permet le calcul de  $T$  ; son existence étant garantie de par les conditions utilisées pour fixer  $I'_{elem}$ .

#### Etape de recherche

L'étape de recherche consiste à déterminer  $M'_{elem}$  vérifiant l'équation (1). Pour cela, nous procédons de façon combinatoire :  $I'_{elem}$  est mis en correspondance avec chaque sous-ensemble de  $M_{elem}$  équivalent pour obtenir un ensemble de transformations  $t$  de même type que la transformation  $T$ . Nous cherchons ensuite quelle est la transformation  $t$  qui vérifie l'équation (1), i.e qui fait correspondre à chaque élément de  $I_{elem} - \{J_{elem}\}$  un élément de  $M_{elem}$ .

Remarque 1 : l'ensemble  $J_{elem}$  n'est pas nécessairement une donnée connue et doit être identifié.

Remarque 2 : il est indispensable de définir une mesure pour estimer quelle est la transformation  $t$  qui vérifie « au mieux » l'équation (1). Celle-ci dépend alors de la transformation recherchée et des éléments caractéristiques (nature et nombre).

### 2.2.1.4 Conclusion

---

<sup>17</sup> Sous-ensemble qui contient des éléments de même type que  $I'_{elem}$  et en quantité identique

Nous avons proposé un schéma général qui répond au problème de mise en correspondance automatique d'éléments caractéristiques entre une image et un modèle. Aussi, nous avons mis en évidence les points délicats soulevés par cette opération :

- fixer un sous-ensemble minimal de primitives d'un plan (qui garantit l'existence de la transformation liant nos deux plans) et
- déterminer le sous-ensemble correspondant dans le plan modèle.

Par la suite, nous montrons comment utiliser ce schéma dans notre cadre applicatif et notamment comment utiliser la connaissance a priori sur les primitives et la transformation liant les deux plans que nous cherchons à mettre en correspondance.

## 2.2.2 Mise en œuvre dans notre cadre applicatif

### 2.2.2.1 Etape d'initialisation

L'étape d'initialisation consiste à fixer un sous-ensemble d'éléments caractéristiques de l'image de telle sorte que l'appariement de chacun de ces éléments avec ceux correspondant du modèle garantisse, avec des conditions minimales, le calcul de la transformation liant l'image et le modèle.

Tout d'abord, il est nécessaire de déterminer en fonction du cadre applicatif quel est le type de la transformation recherchée. Notre modèle – un terrain de football – est assimilé à un plan  $P_M$  dont les caractéristiques sont connues,  $P_M$  désigne alors un plan modèle. Soit  $I$  une image de ce terrain de football acquise par une caméra, alors tout ou partie  $P_I$  de  $I$  correspond au plan modèle ayant subi une transformation projective ;  $P_I$  désigne alors un plan image. La transformation du plan image en plan modèle correspond donc à une homographie que l'on note  $H$  (Figure 22).

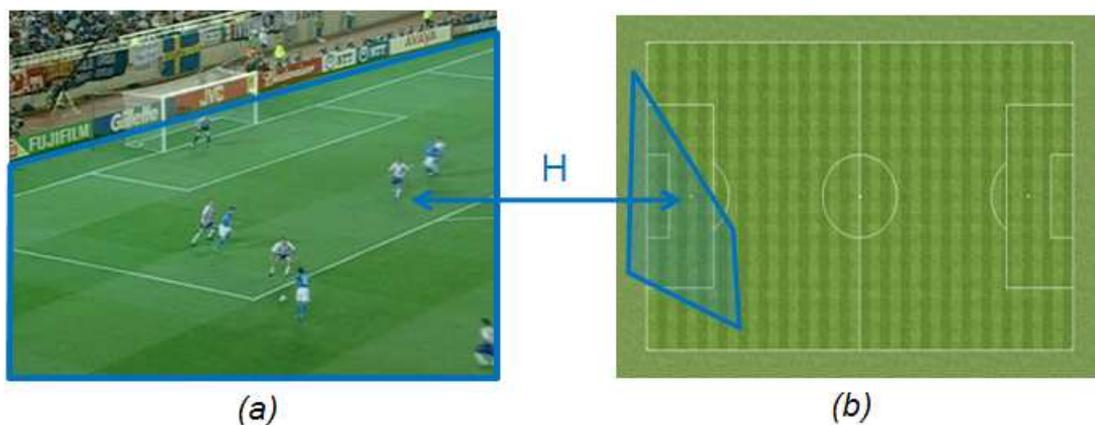


Figure 22 – Exemple de recalage : (a) plan image, (b) plan modèle

Il est également primordial, en fonction du modèle et de la transformation recherchée, de déterminer le type d'éléments caractéristiques à extraire dans l'image ainsi que les conditions nécessaires pour que la mise en correspondance de ces derniers permette le calcul de cette transformation. J. Mundy et A. Zisserman (Mundy, et al., 1992) ont montré que l'appariement d'un quadruplet de points (sans triplets de points alignés) de deux plans permet le calcul de l'homographie qui les associe. Le plan modèle est constitué d'un ensemble de tracés (lignes de but, lignes de touche, surfaces de réparation, etc.) qui sont assimilés à des droites ; ces dernières

décrivent deux groupes de droites parallèles. Ainsi, les intersections de deux droites de deux groupes forment un quadruplet de points qui vérifie la condition énoncée ci-dessus. Les droites du plan image correspondent à celles du plan modèle ayant subi une transformation projective. Elles appartiennent donc à deux groupes de droites – on parle alors de faisceaux – et, de façon similaire, deux droites de deux faisceaux vérifient la condition de J. Mundy et A. Zisserman (Mundy, et al., 1992). La mise en correspondance de ces droites avec celles équivalentes<sup>18</sup> du plan modèle permet alors le calcul de la transformation qui associe le plan image et le plan modèle.

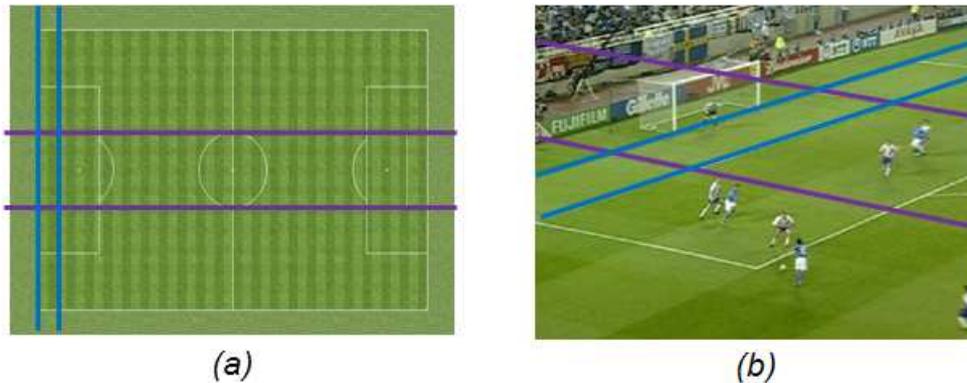


Figure 23 – Condition nécessaire à la mise en correspondance

La Figure 23 est composée de deux droites de deux groupes du plan modèle (a) et des droites correspondantes dans le plan image (b) de telle sorte que le calcul d’une homographie soit réalisable.

A l’inverse des droites du plan modèle, l’appartenance des droites du plan image aux différents faisceaux n’est pas une donnée connue. Cependant, classer ces droites en fonction de leurs points de fuite permet de déterminer ces faisceaux. Nous montrons dans la partie suivante (cf. Classification de droites en fonction de leurs points de fuite) comment déterminer ces faisceaux et nous supposons pour le moment que cette opération est réalisée.

De plus, la détection de ces faisceaux permet d’identifier les droites issues d’erreurs de détection ; droites qui ne correspondent alors à aucune du modèle. En effet, soit  $D = \{d_1, d_2, \dots, d_n\}$  l’ensemble des droites détectées dans l’image et  $D_{erreur} \subset D$  l’ensemble des droites issues d’erreurs de détection, alors la connaissance de notre modèle permet de définir, en fonction des points de fuite de l’image, deux groupes de droites  $D_{groupe1}$  et  $D_{groupe2}$  de telle sorte que :

$$D_{groupe1} \cup D_{groupe2} = D - D_{erreur}$$

$$D_{groupe1} \cap D_{groupe2} = \emptyset$$

Les droites issues d’erreurs de détection  $D_{erreur}$  sont identifiées via l’application de deux critères :

- l’appartenance des droites aux deux groupes  $D_{groupe1}$  et  $D_{groupe2}$ ,
- un score de pertinence attribué à chacune des droites qui est calculée en fonction de la détection de droite de type transformé de *Hough*.

<sup>18</sup> Droites dont les intersections forment un quadruplet de points non trois par trois alignés

Les droites issues d'erreurs de détection qui n'appartiennent à aucun des deux groupes sont naturellement écartées et celles qui ont un score inférieur à un seuil donné sont considérées comme étant des droites issues d'erreurs de détection (Figure 24).

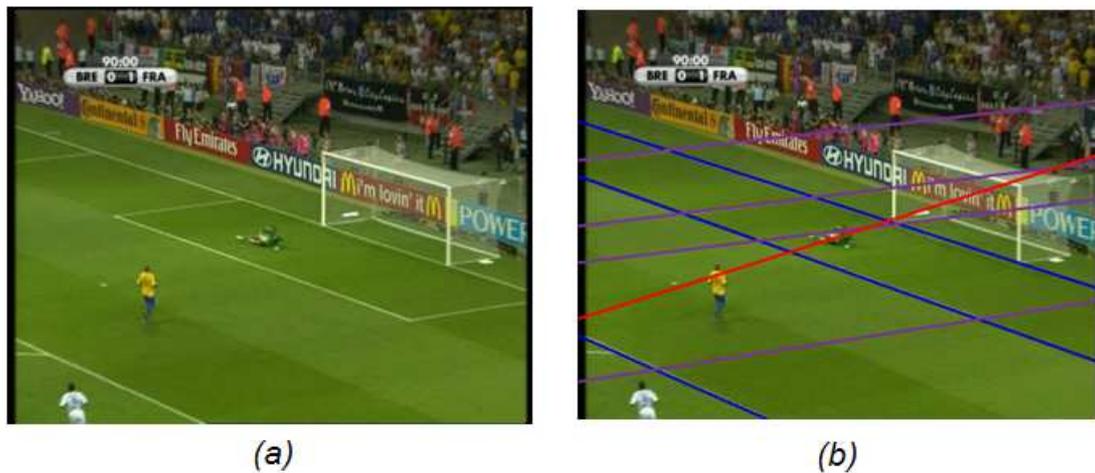


Figure 24 – Détection de droites qui correspondent à des droites du modèle

La Figure 24 représente une image source (a) et l'extraction des faisceaux de droites (en bleu et violet) de cette image. La droite rouge n'appartient à aucun des deux faisceaux et correspond alors à une droite issue d'erreurs de détection.

Nous avons donc montré comment sélectionner un sous-ensemble d'éléments caractéristiques dans le plan image de telle sorte qu'il garantisse, à des conditions minimales, le calcul de la transformation qui associe l'image et le modèle. Il est désormais nécessaire de déterminer le sous-ensemble correspondant dans le plan modèle.

### 2.2.2.2 Etape de recherche

L'étape de recherche consiste à identifier, pour un sous-ensemble d'éléments caractéristiques donné du plan image, le sous-ensemble d'éléments correspondants du plan modèle, i.e qui vérifie l'équation (1). Pour cela, nous procédons de façon combinatoire : le sous-ensemble d'éléments de l'image est mis en correspondance avec chaque sous-ensemble équivalent du modèle pour obtenir un ensemble de transformations. Aussi la transformation  $h$ , qui fait correspondre à chaque élément de l'image un élément du modèle, satisfait l'équation (2) et correspond alors à la transformation  $H$  liant l'image et le modèle.

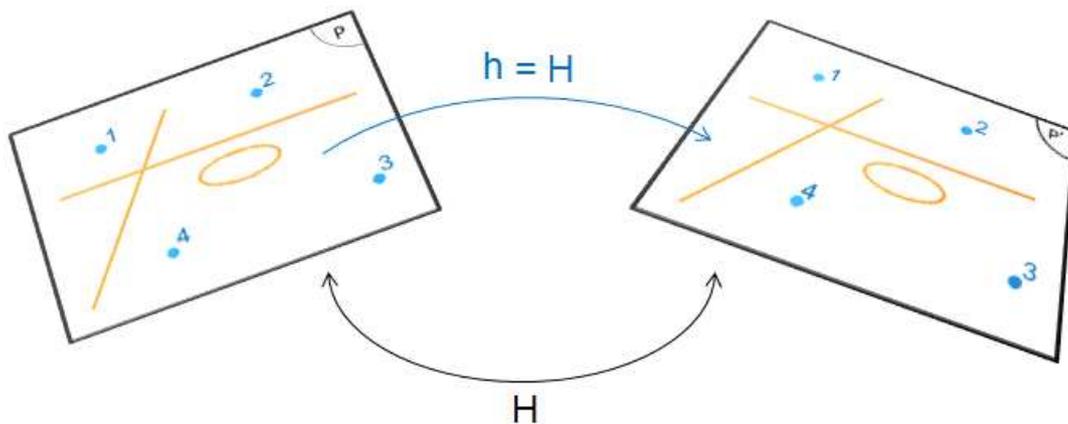


Figure 25 – Recalage plan à plan

La Figure 25 représente deux plans  $P$  et  $P'$  de telle sorte que  $P'$  soit le transformé par  $H$  de  $P$ . La mise en correspondance d'éléments caractéristiques de  $P$  (les points bleus) avec ceux de  $P'$  permet le calcul d'une transformation  $h$ . Cette dernière correspond alors à la transformation  $H$  recherchée si et seulement si chacun des éléments de  $P$  (droites et ellipses oranges) sont transformés par  $h$  en un élément de  $P'$ .

Dans notre cadre applicatif, les éléments caractéristiques extraits dans l'image sont des droites. Le sous-ensemble sélectionné dans l'image (cf. Etape d'initialisation) est mis en correspondance avec chaque sous-ensemble correspondant<sup>19</sup> du modèle pour obtenir un ensemble d'homographies. Pour vérifier quelle est celle qui correspond à la transformation recherchée nous vérifions quelle homographie transforme chaque élément de l'image en élément du modèle.

Pour chacune de ces homographies nous utilisons une mesure qui évalue la probabilité que celle-ci soit la transformation recherchée. Ainsi, pour une homographie  $h$  donnée, chaque droite  $d_i$  de l'image est transformée par  $h$  puis comparée aux droites  $d_m$  du modèle ; comparaison basée sur la métrique :

$$dist(d_i, d_m) = \overline{\|h(d_i), d_m\|}$$

avec  $dist$  la distance entre deux droites dans l'espace polaire. La droite  $d_i$  de l'image correspond alors à la droite  $d_m$  du modèle qui a la distance minimale et le *score* attribué à l'homographie  $h$  est calculé de la façon suivante :

$$score = \max \min dist(d_i, d_m)$$

Ainsi, l'homographie  $h$  qui obtient le plus petit *score* est celle recherchée. Mais, il est nécessaire de déterminer le nombre de droites pour lequel ce *score* est considéré comme discriminant. Or, les droites utilisées pour calculer une homographie  $h$  – appartenant au sous-ensemble de droites de l'image mis en correspondance avec le sous-ensemble du modèle – se transforment nécessairement en droites du modèle. Il est donc indispensable d'utiliser les droites de l'image qui n'appartiennent pas à ce sous-ensemble. De plus, l'utilisation d'une seule droite ne permet pas d'obtenir un score

<sup>19</sup> Quatre droites de telle sorte que deux droites appartiennent à un premier groupe et les deux droites restantes à un second groupe.

discriminant ; trop de cas ambigus sont liés aux propriétés du modèle (symétries, rapport d'échelle, etc.). Par contre, l'utilisation de deux droites – une de chaque groupe – permet de lever la quasi-totalité de ces ambiguïtés. Néanmoins, le plan modèle possède de nombreuses symétries, notamment les symétries d'axe  $Ox$  et d'axe  $Oy$  et les appariements symétriques de « l'appariement solution » sont également solution.

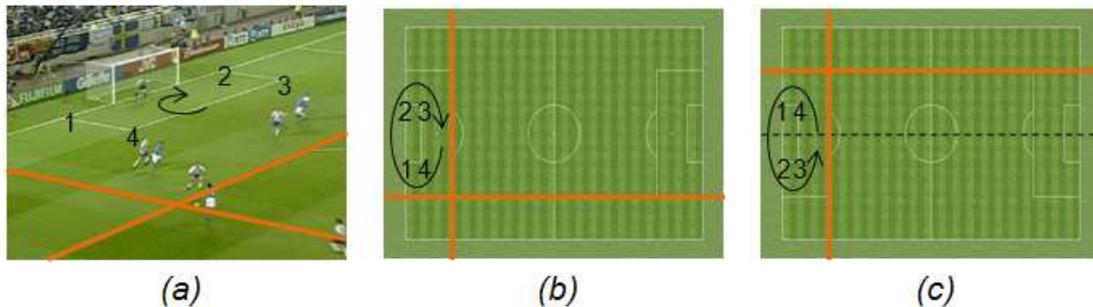


Figure 26 – Symétrie d'axe  $Ox$

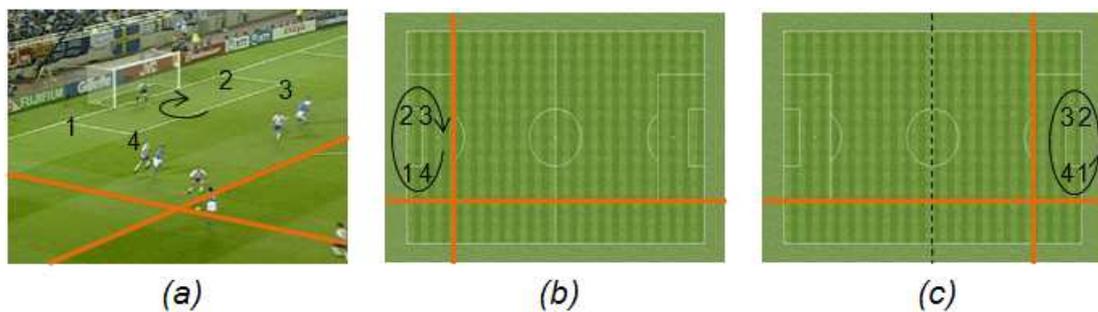


Figure 27 – Symétrie d'axe  $Oy$

Les Figure 26 et Figure 27 sont composées de trois images :

- le plan image qui contient quatre amers (points issus d'intersections de quatre droites) et deux droites (en orange),
- l'appariement des amers du plan image avec leur correspondant dans le plan modèle,
- l'appariement symétrique.

Aussi, les droites du plan image (en orange) se projettent dans le plan modèle sur des droites existantes (b) et (c).

Pour éviter ces « faux positifs » il suffit d'ordonner le quadruplet de points du plan image (orienter le quadrilatère associé) et l'apparier alors avec seulement ceux du plan modèle ordonnés dans le même sens. Mais, ces deux symétries impliquent l'existence d'une symétrie centrale et il existe alors nécessairement deux appariements distincts qui vérifient nos contraintes. Ce cas peut être ignoré en considérant comme plan modèle seulement un demi-terrain de football.

Néanmoins, quelques ambiguïtés subsistent et sont liées à la variation de notre modèle. En effet, les lignes de touche et les lignes de but – les contours de notre modèle – peuvent prendre des valeurs respectivement comprises entre 90-120 mètres et 45-90 mètres. Ainsi, des rapports d'échelles équivalents entraînent inévitablement des ambiguïtés (Figure 28).

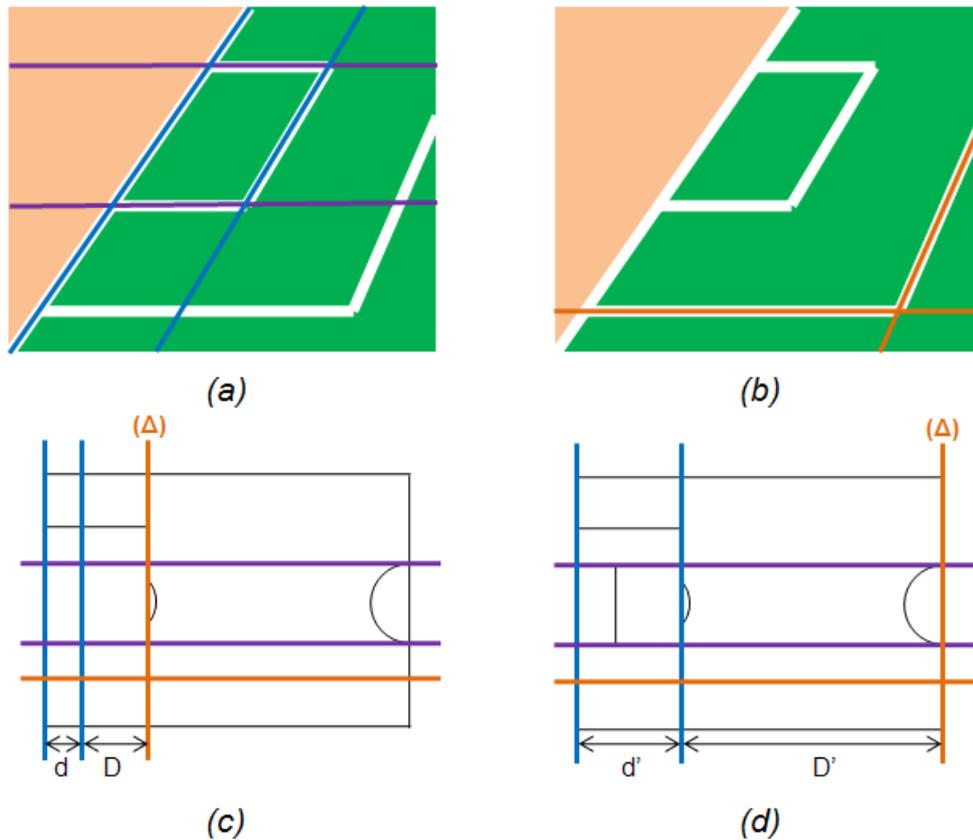


Figure 28 – Cas ambigu

La Figure 28 est composée de : (a) les droites du plan image sélectionnées pour le calcul de la transformation (en bleu et violet), (b) les droites du plan image sélectionnées pour valider cette transformation (en orange), (c) et (d) la mise en correspondance et si  $\frac{d}{D} = \frac{d'}{D'}$  alors les deux droites ( $\Delta$ ) se projettent sur des droites du plan modèle existantes.

Ces rapports d'échelle ne permettent pas de mettre en place un score qui satisfait la validation. En effet, lorsqu'un appariement erroné a un « rapport d'échelle » équivalent à celui de l'appariement « réel », le score de ce dernier peut être moins discriminant (erreurs d'approximations, etc.). Il est alors nécessaire d'utiliser un autre critère qui ne peut être ni géométrique ni numérique. Nous choisissons d'utiliser la connaissance a priori de la scène, et plus particulièrement celle liée à la prise de vue, pour vérifier la cohérence de l'appariement obtenu. En effet, la mise en correspondance d'un quadruplet de points de l'image avec un autre du modèle aboutit au calcul de la transformation liant l'image et le modèle. Celle-ci est alors utilisée pour calculer les paramètres de la caméra ayant servi à l'acquisition de l'image. Ainsi, une caméra ayant une position aberrante permet d'invalider cette transformation, i.e d'écarter l'appariement erroné. En effet, la scène est acquise par une caméra qui respecte un certain nombre de propriétés ; par exemple, la coordonnée hauteur d'une caméra ne peut pas avoir une valeur négative. Nous définissons alors un ensemble de « zones interdites » pour déterminer si une position caméra est aberrante. Nous considérons ici que l'opération de calibrage – le calcul des paramètres de la caméra – est d'ores et déjà réalisée, mais nous montrons dans une prochaine partie (cf. Calcul des paramètres de la caméra) que cette opération n'est pas triviale et comment la résoudre.

### Pseudo-code :

- Etape d'initialisation
  - Extraire l'ensemble des droites  $D_i$  de l'image, leur attribuer un score de pertinence et déterminer les deux principaux points de fuite de l'image
  - Choisir un ensemble de quatre droites  $D'_i \subset D_i$  de telle sorte que les droites de  $D'_i$  participent deux à deux à un même point de fuite et qu'elles aient le score maximal
  - Orienter le quadrilatère formé par ces quatre droites
- Etape de recherche
  - Choisir deux droites  $d_1 \subset D_i - \{D'_i\}$  et  $d_2 \subset D_i - \{D'_i\}$  de telle sorte qu'elles appartiennent à deux groupes différents et qu'elles aient un score maximal
  - Pour chaque sous-ensemble de  $D'_m \subset D_m$  équivalent à  $D'_i$  Faire
    - Calculer une homographie  $h$
    - Attribuer à  $h$  un score  $s_h$  de pertinence
      - Pour chaque droite  $d \subset D_m - \{D'_m\}$  Faire
        - Calculer  $dist(dd_1) = \overrightarrow{\|dd_1\|}$  et  $dist(dd_2) = \overrightarrow{\|dd_2\|}$
        - $s_{d_1} = \min dist(dd_1)$  et  $s_{d_2} = \min dist(dd_2)$
        - $s_h = \max(s_{d_1}, s_{d_2})$
  - Parcourir l'ensemble des  $s_h$  par ordre croissant
    - Calibrer la caméra à partir de  $s_h$
    - Si les paramètres de cette caméra sont cohérents alors
      - $h$  correspond à l'homographie recherchée

### 2.2.3 Résultats

Nous avons proposé un schéma général qui répond au problème de mise en correspondance et qui permet donc le recalage automatique entre une image et un modèle. Nous avons également étayé ce schéma par l'illustration d'un cas pratique. Toutefois, ce dernier a mis en évidence deux verrous scientifiques : la classification des droites de l'image en fonction de leurs points de fuite et le calcul des paramètres de la caméra à partir de la transformation plan à plan liant l'image et le modèle. Dans les parties suivantes, nous apportons une réponse à ces verrous.

## 2.3 Classification de droites en fonction de leurs points de fuite

La mise en correspondance d'éléments caractéristiques entre une image et un modèle implique, dans notre cadre applicatif, la détection des faisceaux de droites de l'image. En effet, les intersections de quatre droites (deux droites de deux faisceaux) permettent de déterminer quatre points qui vérifient les conditions minimales pour le calcul de la transformation liant l'image et le modèle. Les faisceaux de l'image sont composés de droites sécantes en un même point : le point de fuite. Aussi, après un bref rappel et un état des lieux sur la détection des points de fuite, nous proposons une approche entièrement géométrique pour classer un ensemble de droites de l'image en fonction de leurs points de fuite.

La détection des points de fuite dans une image est un problème qui a donné lieu à de nombreuses publications dans des domaines d'applications divers tels que la reconstruction de scène sportive ou encore l'architecture. En effet, les environnements construits par l'homme ont souvent comme caractéristique de contenir de nombreuses droites parallèles ou perpendiculaires. Par exemple, les portes et les fenêtres d'une scène intérieure, les bâtiments et les routes d'une scène extérieure (ou encore dans notre cadre applicatif le tracé d'un terrain de football). Une image 2D de cette scène 3D, acquise par un modèle parfait de caméra de type sténopé (cf. Modèle géométrique de caméra), réalise une projection parfaite de cette scène. Un ensemble de droites parallèles d'une scène 3D se projette alors dans une image 2D en un ensemble de droites concourantes en un même point appelé point de fuite. Ce dernier peut être un point à distance finie ou un point à l'infini (Figure 29). La détection et l'analyse des points de fuite d'une image donnent une information forte sur la structure de la scène qui est généralement exploitée pour extraire les paramètres de la prise de vue.

Déterminer les points de fuite dans une image consiste donc à déterminer des ensembles de droites concourantes. Ces ensembles de droites sont composés de trois droites ou plus et sont appelés « faisceaux de droites ». Les méthodes de détection de points de fuite se confrontent à deux problèmes majeurs :

- la recherche des points qui s'effectue dans un espace non borné (les points de fuite se situent très souvent « au-delà » de l'espace image) ;
- les diverses approximations à la fois liées au système d'acquisition (taille de la scène, distorsion de la lentille, bruits, etc.) et à la détection des droites de l'image (discrétisation, seuillage, etc.)

Aussi, la qualité du recalage dépend en grande partie de la bonne détection des points de fuite. Il est donc indispensable de proposer une détection robuste des points de fuite qui tient compte des difficultés présentées ci-dessus.

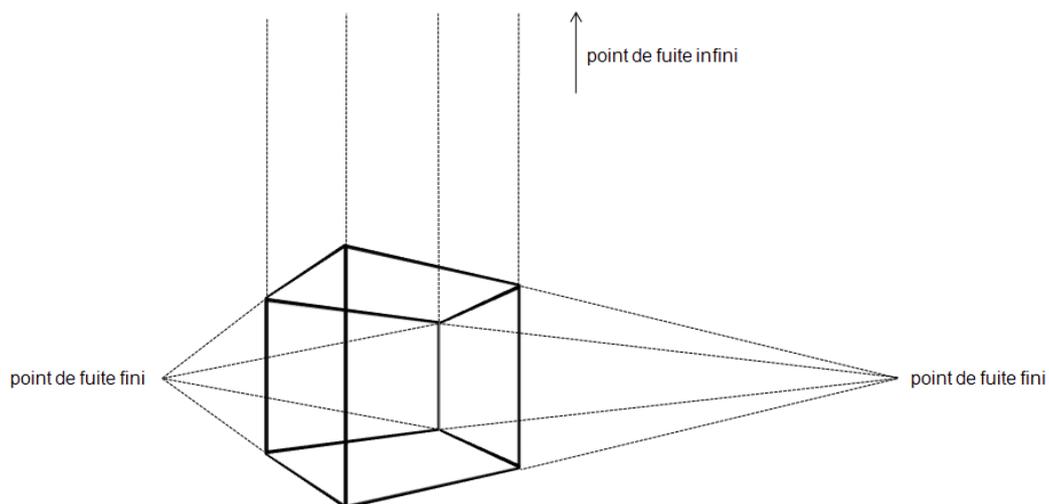


Figure 29 – Points de fuite à distance finie et infinie

Dans la suite, nous présentons les méthodes existantes pour déterminer les faisceaux formés par les droites d'une image. Nous montrons que ces approches s'adaptent difficilement à notre cadre applicatif. Nous proposons alors une méthode de détection de points de fuite entièrement géométrique répondant à notre problématique.

### 2.3.1 Détection de points de fuite

Généralement, les méthodes de détection de points de fuite dans une image utilisent, au préalable, une détection de droites ou de segments (Canny, 1986) (Deriche, 1987) (Duda, et al., 1972) (Le Troter, 2006). Ces méthodes posent ensuite comme hypothèse qu'une grande partie des droites de l'image forme des faisceaux. Elles procèdent ensuite en deux étapes :

- une étape d'accumulation : les droites qui partagent un même point de fuite sont groupées ;
- une étape de recherche : les groupes dominants sont identifiés.

La première étape nécessite la définition d'un espace d'accumulation ainsi qu'une discrétisation de cet espace (cellules d'accumulations). L'espace image peut être utilisé comme un espace d'accumulation (Quan, et al., 1989) (van den Heuvel, 1998) (Rother, 2000). Ce dernier n'est pas borné, ce qui est problématique dans le choix de sa discrétisation (notamment pour les points de fuite à l'infini). L'espace d'accumulation le plus largement utilisé est la sphère de Gauss (Barnard, 1983) (Magee, et al., 1984) (Lutton, et al., 1994) (Tuytelaars, et al., 1998) (Shufelt, 1999) qui a été introduite par S. T. Barnard (Barnard, 1983) et complétée par M. G. Magee et J. K. Aggarwal (Magee, et al., 1984). Contrairement à l'espace image, la sphère de Gauss définit un espace borné. En effet, chaque droite de l'image associée au centre de la sphère de Gauss définit un plan. L'intersection de ce plan avec la sphère de Gauss correspond alors à un grand cercle. Ainsi, un point de fuite, à distance finie ou infinie, correspond à l'intersection de deux grands cercles (Figure 30).

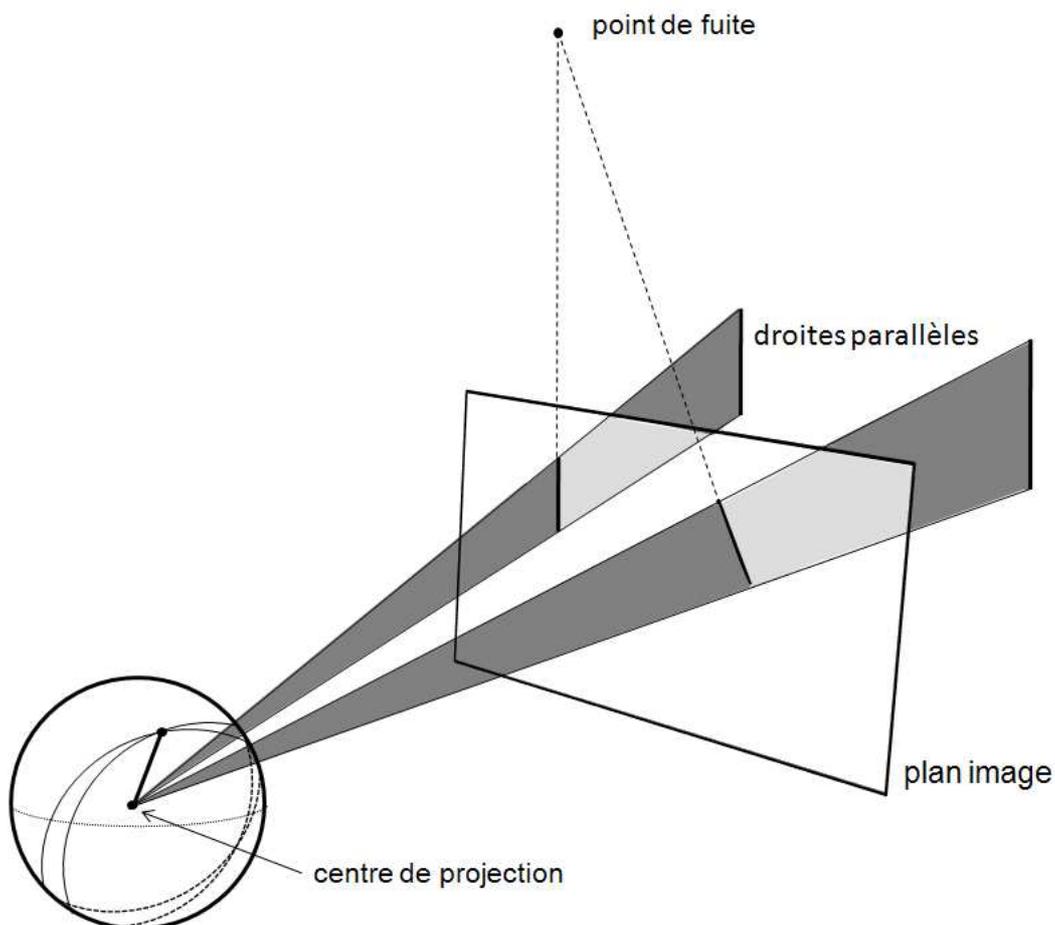


Figure 30 – La sphère de Gauss comme espace d'accumulation

L'utilisation de la sphère de Gauss comme espace d'accumulation pour la détection des points de fuite nécessite des calculs lourds, mais elle est considérée comme très précise. Néanmoins, un certain nombre de problèmes ont été identifiés, principalement liés à la discrétisation de l'espace (par exemple aux pôles). Ces problèmes ont été abordés par Lutton *et al.* (Lutton, et al., 1994) qui ont proposé une méthode tenant compte des erreurs liées à l'espace d'accumulation. Ils utilisent une transformée de Hough dans un espace discrétisé de façon « semi-régulière ». Ces travaux ont été complétés par ceux de T. Tuytelaars *et al.* (Tuytelaars, et al., 1998) qui proposent une transformée de Hough itérative et hiérarchique (Transformée de Hough en Cascade).

L'étape de recherche consiste quant à elle à déterminer quels sont les groupes dominants, i.e les points de fuite qui s'accumulent le plus. Généralement, on procède de la façon suivante : le point de fuite dominant – la cellule d'accumulation qui contient le plus de segments – est détecté puis, on enlève les segments associés à ce point de fuite. Ce processus est répété itérativement jusqu'à ce que le nombre de segments définissant un point de fuite soit en dessous d'un seuil fixé au préalable. D'autres critères ont également été utilisés pour faciliter la recherche de points de fuite dominants et sont le plus souvent :

- l'ajout de connaissances (Quan, et al., 1989) (van den Heuvel, 1998) (Rother, 2000) (Shufelt, 1999) issu de la scène 3D telle que la prise de vue ou encore la géométrie des objets,
- la modélisation d'erreurs (Shufelt, 1999) (M., et al., 1999) (Almansa, et al., 2003) (Brillault-O'Mahony, 1991) : les approximations liées à la détection de droites dans l'image.

Une méthode de détection de points de fuite qui n'utilise pas d'espace d'accumulation a été proposée récemment par M. Kalantari *et al.* (Kalantari, et al., 2008). Ces derniers utilisent une propriété géométrique des droites projectives, exploitant le théorème de Chasles-Steiner, qui transforme le problème de détection de points de fuite à partir de segments en problème de détection de cercles dans un nuage de points. Le problème de recherche de points de fuite dans un espace non borné est naturellement écarté, cependant, extraire des cercles à partir d'un ensemble de points est une opération toute aussi complexe. En effet, M. Kalantari *et al.* utilisent une notion d'incertitude (associée à chacun des points) ainsi qu'une méthode robuste de type RanSac (Fischler, et al., 1981) qui nécessite (à l'instar des approches de type accumulation) de nombreuses occurrences du phénomène observé (triplets de points formant un cercle).

Un problème intrinsèque à la recherche des points de fuite dans une image est lié à l'espace de recherche de ces derniers. Ce problème peut-être écarté par l'introduction de méthodes basées sur l'utilisation de la sphère de Gauss comme espace de recherche. Elles utilisent le plus souvent des approches de type accumulation, ce qui entraîne d'autres difficultés (discrétisation, recherche de maxima, etc.). Ainsi, pour être robustes, ces méthodes nécessitent des calculs lourds, l'ajout de connaissances ou encore l'utilisation de modèles d'incertitude. De plus, les méthodes de type accumulation font l'hypothèse implicite que le nombre d'occurrences du phénomène observé (droites qui forment un faisceau) est important ; hypothèse qui n'est pas garantie dans notre cadre applicatif.

### **2.3.2 Détection des faisceaux de droites**

Pour identifier les différents faisceaux de droites d'une image nous proposons une méthode originale qui se décompose en deux étapes :

- déterminer tous les triplets de droites concourantes : nous proposons pour cela une approche totalement géométrique qui permet de s'abstraire des difficultés liées à l'espace de recherche ;
- fusionner les triplets de droites qui participent à un même faisceau : nous utilisons une approche combinatoire pertinente dans le cas d'un nombre de droites restreint, hypothèse valide dans notre cadre applicatif (lignes du terrain et quelques droites issues d'erreurs de détection).

### 2.3.2.1 Détection des triplets de droites concourantes

Notre objectif est de déterminer l'ensemble des triplets de droites concourantes d'une image. Nous présentons dans cette partie une approche entièrement géométrique qui permet de vérifier si trois droites sont concourantes. Celle-ci s'appuie sur le théorème de Thalès et plus particulièrement sur le cercle de Thalès, cas particulier du théorème de Chasles-Steiner. En effet, un triangle inscrit dans un cercle, dont un des cotés est un diamètre, est un triangle rectangle (théorème de Thalès). Ainsi, tout angle inscrit dans un demi-cercle est droit. On définit alors le cercle de Thalès comme étant un demi-cercle dont le diamètre est l'hypoténuse d'un triangle rectangle et on peut alors énoncer le corollaire suivant :

**Corollaire (Thalès)** – Soit trois droites  $\mathcal{D}_1, \mathcal{D}_2$  et  $\mathcal{D}_3$ . Soit le point  $A$  et  $A_1, A_2$  et  $A_3$  ses projetés orthogonaux sur ces droites. Si le cercle formé par  $A_1, A_2$  et  $A_3$  passe par  $A$  alors c'est un cercle de Thalès. Ainsi, les droites  $\mathcal{D}_1, \mathcal{D}_2$  et  $\mathcal{D}_3$  sont sécantes en un même point.

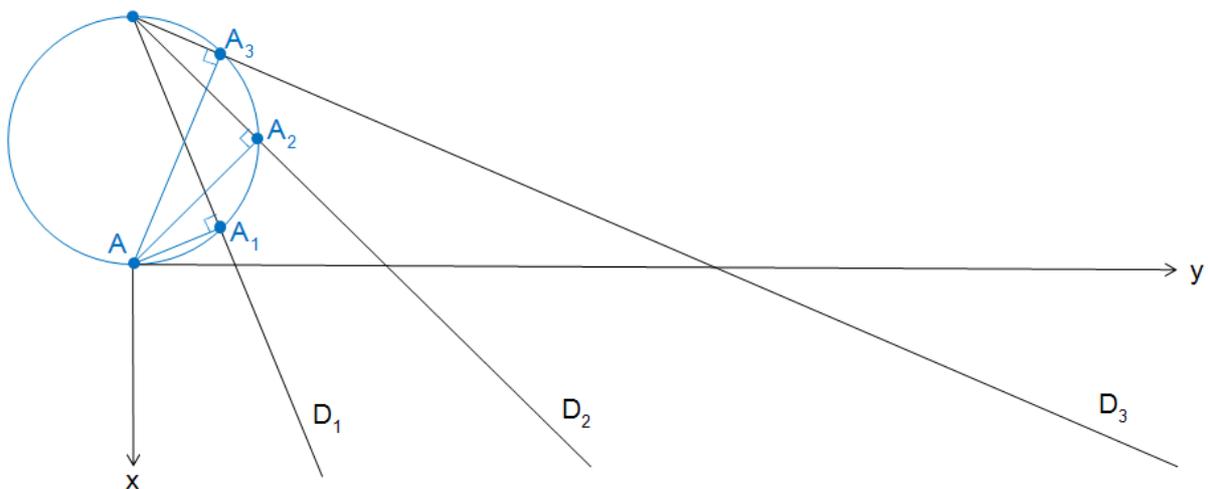


Figure 31 – Application du théorème de Thalès

Déterminer si trois droites sont concourantes revient alors, d'après le corollaire de Thalès, à vérifier si un point  $A$  et ses projetés sur ces trois droites sont sur un même cercle ;  $A$  étant un point quelconque de l'image qui n'appartient à aucune droite. Ce dernier peut être choisi dans l'image de telle sorte qu'il corresponde au point ayant la distance minimale aux droites de l'image (par projection orthogonale) la plus grande et ainsi éviter les cas dégénérés. En effet, si  $A$  appartient à une droite (ou s'il est « proche » d'une droite) alors le corollaire de Thalès est nécessairement vrai (trois points forment un cercle) ; même dans le cas de trois droites non concourantes.

Pour déterminer si trois droites sont concourantes il suffit alors d'estimer si ces dernières vérifient le corollaire de Thalès, i.e. que le point  $A$  et ses projetés sont cocycliques. Cette opération n'est pas complexe, mais peut être source de problèmes numériques. Nous avons donc choisi d'utiliser une méthode géométrique, qui s'appuie sur le théorème de Ptolémée, de telle sorte que l'on évite le calcul explicite des paramètres du cercle (centre et rayon) et qui garantit ainsi la stabilité numérique de nos résultats. En effet, l'application du théorème de Ptolémée nécessite seulement le calcul de distance entre des points (projetés orthogonaux de  $A$  sur les différentes droites de l'image) qui se situent dans l'image (ou dans un voisinage proche), même dans le cas de points issus de droites participant à un point de fuite infini.

**Théorème (Ptolémée)** Un quadrilatère convexe est inscriptible si et seulement si le produit des longueurs des diagonales est égal à la somme des produits des longueurs des cotés opposés, c'est à dire  $AA_1A_2A_3$  est un quadrilatère convexe inscriptible si et seulement si :

$$\|\overrightarrow{AA_2}\| \cdot \|\overrightarrow{A_1A_3}\| = \|\overrightarrow{AA_1}\| \cdot \|\overrightarrow{A_2A_3}\| + \|\overrightarrow{AA_3}\| \cdot \|\overrightarrow{A_1A_2}\|$$

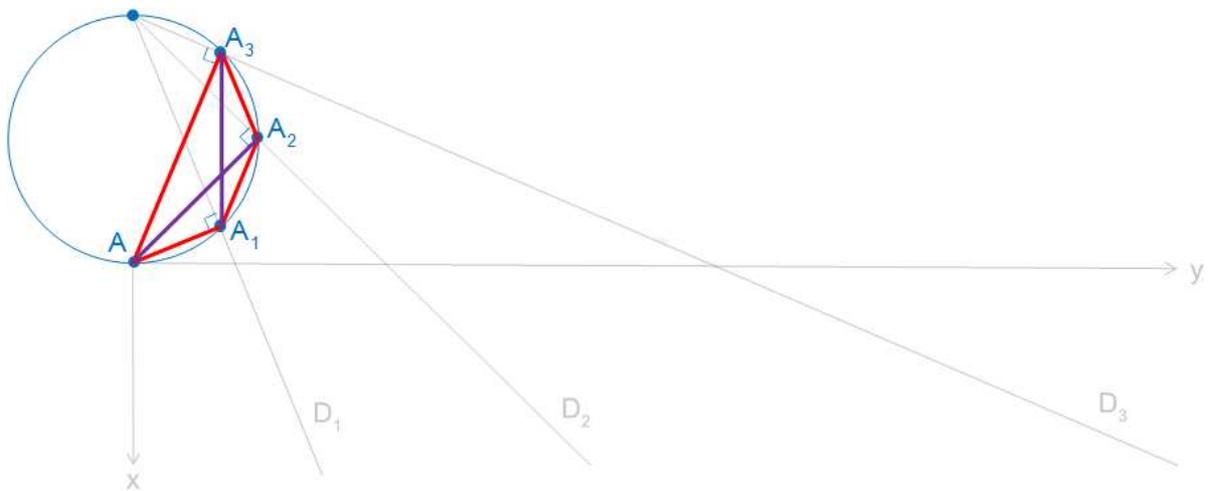


Figure 32 – Théorème de Ptolémée

Remarque : les points de fuite infinis sont « naturellement » traités. En effet, le cercle de Thalès formé par un ensemble de droites (participant à un point de fuite infini) correspond à un « cercle plat » et vérifie donc le théorème de Ptolémée.

Nous avons proposé une approche géométrique qui permet de vérifier si trois droites sont concourantes. Pour cela, on vérifie qu'un point  $A$  et ses projetés sur ces trois droites forment un cercle de Thalès, cette vérification étant réalisée par l'application du théorème de Ptolémée. Notre approche permet ainsi de déterminer, pour un ensemble de droites données, la totalité des triplets de droites concourantes. Ces triplets de droites sont de plus dénombrables. En effet, pour chaque faisceau de  $n$  droites il existe  $C_n^3$  triplets de droites concourantes. Ces derniers peuvent alors participer aux mêmes faisceaux et doivent dans ce cas être identifiés et fusionnés.

### 2.3.2.2 Fusion des faisceaux

Pour identifier les triplets de droites participant au même faisceau il est nécessaire de déterminer un critère de similarité. Une idée intuitive consiste à comparer les cercles de Thalès définis par ces triplets de droites (cf. Détection des triplets de droites concourantes). Mais, cette opération

nécessite l'équation (centre et rayon) de ces cercles et engendre inévitablement des problèmes numériques (problème d'approximation, etc.). Un espace d'accumulation, pour faire émerger les cercles de Thalès équivalents – les triplets de droites concourantes participant au même faisceau – peut être utilisé, mais ne permet pas de garantir la stabilité numérique des résultats obtenus, notamment dans le cas de points de fuite à l'infini (cercle « plat »). De plus, cette approche n'est pertinente que dans le cas où le nombre de cercles de Thalès équivalents est important.

Nous proposons ici une méthode qui permet de déterminer, à partir de l'ensemble des triplets de droites concourantes d'une image, les différents faisceaux de droites ainsi que la totalité des droites qui les composent. Notre méthode s'appuie sur un critère qui permet de discriminer si les droites de deux triplets de droites concourantes participent à un même faisceau. Ce critère utilise la cohérence entre ces triplets et peut être énoncé par : deux triplets de droites concourantes participent au même faisceau s'ils ont des droites communes. En effet, si un faisceau est composé de  $n$  droite alors il existe nécessairement des triplets de droites parmi ces  $n$  droites qui ont des droites communes.

Dans un premier temps, il est primordial de déterminer, pour deux triplets de droites données, combien de droites communes sont nécessaires pour affirmer que les droites de ces triplets participent au même faisceau. Dans un second temps, il est indispensable de vérifier que le critère que nous avons choisi n'est pas trop restrictif et permet de déterminer la totalité des droites des différents faisceaux.

Remarque : notre approche suppose que les triplets de droites concourantes participent à un même point de fuite ; supposition qui peut être erronée, notamment dans le cas d'images avec un nombre de droites trop important. Dans ce cas précis trois droites peuvent potentiellement être concourantes sans pour autant participer à un même point de fuite.

Si deux triplets de droites ne partagent aucune droite alors ces droites peuvent appartenir à un unique faisceau (composé d'au moins six droites) ou à deux faisceaux distincts. De même, si deux triplets partagent une unique droite alors ces droites peuvent appartenir à un unique faisceau ou à deux faisceaux distincts ; ce qui peut être fréquent (Figure 33) notamment dans le cas de droites issues d'erreurs de détection.

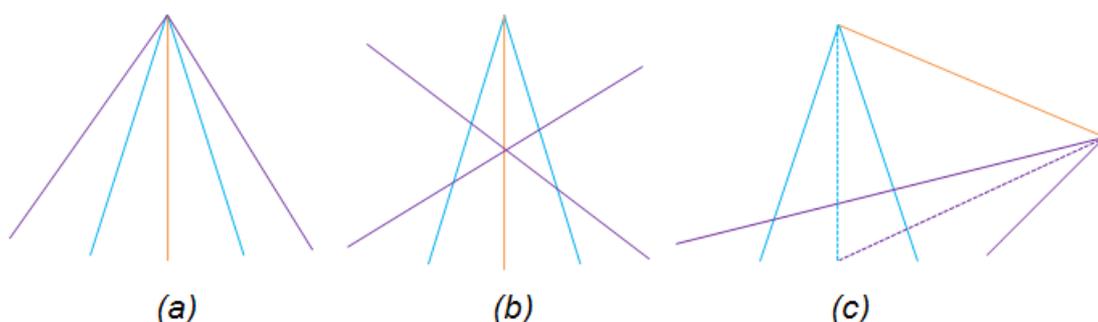


Figure 33 – Exemple de faisceaux partageant une droite commune

La Figure 33 est composée de différents exemples de triplets de droites (bleu et violet) partageant une droite commune (orange) et qui appartiennent au même faisceau (a), à deux faisceaux distincts (b) (c). Le cas (b) représente deux droites (violetttes) issues d'erreurs de détection qui se coupent sur

une troisième droite qui appartient à un faisceau existant. Le cas (c) représente une droite qui appartient à deux faisceaux existants et qui correspond alors à la ligne de fuite.

Ainsi, pour vérifier si les droites de deux triplets de droites participent au même faisceau on utilise les deux propriétés suivantes :

**Propriété 1** – Si un triplet de droites concourantes n’a aucune droite commune avec chacun des autres triplets, alors ce triplet correspond à un faisceau composé de trois droites.

**Propriété 2** – Si un triplet de droites concourantes a deux droites en commun avec d’autres triplets, alors chacune des droites de ces triplets participent au même faisceau.

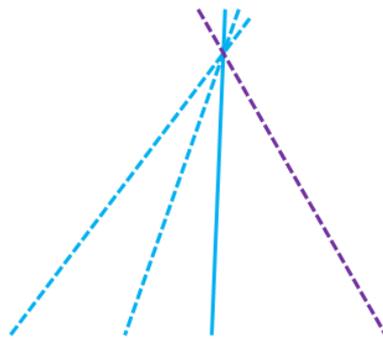


Figure 34 – Justification de la propriété 2

La Figure 34 est composée d’un triplet de droites concourantes (bleu) et d’une quatrième droite (violette) qui associée à deux droites de ce triplet forme un second triplet de droites concourantes (en pointillé). Ces deux droites (en pointillés bleus) sont nécessairement sécantes en un point. Ainsi, le triplet de droites concourantes formé par ces dernières et une troisième (en pointillés violets) ont nécessairement comme intersection commune ce point. Ces quatre droites sont donc concourantes, i.e appartiennent au même faisceau.

Ces deux propriétés sont suffisantes pour déterminer, à partir de l’ensemble des triplets de droites concourantes, la totalité des faisceaux de l’image. En effet, l’unique cas ignoré correspond au cas où deux triplets de droites auraient une unique droite commune. Or, si ces deux triplets de droites appartiennent au même faisceau alors il existe nécessairement plusieurs triplets de droites qui contiennent la totalité des droites de ces deux triplets et qui vérifient deux à deux la propriété 2.

**Démonstration** – Soit  $t_1 = \{d_1, d_2, d_3\}$  et  $t_2 = \{d_1, d_4, d_5\}$  deux triplets de droites concourantes participant au même faisceau de telle sorte  $t_1$  et  $t_2$  partagent une même droite  $d_1$ . Alors,  $t_3 = \{d_1, d_2, d_4\}$  et  $t_4 = \{d_1, d_2, d_5\}$  sont nécessairement des triplets de droites concourantes. Or,  $t_3$  et  $t_4$  partagent deux droites avec  $t_1$ . La propriété 2 permet alors, par l’intermédiaire de  $t_3$  et  $t_4$ , d’affirmer que les droites de  $t_1$  et de  $t_2$  participent au même faisceau.

A partir d’un ensemble de triplets de droites concourantes, l’utilisation de ces deux propriétés permet de déterminer les droites qui appartiennent à un même faisceau. Il est ainsi possible, après l’application de l’approche précédente (cf. Détection des triplets de droites concourantes), de déterminer la totalité des faisceaux de droites contenues dans une image.

Pseudo-code :

- Initialisation
  - $F = \emptyset$  : l'ensemble des faisceaux de l'image
  - $k = 0$  : le nombre d'éléments de  $F$
- Pour chaque triplet de droites  $t_i$  non marqué Faire
  - Pour chaque triplet de droites  $t_j$  ( $j \neq i$ ) non marqué Faire
    - Si  $t_i$  et  $t_j$  ont deux droites communes Alors
      - Si  $t_i$  n'est pas marqué Alors
        - marquer  $t_i$  et  $t_j$
        - Incrémenter  $k$
        - $f_k = t_i \cup t_j$
      - Sinon
        - marquer  $t_j$
        - $f_k = f_k \cup t_j$
    - Si  $t_i$  ne partage aucune droite avec chacun des  $t_j$  Alors
      - marquer  $t_i$
      - Incrémenter  $k$
      - $f_k = t_i$

### 2.3.3 Résultats

Nous avons proposé une méthode qui permet de déterminer la totalité des faisceaux formés par l'ensemble des droites d'une image. Celle-ci se déroule en deux temps :

- l'ensemble des triplets de droites concourantes est détecté à l'aide d'une approche totalement géométrique qui écarte naturellement les problèmes liés aux espaces de recherche (borne, discrétisation, etc.) et qui garantit ainsi la stabilité numérique de nos résultats,
- les triplets de droites sont ensuite fusionnés (pour former les faisceaux recherchés) à l'aide d'une approche basée sur la cohérence entre ces triplets ; cette approche pouvant être utilisée dans le cas où le nombre de droites, et de faisceaux, est peu élevé.

Cette méthode a été testée avec succès dans notre cadre applicatif (images de scène de football issues de caméras non calibrées) ainsi que dans des cas « génériques » qui vérifient les mêmes hypothèses que notre cadre applicatif (chacune des images test sont des projections de scène 3D contenant peu de droites). Nous vous présentons ici des exemples significatifs de ces résultats dans ces cas « génériques » (Figure 35) et dans notre cadre applicatif (Figure 36).

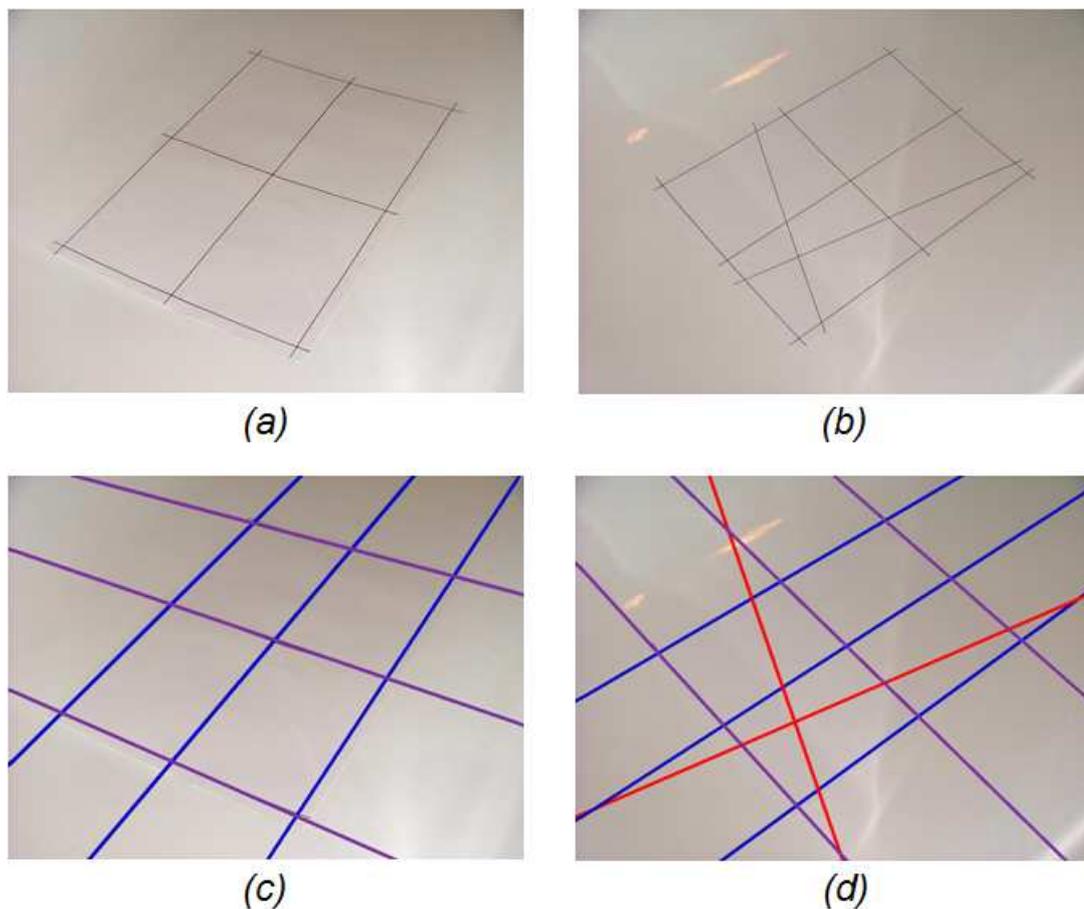


Figure 35 – Détection de faisceaux de droites

La Figure 35 est composée de deux images acquises par un appareil photo numérique (a) (b). Pour chacune d'entre elles nous avons procédé à une détection de droites (approche classique de type *Hough*) puis nous avons déterminé l'appartenance de ces droites à différents faisceaux. Les images (a) et (c) correspondent à un cas simple composé de deux faisceaux de trois droites (rouge et violet). Les images (b) et (d) correspondent à un cas plus ambigu. Ces images contiennent, en plus des deux faisceaux, des droites n'appartenant à aucun faisceau (en rouge).

Remarque : les droites n'appartenant à aucun faisceau sont naturellement écartées par notre méthode. Ainsi, cette dernière peut être utilisée dans le cas d'erreurs d'acquisition ou de mauvaise détection de droites.

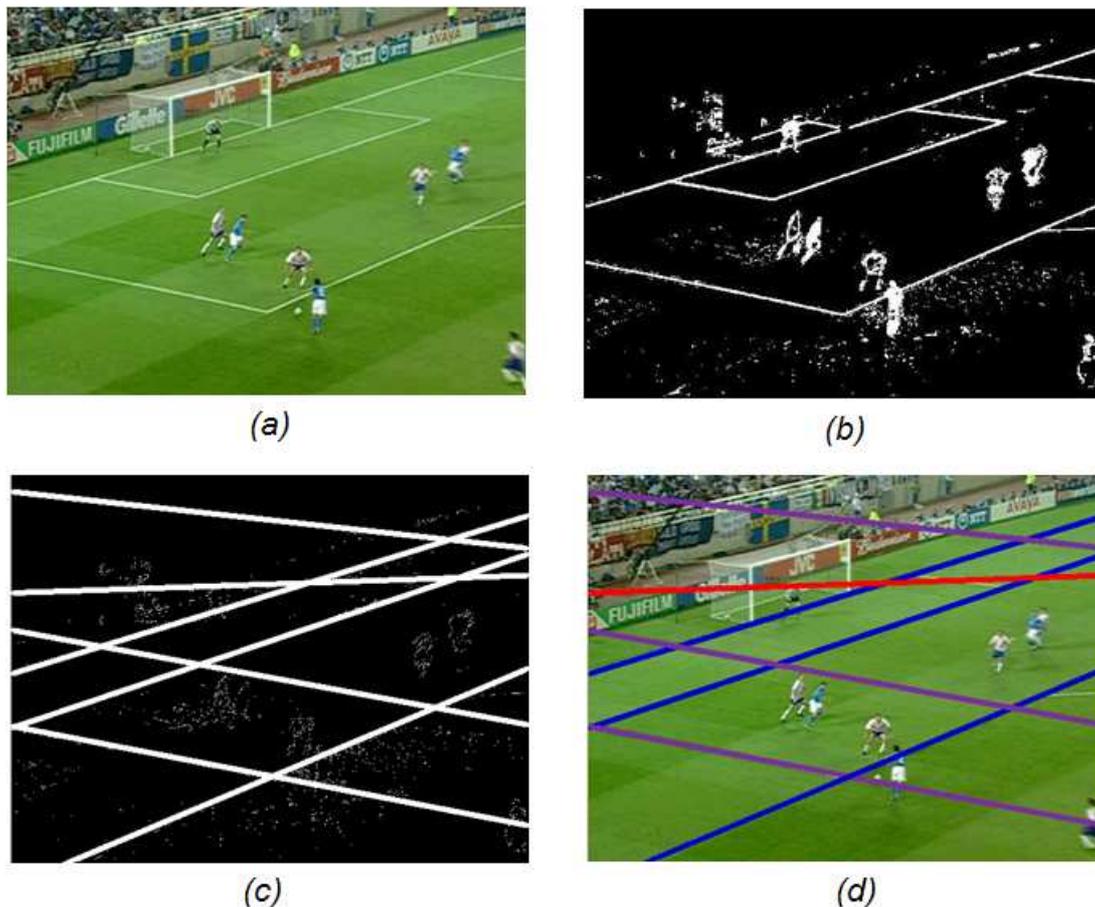


Figure 36 – Exemple de détection de faisceaux de droites

La Figure 36 représente (a) une image source (b) la segmentation couleur de cette image (c) la détection de droites par une approche de type *Hough* (d) l'attribution de ces droites aux différents faisceaux par notre approche géométrique. On peut noter qu'une droite détectée dans l'image ne correspond à aucune droite « réelle », mais cette dernière (en rouge) n'est finalement attribuée à aucun faisceau dans l'image.

Ces résultats permettent d'obtenir une information forte sur les droites contenues par une image. Celle-ci est alors mise à contribution pour sélectionner quatre points dans l'image – les intersections de deux droites de deux faisceaux – qui vérifient les conditions minimales pour réaliser l'étape de mise en correspondance.

Une perspective intéressante à ces travaux consiste à intégrer notre méthode de classification en faisceaux à l'étape de segmentation de droites, i.e détecter dans l'image non pas des droites quelconques, mais participantes à un faisceau de droites concourantes. Pour cela, il est envisageable d'utiliser une approche de détection de droites de type Hough avec comme espace d'accumulation celui classiquement utilisé auquel on aurait rajouté une dimension supplémentaire (en lien avec l'appartenance des droites à des faisceaux). Cette perspective permettrait alors d'étendre notre méthode à des images contenant un nombre de droites plus important.

## **2.4 Calcul des paramètres de la caméra**

L'étape de mise en correspondance implique, de par les spécificités du modèle utilisé dans notre cadre applicatif (SimulFoot), la résolution d'un certain nombre d'incertitudes. Toutefois, le calcul des paramètres de la caméra permet de lever ces ambiguïtés. Aussi, après un bref descriptif des techniques de calibrage de caméra existantes et de leurs limitations, nous exprimerons un certain nombre d'hypothèses nécessaires à notre méthode de calibrage. Nous donnerons ensuite l'expression analytique des positions du point de vue de la caméra et de l'angle de prise de vue.

### **2.4.1 Calibrage de la caméra**

La caractérisation des paramètres d'une caméra à partir de l'image d'une scène 3D est un problème ouvert en « vision par ordinateur » (Fitzgibbon, et al., 1998) (Pollefeys, 2000). Les solutions à ce problème dépendent à la fois de la complexité du dispositif de prise de vue (déformation par l'objectif, position de la caméra par rapport aux éléments de la scène, etc.), du type de scène photographiée (présence ou absence d'objets, scène essentiellement 2D ou utilisant fortement l'espace 3D, etc.) et de la qualité des images obtenues.

De nombreux travaux ont été publiés pour déterminer les différents paramètres de la caméra. Dans le cas général, ces solutions induisent une complexité importante ou la nécessité de disposer d'un ensemble d'informations qui n'est pas toujours connue (objets appariés en nombre important, par exemple). Ainsi, l'une des techniques de calibrage les plus utilisées est la méthode proposée par R. Y. Tsai (Tsai, 1986) (Tsai, 1987). Il s'appuie sur un modèle de caméra type « sténopé » (cf. Prérequis) et il cherche à estimer onze paramètres (cinq paramètres intrinsèques et six paramètres extrinsèques). Cette méthode exprime le processus de formation d'une image sous la forme de différentes équations et permet, à partir d'un ensemble de points dans « l'espace monde » et leur correspondants dans « l'espace image », de déterminer l'ensemble de ces paramètres. Ce calcul nécessite néanmoins un grand nombre de points pour être robuste.

Des recherches ont également permis d'améliorer la méthode de R. Y. Tsai, par exemple en prenant compte de l'ensemble du processus de calibrage de la caméra (Heikkila, et al., 1997) ; principalement l'extraction des points dans l'image, l'appariement avec le modèle et la correction d'erreurs d'acquisition. D'autres types d'approches ont également été proposés pour calibrer la caméra. On peut citer entre autres celles de Z. Zhang (Zhang, 1996) (Zhang, 2000) qui utilise des cibles, ou encore celle de C. Tomasi (Tomasi, et al., 1992), qui utilise une caméra en mouvement. Mais, ces méthodes sont incompatibles avec notre cadre applicatif.

### **2.4.2 Calcul des paramètres simplifiés de la caméra**

L'état de l'art sur les techniques de calibrage de caméra a mis en évidence un certain nombre de déficiences. En effet, ces techniques impliquent une connaissance trop importante des relations entre l'image et le modèle ou encore l'insertion d'information (par exemple l'ajout de cibles) impossible à mettre en œuvre dans notre cadre applicatif. Aussi, nous avons décidé d'ajouter un certain nombre d'hypothèses (formulé dans la section suivante) pour simplifier le problème de calibrage de caméra, mais nous verrons que ces hypothèses correspondent à un cas très général de prise de vue.

### 2.4.2.1 Hypothèses et expression des contraintes

Les hypothèses que nous utilisons peuvent être formulées par :

- la scène est essentiellement 3D, ou plutôt, nous disposons de certains éléments de repérage situés dans un plan donné de la scène 3D ;
- la caméra est suffisamment éloignée de la scène filmée pour que l'on puisse assimiler l'image à une projection conique de celle-ci sur le plan ;
- l'axe vertical de l'image sur le plan de projection se trouve dans un plan vertical.

Les deux premières hypothèses nous permettent d'énoncer que les points du plan 2D de la scène et ceux de l'image se correspondent bijectivement par l'intermédiaire d'une homographie, représentée donc dans l'espace homogène par une matrice de trois lignes et de trois colonnes.

Nous supposons également que la caméra se trouve dans un plan vertical, ce que nous pourrions qualifier de position « tête droite » et qui est généralement le cas des caméras de télévision assujetties à un support mécanique (ce n'est pas le cas, par contre, des « caméras au poing »).

La dernière hypothèse revient à supposer que, dans le plan  $P$  de projection correspondant à l'image (en 3D), le premier vecteur de base ( $\vec{V}_1$  sur les dessins ci-dessous) est « horizontal » (ou plutôt parallèle au plan  $xOy$ ). L'origine  $C$  du repère dans le plan  $P$  est la projection orthogonale du point  $S$  (centre de la projection conique et position de la caméra).

Considérant que nous disposons de la matrice de transformation nous proposons dans cette communication d'en extraire la position et l'ouverture de la caméra en les exprimant analytiquement. Les paramètres d'observation de la scène sont, par exemple (car nous pouvons choisir d'autres jeux de paramètres qui soient équivalents) : la position du point de vue  $S(x_S, y_S, z_S)$  – c'est-à-dire la caméra – la position d'un point visé  $M(x_M, y_M, z_M)$  – nous choisirons pour  $M$  le point du plan  $xOy$  correspondant au centre  $C$  de l'écran – et l'angle d'ouverture  $\theta$  ainsi que cela est illustrée sur le dessin ci-dessous (Figure 37).

Remarque : si nous n'avions pas fait l'hypothèse « tête droite », nous aurions également dû déterminer un autre paramètre, qui est l'angle de rotation autour de l'axe d'observation  $SC$ .

Remarque :  $x_S$ ,  $y_S$  et  $z_S$  sont les seules vraies inconnues ; en effet, le point  $M$  – le point de visée – est immédiatement obtenu comme transformé du point  $C$  et l'angle  $\theta$  s'obtient aisément à partir de tous ces éléments, comme nous le préciserons à la fin de cette section.

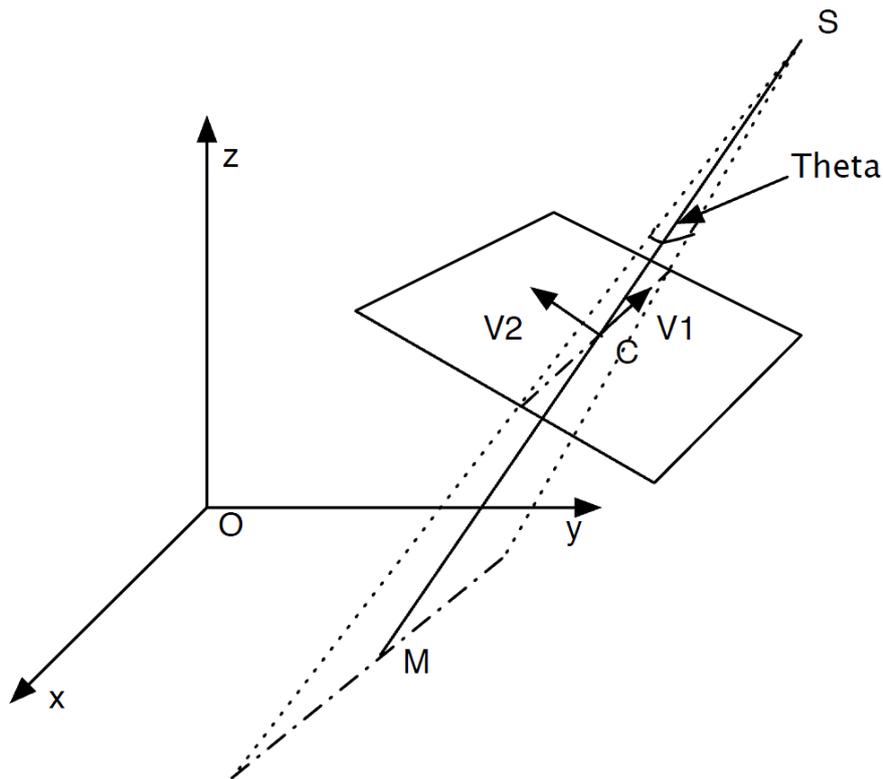


Figure 37 – Paramètres d'observation choisis :  $S$  la position de la caméra,  $M$  le projeté de  $C$  dans le plan  $xOy$  et  $\theta$  l'ouverture de la caméra

Le problème peut donc se ramener à l'énoncé suivant : nous disposons d'une matrice  $3 \times 3$  et nous recherchons les trois inconnues  $x_S$ ,  $y_S$  et  $z_S$  qui caractérisent la position de la caméra (l'image ayant une taille de  $2a \times 2b$ ). La méthode que nous proposons consiste à considérer un ensemble de couple  $(C_i, M_i)$ ,  $C_i$  étant un point de l'image (ou plutôt de son plan) et  $M_i$  le point correspondant dans le plan  $xOy$ , et de déterminer les valeurs des trois inconnues  $x_S$ ,  $y_S$  et  $z_S$  pour lesquelles ces droites sont concourantes. Nous associons au centre  $C$  et aux quatre coins  $C_1, C_2, C_3$  et  $C_4$  de l'image (en 3D) les cinq points  $M, M_1, M_2, M_3$  et  $M_4$  du plan  $xOy$  du modèle.

Pour des raisons d'homogénéité dans l'expression des calculs, nous allons introduire une quatrième variable,  $k$ , qui exprime l'éloignement relatif du plan de projection par rapport au point de vue,  $k = \frac{CS}{MS}$ , qui s'éliminera naturellement dans les développements que nous allons détailler par la suite.

Exprimer que les droites  $M_i C_i$  sont concourantes (en  $S$ ) revient à exprimer que les droites  $M_i C_i$  et  $M_i S$  sont identiques (ou que les vecteurs associés sont proportionnels). Nous obtiendrons ainsi les quatre équations qui nous permettront, si elles ne sont pas liées, d'obtenir les quatre inconnues recherchées et donc  $x_S$ ,  $y_S$  et  $z_S$ .

Dans le repère associé à l'écran, dont l'origine est le centre  $C$ , dont l'axe  $CX$ , « orienté vers le haut » et supporté par le vecteur unitaire  $\vec{V}_1$  (horizontal), et l'axe  $CY$ , « orienté vers le haut » et supporté par le vecteur unitaire  $\vec{V}_2$ , qui se trouve dans le plan vertical contenant  $C$  et  $M$ , les quatre coins de l'image ont les coordonnées suivantes :



$$\cos(\alpha) = \frac{-(x_1 + x_3)}{\sqrt{(x_1 + x_3)^2 + (y_1 + y_3)^2}} \quad \sin(\alpha) = \frac{-(y_1 + y_3)}{\sqrt{(x_1 + x_3)^2 + (y_1 + y_3)^2}}$$

### 2.4.2.3 Résolution

A présent, nous pouvons considérer, après rotation d'axe  $Mz$  et d'angle  $-\alpha$ , que le point de vue  $S$  est dans le plan  $xMz$  – ses coordonnées sont alors  $(x, 0, z)$  – et que les points  $M_1, M_2, M_3$  et  $M_4$  s'expriment en fonction de quatre paramètres  $X_1, Y_1, X_2$  et  $Y_2$  soit :

$$M_1 \begin{pmatrix} X_1 \\ Y_1 \end{pmatrix} \quad M_2 \begin{pmatrix} X_2 \\ Y_2 \end{pmatrix} \quad M_3 \begin{pmatrix} X_1 \\ -Y_1 \end{pmatrix} \quad M_4 \begin{pmatrix} X_2 \\ -Y_2 \end{pmatrix}$$

Remarque :  $X_1 < 0, Y_1 > 0, X_2 > 0, Y_2 > 0, -(X_1) > X_2$  et  $Y_1 > Y_2$ .

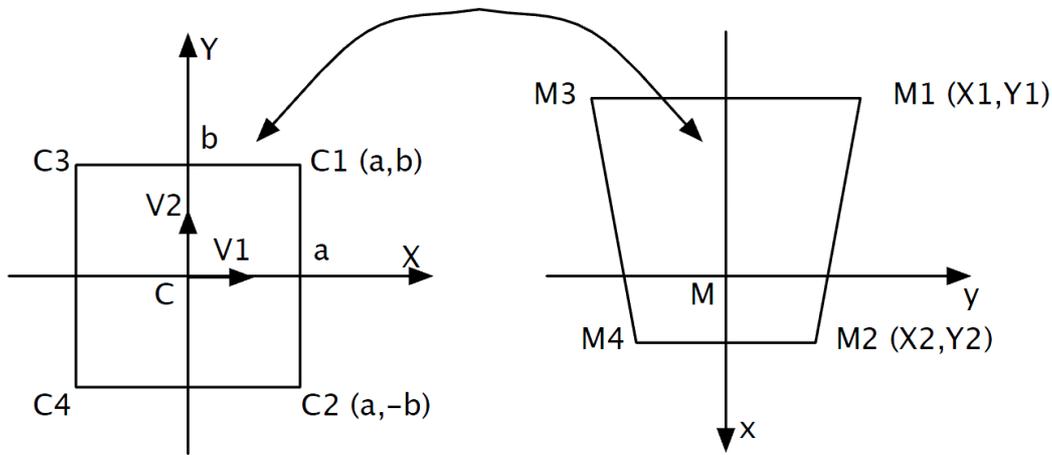


Figure 40 – Illustration des points  $M_i$  après une rotation d'angle  $-\alpha$  dans le repère  $xMz$

Nous cherchons donc à éliminer la valeur des trois inconnues  $x, z$  et  $k$  (ou plutôt de  $x$  et  $z$  en éliminant  $k$ ) définies par :

$$\vec{CS} = k \cdot \vec{MS} \text{ et } S \begin{pmatrix} x \\ 0 \\ z \end{pmatrix} \text{ dans le repère } xMz \text{ après rotation } -\alpha.$$

Nous pouvons, toujours dans ce repère, définir le vecteur unitaire  $\vec{U}$  porté par  $MS$  et dirigé de  $M$  vers  $S$ , le vecteur unitaire  $\vec{V}_1$  du repère lié à l'image et le vecteur unitaire  $\vec{V}_2$ , produit vectoriel des deux précédents :

$$\vec{U} \begin{pmatrix} \frac{x}{\sqrt{x^2 + z^2}} \\ 0 \\ \frac{z}{\sqrt{x^2 + z^2}} \end{pmatrix} \quad \vec{V}_1 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \vec{V}_2 = \vec{U} \wedge \vec{V}_1 = \begin{pmatrix} \frac{-z}{\sqrt{x^2 + z^2}} \\ 0 \\ x \end{pmatrix}$$

Cela nous permet d'exprimer les vecteurs  $\vec{M_1C_1}, \vec{M_1S}, \vec{M_2C_2}$  et  $\vec{M_2S}$  à partir de  $x, z$  et  $k$  :

$$\vec{M_1C_1} = \vec{MC} + a\vec{V}_1 + b\vec{V}_2 - \vec{MM_1} = (1 - k)\vec{MS} + a\vec{V}_1 + b\vec{V}_2 - \vec{MM_1}$$

$$\overrightarrow{M_1 C_1} \begin{pmatrix} (1-k)x - \frac{bz}{\sqrt{x^2+z^2}} - X_1 \\ a - Y_1 \\ (1-k)z + \frac{bx}{\sqrt{x^2+z^2}} \end{pmatrix} \quad \overrightarrow{M_1 S} \begin{pmatrix} x - X_1 \\ -Y_1 \\ z \end{pmatrix}$$

et en remplaçant  $b$  par  $-b$ ,  $X_1$  par  $X_2$  et  $Y_1$  par  $Y_2$ , nous obtenons aussi :

$$\overrightarrow{M_2 C_2} \begin{pmatrix} (1-k)x + \frac{bz}{\sqrt{x^2+z^2}} - X_2 \\ a - Y_2 \\ (1-k)z - \frac{bx}{\sqrt{x^2+z^2}} \end{pmatrix} \quad \overrightarrow{M_2 S} \begin{pmatrix} x - X_2 \\ -Y_2 \\ z \end{pmatrix}$$

La proportionnalité de ces vecteurs deux à deux nous donnera deux fois deux équations (par exemple, proportionnalité des  $x$  et des  $y$  d'une part, des  $y$  et des  $z$  d'autres part).

Remarque : il est inutile de faire de même sur les points 3 et 4 car on aurait, par symétrie, les mêmes équations, et nous avons déjà exploité la symétrie pour trouver  $\alpha$ .

En exprimant la proportionnalité des composantes  $x$  et  $y$  caractérisant l'alignement des points  $M_1$ ,  $C_1$  et  $S$  nous obtenons donc les équations suivantes :

$$\begin{aligned} \left[ (1-k)x - \frac{bz}{\sqrt{x^2+z^2}} - X_1 \right] (-Y_1) &= (a - Y_1)(x - X_1) \\ (1-k)x &= \frac{bz}{\sqrt{x^2+z^2}} + X_1 - \frac{(a - Y_1)(x - X_1)}{Y_1} \end{aligned} \quad (1)$$

Et pour les points  $M_2$ ,  $C_2$  et  $S$  :

$$(1-k)x = \frac{-bz}{\sqrt{x^2+z^2}} + X_2 - \frac{(a - Y_2)(x - X_2)}{Y_2} \quad (2)$$

De même, la proportionnalité des composantes  $y$  et  $z$  caractérisant l'alignement des points  $M_1$ ,  $C_1$  et  $S$  d'une part, et  $M_2$ ,  $C_2$  et  $S$  d'autre part, nous donne les équations suivantes :

$$\begin{aligned} \left[ (1-k)z + \frac{bx}{\sqrt{x^2+z^2}} \right] (-Y_1) &= (a - Y_1)z \\ (1-k)z &= \frac{-bx}{\sqrt{x^2+z^2}} - \frac{(a - Y_1)z}{Y_1} \end{aligned} \quad (3)$$

$$(1-k)z = \frac{bx}{\sqrt{x^2+z^2}} - \frac{(a - Y_2)z}{Y_2} \quad (4)$$

En identifiant les deux deuxièmes membres des équations (1) et (2), nous éliminons  $k$  et obtenons une équation à deux inconnues ( $x$  et  $z$ ). En faisant de même avec les équations (3) et (4), nous

obtenons une deuxième équation à deux inconnues ( $x$  et  $z$ ). Lorsque nous aurons résolu ce système, non linéaire, de deux équations à deux inconnues, nous obtiendrons les valeurs de  $x$  et de  $z$ .

Remarque : si nous souhaitons avoir la valeur de  $k$ , nous pouvons l'obtenir en remplaçant ces valeurs de  $x$  et  $z$  dans l'une quelconques des quatre équations. Par exemple avec l'équation (4) :

$$k = \frac{a}{Y_2} - \frac{bx}{z\sqrt{x^2 + z^2}} \quad (\text{Expression de } k \text{ en fonction de } x \text{ et de } z)$$

Remarque : cette expression est « sans dimension » à condition d'exprimer les dimensions de l'image,  $a$  et  $b$ , (habituellement en nombre de pixels) dans la même unité que les coordonnées des points du modèle, ce qui nécessite une attention particulière.

Considérons les équations (1) et (2) et égalons les deux deuxièmes membres :

$$\begin{aligned} \frac{bz}{\sqrt{x^2 + z^2}} + X_1 - \frac{(a - Y_1)(x - X_1)}{Y_1} &= \frac{-bz}{\sqrt{x^2 + z^2}} + X_2 - \frac{(a - Y_2)(x - X_2)}{Y_2} \\ \frac{2bz}{\sqrt{x^2 + z^2}} &= X_2 - X_1 + \frac{(a - Y_1)(x - X_1)}{Y_1} - \frac{(a - Y_2)(x - X_2)}{Y_2} \\ 2bY_1Y_2z - a[(Y_2 - Y_1)x + (X_2Y_1 - X_1Y_2)]\sqrt{x^2 + z^2} &= 0 \end{aligned} \quad (5)$$

Et faisons de même avec les équations (3) et (4)

$$\begin{aligned} \frac{bx}{\sqrt{x^2 + z^2}} - \frac{(a - Y_2)z}{Y_2} &= \frac{-bx}{\sqrt{x^2 + z^2}} - \frac{(a - Y_1)z}{Y_1} \\ \frac{2bx}{\sqrt{x^2 + z^2}} &= \frac{(a - Y_2)z}{Y_2} - \frac{(a - Y_1)z}{Y_1} \\ 2bY_1Y_2x + a(Y_2 - Y_1)z\sqrt{x^2 + z^2} &= 0 \end{aligned} \quad (6)$$

Nous allons à présent « combiner » ces deux équations (5 et 6) pour en dériver deux autres équations, toujours indépendantes, mais dont l'expression est plus simple. Ainsi, si l'on « isole »  $\sqrt{x^2 + z^2}$  du reste de l'expression dans chacune des équations et que nous identifions les résultats, nous obtenons :

$$(Y_2 - Y_1)z^2 + [(Y_2 - Y_1)x + (X_2Y_1 - X_1Y_2)]x = 0 \quad (7)$$

Soit, en reportant dans l'équation (6) :

$$2bY_1Y_2x - a[(Y_2 - Y_1)x + (X_2Y_1 - X_1Y_2)]x \sqrt{1 + \frac{x^2}{z^2}} = 0$$

$$2bY_1Y_2x - a[(Y_2 - Y_1)x + (X_2Y_1 - X_1Y_2)]x \sqrt{\frac{(X_2Y_1 - X_1Y_2)x}{[(Y_2 - Y_1)x + (X_2Y_1 - X_1Y_2)]x}} = 0$$

$$2bY_1Y_2 - a\sqrt{(X_2Y_1 - X_1Y_2)[(Y_2 - Y_1)x + (X_2Y_1 - X_1Y_2)]} = 0$$

Remarque :  $(X_2Y_1 - X_1Y_2)$  ne peut pas être égal à 0 car  $X_1$  est négatif et  $X_2, Y_1$  et  $Y_2$  sont positifs.

$$x = \frac{1}{(Y_2 - Y_1)} \left[ \frac{1}{(X_2Y_1 - X_1Y_2)} \left( \frac{2bY_1Y_2}{a} \right)^2 - (X_2Y_1 - X_1Y_2) \right]$$

$$x = \frac{(2bY_1Y_2)^2 - a^2(X_2Y_1 - X_1Y_2)^2}{a^2(Y_2 - Y_1)(X_2Y_1 - X_1Y_2)} \quad (\text{Expression de } x)$$

Remarque : cette expression dépend des valeurs de  $a$  et de  $b$ , mais pas de l'unité dans laquelle elles sont exprimées puisque numérateur et dénominateur sont homogènes au carré de ces valeurs.

En reportant ces résultats dans l'équation (7), nous obtenons la valeur de  $z$  :

$$z = \sqrt{\frac{[(Y_2 - Y_1)x + (X_2Y_1 - X_1Y_2)]x}{(Y_1 - Y_2)}} \quad (\text{Expression de } z \text{ en fonction de } x)$$

Remarque : la valeur de  $-z$  est également une solution

Il nous reste, comme nous l'avons mentionné en début de section, à calculer l'angle  $\theta$  qui caractérise l'ouverture. Soit  $M_{12}$  le point d'intersection du segment  $M_1M_2$  et de l'axe  $My$ . Ce point  $M_{12}$  est la projection sur le plan  $xOy$  du point  $C_{12}$  de coordonnées  $(a, 0)$  dans le repère  $(C, V_1, V_2)$ . Ce point  $M_{12}$  a donc pour abscisse 0 (il est sur l'axe  $My$ ) et pour ordonnée :

$$y_{12} = \frac{Y_2(X_2 - X_1) - X_2(Y_2 - Y_1)}{(X_2 - X_1)}$$

Le triangle  $SMM_{12}$  est rectangle en  $M$ , et l'angle  $(SM, SM_{12})$  vaut  $\frac{\theta}{2}$ . En conséquence :

$$\theta = 2 \cdot \arctan \left[ \frac{Y_2(X_2 - X_1) - X_2(Y_2 - Y_1)}{(X_2 - X_1)\sqrt{x^2 + z^2}} \right] \quad (\text{Expression de } \theta)$$

### Synthèse des calculs et application

Nous disposons d'une matrice 3x3 correspondant à une projection conique (en coordonnées homogènes) d'un plan sur un plan, les repères associés étant un repère absolu 3D et un repère lié à l'image. Nous cherchons les coordonnées  $x_s, y_s$  et  $z_s$  de la position de la caméra (centre de projection) et son ouverture  $\theta$ . Nous procédons alors de la manière suivante :

- nous calculons les images des cinq points qui sont le centre  $C$  et les quatre coins  $C_i$  de l'image, soient  $M$  et  $M_i$  ces points (de coordonnées  $x_M, y_M, x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$ ) ;

- nous calculons  $X_1, Y_1, X_2$  et  $Y_2$  à partir de ces valeurs :  $Y_1$  est égal à  $M_1M_3/2$  et  $Y_2$  à  $M_2M_4/2$  ; quant à  $X_1$  et  $X_2$ , nous les obtenons par  $X_1^2 + Y_1^2 = MM_1^2$  et  $X_2^2 + Y_2^2 = MM_2^2$  en tenant compte du fait que  $X_1$  est négatif et  $X_2$  positif ;

$$x = \frac{(2bY_1Y_2)^2 - a^2(X_2Y_1 - X_1Y_2)^2}{a^2(Y_2 - Y_1)(X_2Y_1 - X_1Y_2)} \quad y = 0 \quad z = \sqrt{\frac{[(Y_2 - Y_1)x + (X_2Y_1 - X_1Y_2)]x}{(Y_1 - Y_2)}}$$

$$\cos(\alpha) = \frac{-(x_1 + x_3)}{\sqrt{(x_1 + x_3)^2 + (y_1 + y_3)^2}} \quad \sin(\alpha) = \frac{-(y_1 + y_3)}{\sqrt{(x_1 + x_3)^2 + (y_1 + y_3)^2}}$$

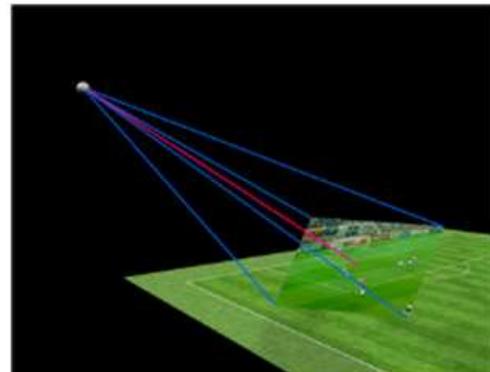
$$x_s = x_M + x \cdot \cos(\alpha) \quad y_s = y_M + x \cdot \sin(\alpha) \quad z_s = z$$

- nous obtenons l'ouverture  $\theta$  :

$$\theta = 2 \cdot \arctan \left[ \frac{Y_2(X_2 - X_1) - X_2(Y_2 - Y_1)}{(X_2 - X_1)\sqrt{x^2 + z^2}} \right]$$



(a)



(b)

Figure 41 – Calibrage de caméra

La Figure 41 est un exemple de calibrage de caméra à partir d'une image issue d'une caméra ayant acquis une scène de football : l'image source (a), l'illustration des paramètres de la caméra ; position (boule blanche), « cône de vision » (trait bleu) et axe de visée (trait rouge).

### 2.4.3 Résultats

Dans cette partie, nous avons présenté une méthode explicite d'obtention des caractéristiques simplifiées d'une caméra à partir d'une matrice de transformation, dans un cas à la fois spécifique et malgré tout très général. Cette méthode est utilisée pour vérifier la cohérence des résultats de l'étape de mise en correspondance dans notre cadre applicatif (cf. Mise en œuvre dans notre cadre applicatif).

De plus, le calcul des paramètres d'observation aide l'identification entre la scène réelle et la scène reconstruite. En effet, il est important, pour les utilisateurs, de pouvoir simuler un déplacement de la caméra à partir des caractéristiques de prise de vue de l'image analysée, ce qui permet un « fondu

enchaîné » entre l'image initiale et celle produite par la visualisation du modèle, suivi d'un déplacement de la caméra selon un type de trajectoire donnée. Ce type de visualisation permet une bonne appréhension de la scène 3D en même temps qu'une mise en relation avec l'image initiale (Figure 42). Or, reproduire une visualisation identique à celle de la caméra nécessite le plus souvent (en plus de la transformation plan à plan) l'extraction d'informations relatives à la « profondeur ». Par exemple, A. Le Trotter (Le Troter, 2006) a proposé l'ajout et la sélection manuelle de quatre points dans l'image pour indiquer cette notion de trois dimensions. Notre méthode de calcul des paramètres de la caméra nous permet ainsi de proposer ce genre de visualisation uniquement à partir de la transformation 2D, ce qui représente une avancée non négligeable.

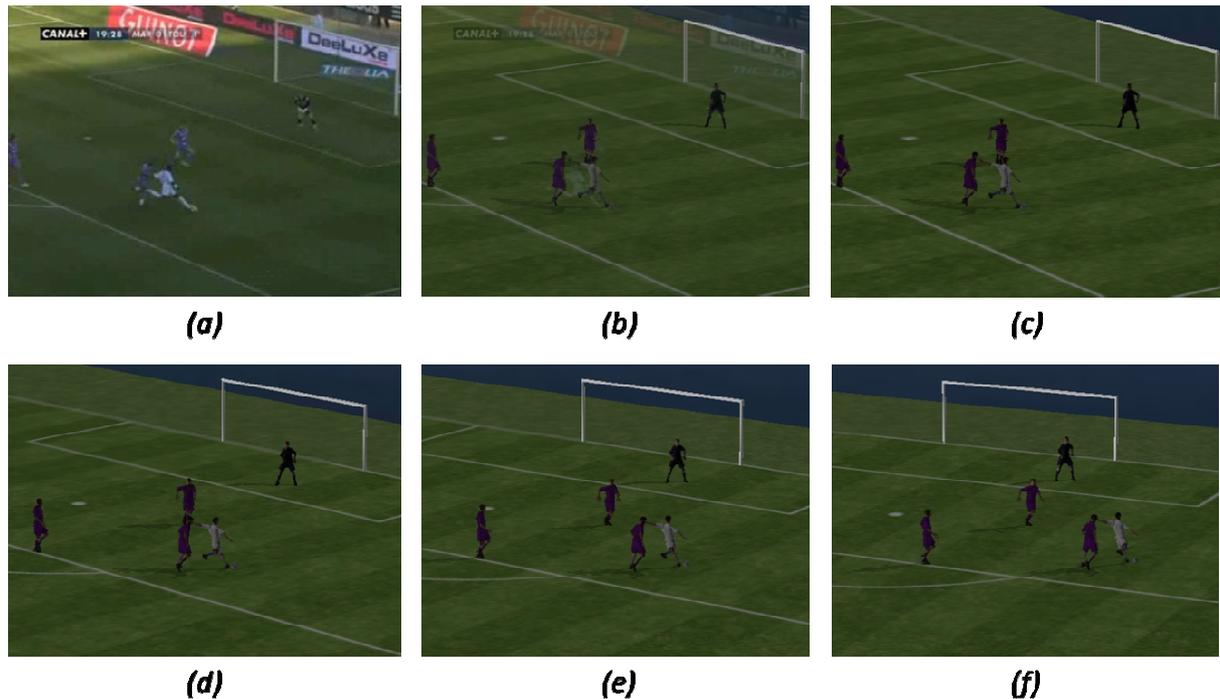


Figure 42 – Reconstruction 3D d'une scène de football

La Figure 42 montre un mouvement circulaire, à une hauteur constante, réalisé à partir de la position de la caméra : la première image (a) est celle issue de la caméra, l'image (b) est un fondu entre cette dernière et celle de synthèse (c), i.e l'image reconstruite en 3D. Les images (d) (e) et (f) représentent une rotation séquentielle de  $-10^\circ$  autour de l'axe vertical qui passe par le projeté sur le terrain du centre de l'image.

Pour calculer les paramètres de la caméra, nous avons émis un ensemble d'hypothèses. Or, ces dernières ne sont vérifiées qu'en partie ; elles répondent dans la majorité des cas à la réalité, mais sont dans une certaine mesure fausses. Toutefois, nous avons montré que les résultats obtenus par notre méthode restent satisfaisants dans notre cadre applicatif. Nous pourrions néanmoins envisager une validation théorique de nos travaux. Pour cela, nous pourrions intégrer une erreur (que l'on maîtrise) dans nos hypothèses, puis calculer de façon analogue les paramètres de la caméra pour ensuite estimer l'écart obtenu. Par exemple, si l'on intègre une légère distorsion, quelle serait alors la dérive de notre calcul ?

## 2.5 Protocole de test

Dans cette partie, nous proposons un protocole de test pour éprouver la robustesse, dans notre cadre applicatif, du processus de recalage. Après un bref rappel de ses composantes et des hypothèses associées, il apparaît que seule l'étape de mise en correspondance est délicate et nécessite toute notre attention. Aussi, nous exposons l'ensemble des résultats obtenus à partir d'un échantillon pertinent de données.

### 2.5.1 Description et hypothèses

Le processus de recalage, présenté dans les chapitres précédents, est réalisé à partir d'une image et consiste à déterminer la transformation liant celle-ci avec le modèle (un terrain de football). Ce processus se compose des deux étapes suivantes : l'extraction d'information et la mise en correspondance.

La première étape est réalisée à partir de la détection de la région d'intérêt de l'image et la segmentation des droites de cette région. Cela a fait l'objet d'une thèse et l'élaboration de méthodes dont la robustesse a alors été établie (Le Troter, 2006).

La seconde étape s'appuie sur les trois opérations suivantes :

- la classification de ces droites (en fonction de leurs points de fuite),
- leurs appariements avec celles correspondantes du modèle,
- le calibrage de la caméra (pour valider si nécessaire la transformation obtenue).

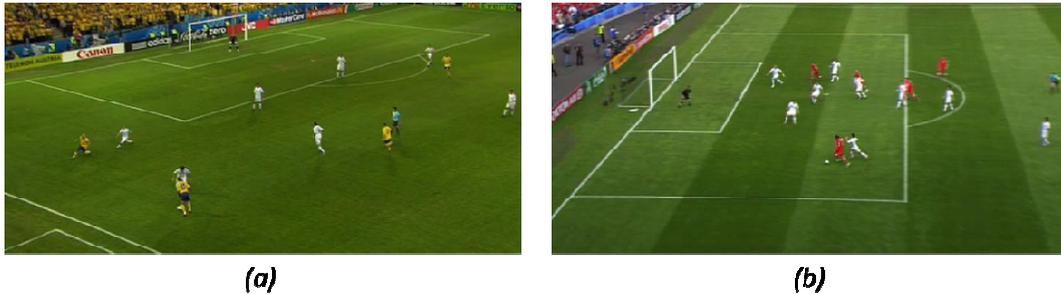
La robustesse des opérations de classification de droites en faisceaux et de calibrage de la caméra a clairement été démontrée tout au long de ce mémoire. En effet, la classification de droites est réalisée de façon géométrique et seulement quelques cas particuliers, et clairement identifiés, peuvent être problématiques. De même, l'opération de calibrage correspond à la résolution mathématique d'un système analytique (basé sur un ensemble d'hypothèses). Celle-ci fournit les paramètres exacts de la caméra lorsque ces hypothèses sont respectées. Par contre, l'appariement entre les droites de l'image et celles du modèle nécessite toute notre attention. En effet, cette opération est particulièrement délicate et requiert, pour être garantie, un ensemble d'hypothèses ainsi que l'utilisation d'un critère de cohérence que l'on détaille par la suite

Aussi, avant de présenter et de discuter les résultats obtenus par notre approche d'appariement de droites, il est important de rappeler les hypothèses utilisées par cette dernière et de souligner leurs validités dans notre cadre applicatif. Ces hypothèses sont de deux types : celles liées à la scène analysée et celles liées aux images.

Les premières s'expriment principalement sous la forme de connaissances intrinsèques à notre modèle (un terrain de football). Elles sont utilisées par notre approche de mise en correspondance et concernent aussi bien les dimensions du terrain de jeu que les tracés sur ce dernier (surface de but et surface de réparation). Ces informations ne sont pas particulièrement compliquées à obtenir ; elles sont spécifiques aux stades et accessibles à tous.

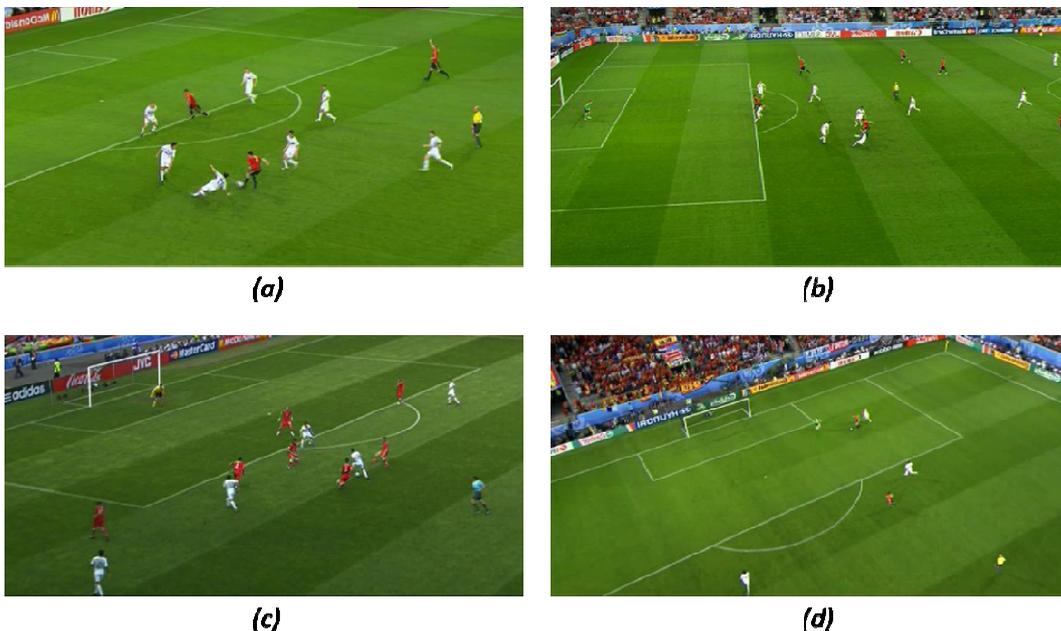
Des hypothèses liées aux images sont elles aussi utilisées par notre approche. En effet, cette dernière nécessite la détection d'un ensemble de droites dans l'image ; ce qui implique une prise de vue

suffisamment large, i.e la représentation d'une partie assez grande du terrain de jeu. Cette supposition est relativement restrictive, néanmoins, les images qui intéressent l'entraîneur pour la préparation d'un match respectent ce pré-requis (Figure 43). En effet, l'entraîneur analyse le plus souvent des images qui représentent une formation, un système de jeu ou encore une action tactique. De ce fait, celles-ci recouvrent nécessairement une part conséquente du terrain de football. De plus, une connaissance induite par la prise de vue est utilisée pour vérifier la pertinence du résultat de notre approche.



**Figure 43 – Exemple d'images utilisées par un entraîneur pour la formation. Ces dernières vérifient les hypothèses liées aux images (prise de vue, recouvrement conséquent et nombre de droites suffisant)**

Pour tester la robustesse de notre approche de mise en correspondance, nous disposons d'une banque de données de cent images (Figure 44) issues de séquences vidéo choisies par des experts, complétée par des prises de vues intéressantes (angle de caméra, nombre de lignes visibles, etc.). Nous détaillons les résultats obtenus par notre approche par la suite.



**Figure 44 – Exemple d'images de notre base de données : (a) (b) et (c) sont des images extraites de séquences sélectionnées par un expert, (a) et (b) représentent la même action de deux points de vue différents, (d) est une image choisie pour son point de vue pertinent**

## 2.5.2 Résultats

Pour définir la réussite ou l'échec de notre approche, un bref rappel de cette dernière n'est pas inutile. Celle-ci se déroule en deux étapes :

- l'étape d'initialisation : la sélection d'un quadruplet de points de l'image de telle sorte qu'il existe un quadruplet du modèle garantissant, à des conditions minimales, le calcul de la transformation liant l'image et le modèle ;
- l'étape de recherche : la détermination du quadruplet du modèle, qui correspond à celui de l'image, à partir d'un ensemble de transformations « probables » ; celles-ci étant obtenues à partir de l'appariement du quadruplet de l'image avec chacun des quadruplets possibles du modèle.

Aussi, notre approche réussit si le quadruplet de points sélectionné dans l'image correspond « réellement » au quadruplet trouvé dans le modèle.

L'étape d'initialisation est réalisée sans aucune difficulté : nous avons montré que quatre amers, formés par l'intersection de deux droites de deux faisceaux, garantissent le calcul de l'homographie liant l'image et le modèle. Par contre, l'étape de recherche est plus délicate et nous discutons de celle-ci après avoir exposé nos résultats.

Notre approche a obtenu les résultats suivants : 8% d'échec et 92% de réussite, dont 78% de réussite « directe » et 14% après l'utilisation d'une hypothèse de cohérence ; une supposition basée sur la corrélation entre les résultats obtenus et une « réalité terrain ».

L'étape de recherche consiste à déterminer le quadruplet de points du modèle qui correspond à celui de l'image. Pour cela, chaque quadruplet du modèle est mis en correspondance avec celui de l'image afin de calculer un ensemble d'homographies « probables ». Nous attribuons ensuite un « score de pertinence » à ces transformations ; score qui est la corrélation entre deux droites de l'image, transformées par cette homographie, et celles du modèle. Ainsi, l'appariement – et donc la transformation – ayant le meilleur score correspond à l'appariement recherché.

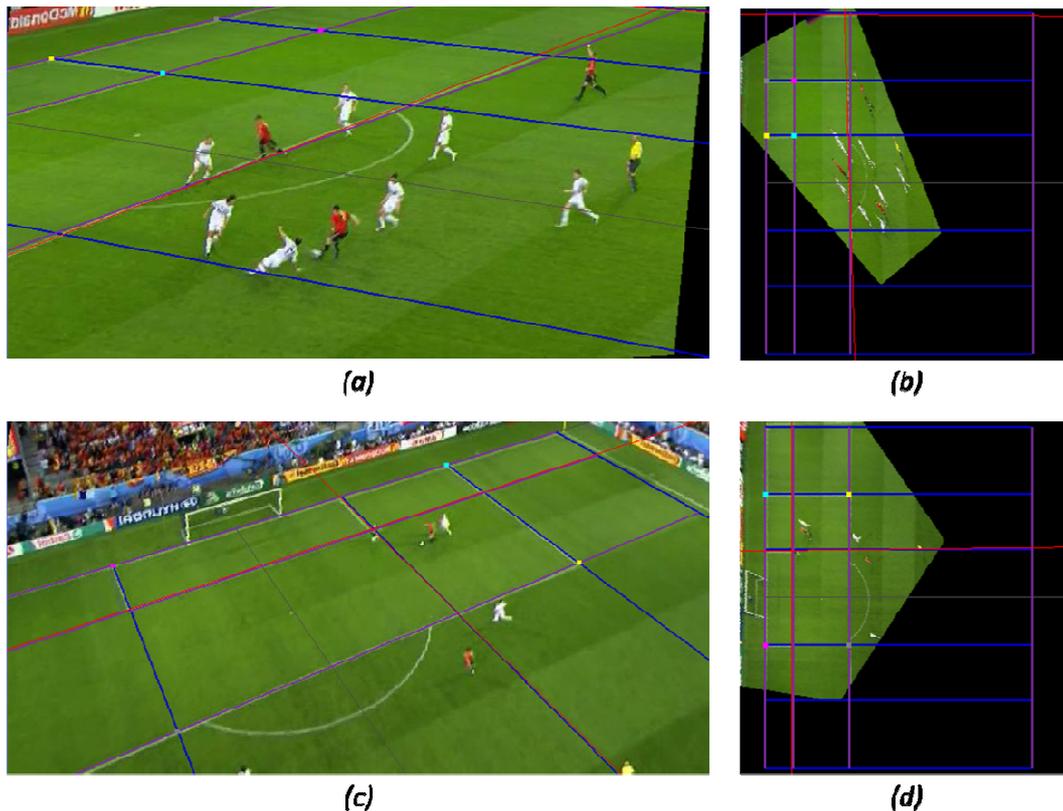


Figure 45 – Illustration des résultats de la mise en correspondance : (a) (c) sont des images sources et (b) (d) le modèle

L'ensemble des droites contenues par les images et détecté puis classé en faisceaux (Figure 45) ; celles-ci sont tracées en bleu et en violet sur les images (a) et (b). Ensuite, un quadruplet de points orienté de l'image est mis en correspondance avec tous ceux du modèle. La transformation qui obtient le meilleur score est mise en évidence par les illustrations (b) et (d). En effet, ces dernières représentent le modèle et le quadruplet de points ayant servi au calcul de cette homographie. Il est alors aisé de vérifier que le quadruplet trouvé dans le modèle correspond « réellement » à celui de l'image.

Des appariements erronés (22% de notre banque de données) sont toutefois solutions de notre approche. Cependant, l'utilisation de l'hypothèse de cohérence permet de réduire leur taux à 8%. Cette hypothèse est utilisée pour vérifier la pertinence de l'appariement solution, i.e le quadruplet de points de l'image qui correspond à celui du modèle. En effet, la transformation qui met en correspondance ces quadruplets permet le calcul des paramètres de la caméra ayant fait l'acquisition de l'image. Si ces paramètres sont aberrants, i.e qu'ils divergent d'une « réalité terrain », alors la transformation n'est pas celle qui met en correspondance l'image et le modèle.

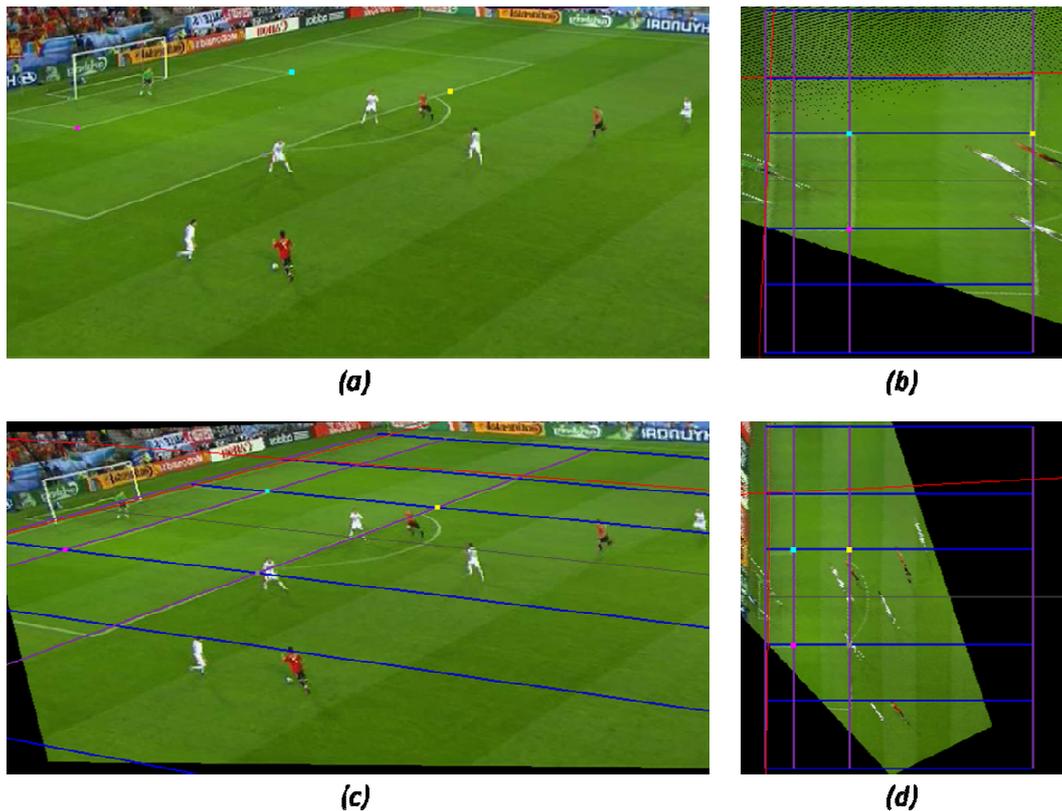


Figure 46 – Illustration du critère de cohérence et de son utilisation

Les images (a) et (b) de la Figure 46 représentent un appariement erroné ayant obtenu, pour la transformation associée, le meilleur score. Néanmoins, la position de la caméra – calculée à partir cette transformation – ne vérifie pas l’hypothèse de cohérence. En effet, celle-ci a pour coordonnées (155, -58, 51) ; position aberrante (notamment l’altitude  $z = 51$ ). Aussi, l’appariement ayant le second meilleur score – images (c) et (d) – correspond à l’appariement recherché ; sa position de caméra, (56, -74, 14), est cohérente.

Le pourcentage d’échec de notre méthode (8%) ainsi que la nécessité de l’emploi d’une hypothèse supplémentaire – la cohérence de la position de la caméra – pour résoudre certains cas ambigus (14%), s’expliquent principalement par les critères suivants :

- l’influence de la qualité de détection des droites dans l’image,
- le rapport d’échelle de notre modèle.

L’étape de recherche consiste à déterminer l’homographie liant l’image et le modèle parmi un ensemble d’homographies « probables ». Or, la qualité du calcul de ces transformations découle de celle de la détection des droites de l’image. De plus, la transformation qui est retenue, i.e l’appariement considéré valide, est fonction de la comparaison des droites de l’image – transformées par cette homographie – avec celles du modèle. Aussi, les approximations de calcul des homographies « probables » entraînent inévitablement des erreurs.

Les propriétés de notre modèle et notamment ses rapports d’échelles, sont une source d’erreurs supplémentaire. En effet, certaines lignes du terrain sont connues (surface de but, surface de réparation) alors que d’autres sont variables (les dimensions du terrain) ; cette variation se répercute

sur la comparaison de droites et entraîne parfois le choix d'un appariement erroné. De plus, l'exactitude des tracés connus n'est pas garantie et implique potentiellement des pertes de précisions. Néanmoins, l'utilisation du critère de cohérence permet un « bond qualitatif » de nos résultats, en particulier lorsque la connaissance de notre modèle n'est que partielle.

### 2.5.3 Perspectives

Nous avons établi dans cette partie la robustesse de notre approche de mise en correspondance dans notre cadre applicatif. Nous avons ainsi montré, qu'à partir d'un certain nombre d'hypothèses relativement restreintes sur l'image et sur la scène étudiée, il est possible d'automatiser entièrement le processus de recalage entre une image et son modèle. Néanmoins, il paraît intéressant de s'affranchir au maximum de ces hypothèses, notamment celle sur la prise de vue, i.e l'hypothèse de cohérence. Pour cela, un espace de Hough pourrait être utilisé pour faire émerger la transformation solution, i.e l'appariement recherché.

Notre approche nécessite au minimum six droites de l'image : quatre droites sont utilisées pour réaliser l'étape d'initialisation (déterminer un quadruplet de points dans l'image) et deux autres pour l'étape de recherche (déterminer le quadruplet de points correspondant dans le modèle). Et, l'utilisation de toutes les combinaisons de quatre droites (étape d'initialisation) et de deux droites (étape de recherche) pourrait être utilisée pour accumuler la transformation solution. En effet, l'appariement solution – la transformation ayant obtenu le meilleur score – permet de déterminer l'ensemble des amers de l'image et du modèle qui se correspondent. Ainsi, la multiplication des combinaisons pourrait être utilisée pour accumuler les couples d'amers les plus souvent mis en correspondance ; les « solutions erronées » seraient alors écartées car peu accumulées.

## 2.6 Conclusion

L'état de l'art sur le suivi d'objets à partir d'une séquence d'images (flux vidéo) a mis en évidence l'importance de l'étape de détection du mouvement. Or, cette dernière est particulièrement délicate dans le cas d'images issues de caméras mobiles non calibrées. Néanmoins, elle peut être réalisée via l'immersion de ces images dans un référentiel commun, i.e le modèle. Dans notre cadre applicatif, celui-ci correspond à un terrain de football et il est alors indispensable de recalibrer chacune de ces images avec le modèle.

Le recalage est une opération qui consiste à calculer la transformation liant une image et le modèle, il se divise en deux étapes : l'extraction d'éléments caractéristiques de l'image et leur mise en correspondance avec ceux du modèle. Cette dernière étape est particulièrement délicate, aussi nous avons proposé une méthode robuste de mise en correspondance dans un cadre général et sa mise en œuvre dans notre cadre applicatif. Toutefois, cette mise en œuvre implique la détection de faisceaux de droites, opération réalisée à partir d'une méthode originale de classification de droites en fonction de leurs points fuite. De par sa généralité, cette approche peut être appliquée dans de nombreux autres domaines tels que l'architecture (redressement d'images), imagerie médicale, etc. Aussi, un certain nombre de difficultés liées aux propriétés de notre modèle (symétries, rapports d'échelles, etc.) sont sources d'ambiguïtés lors de l'étape de mise en correspondance. Mais, ces erreurs sont naturellement écartées via le calibrage de la caméra. Pour cela, nous avons détaillé le calcul des paramètres de la caméra à partir de la transformation liant l'image et le modèle.

De plus, le protocole de test présenté ci-dessus a permis de valider l'ensemble du processus de recalage pour une image donnée qui comprend : la détection de la région d'intérêt de l'image, la détection de droites et leurs classifications en faisceaux de droites, leurs mises en correspondance avec les droites du modèle et le calcul de la transformation liant l'image et le modèle.

Enfin, ces travaux permettent d'appliquer des techniques « classiques » de suivi d'objets dans le cas d'images issues de caméras dynamiques. En effet, ces techniques nécessitent une première étape de détection du mouvement (soustraction d'arrière-plan, flot optique, etc.) ; opération qui est mal adaptée à ce type d'images, mais facilement réalisée à partir des images projetées dans le modèle, celui-ci étant « figé ».

Dans la partie suivante, nous détaillons la mise en œuvre de ces avancées dans un contexte industriel (transfert technologique), mais aussi le développement d'un prototype et les réflexions préalables à cela (veille technologique et étude d'ergonomie cognitive).

# Partie B.

## Développements et utilisation industrielle

---

### *Réalisation du logiciel SimulFoot Entraînement*

Les travaux réalisés sur le projet SimulFoot ont donné naissance à une démarche de valorisation. C'est dans ce cadre que cette bourse CIFRE a démarré. Nous présentons dans un premier temps les différents partenaires qui ont collaboré sur ce projet : l'équipe SimGraph du laboratoire LSIS et l'agence SII d'Aix-en-Provence. Dans un second temps, nous détaillons le transfert technologique réalisé qui concerne aussi bien la technologie développée en amont par l'équipe SimGraph que celle exposée dans la partie précédente de ce mémoire. Enfin, sur la base de ce transfert technologique et l'émergence du besoin d'un outil de formation orienté 3D, nous présentons le logiciel SimulFoot Entraînement (SFE) développé avec le soutien de l'organisme ValorPACA (via un plan de financement de maturation). Nous montrons donc dans un troisième temps les différentes étapes nécessaires à l'élaboration de cet outil et notamment une veille technologique, une étude d'ergonomie cognitive et le développement du logiciel SFE.

# Chapitre I. Valorisation de la recherche

## 1.1 Contexte

### 1.1.1 Historique, Laboratoire LSIS

Le laboratoire des sciences de l'information et des systèmes (LSIS – UMR CNRS 6128) est structuré sur la base de projets de recherche regroupés autour de trois pôles : informatique, systèmes et image. Le projet SimGraph s'inscrit dans le pôle image et propose d'étudier et de mettre en œuvre une méthodologie de simulation graphique dont l'objectif est de présenter de manière pertinente l'information extraite d'un important volume de données associé à des images d'origines diverses. Cette problématique est fortement émergente tant en matière de recherche qu'en ce qui concerne les applications, car elle est à la confluence de plusieurs thématiques, elles-mêmes d'actualité, comme la Multimodalité, la « Fouille de Données » (*Data Mining*) et la « Réalité Augmentée ».

Les enjeux scientifiques sont majeurs, car « un important volume de données » ne signifie pas « un important volume d'information » : il ne suffit donc pas d'utiliser des moyens plus importants pour accéder à cette information un peu plus riche et présentée de manière plus redondante. De manière analogue, la qualité de la simulation graphique ne dépend pas uniquement du réalisme des images produites, et en particulier des moyens mis en jeu, mais aussi et surtout de sa capacité à communiquer cette information. La simulation de situations données à partir de séquences vidéo ou d'images de télédétection en constitue un champ d'applications important.

Aussi, les thématiques de recherche de l'équipe SimGraph s'articulent autour de l'extraction d'information d'images ou de séquences vidéo et de la visualisation de cette information en accord avec un contexte donné (expert, métier, application, ...); ce qui donne une importance toute particulière à la composante « ergonomie cognitive ». Aussi, cette démarche implique la fusion d'informations en provenance de sources et de modalités différentes, elle nécessite donc de s'intéresser à la problématique de recalage. Ces thèmes se déclinent de la manière suivante :

- l'analyse de séquences vidéo,
- le recalage d'images de modalités différentes,
- l'ergonomie cognitive,
- les modèles de vraisemblance du contour,
- l'influence du bruit en analyse de séries temporelles.

L'analyse de séquences vidéo implique des images de même nature et de même modalité, mais dont les paramètres d'acquisition (de la caméra) ne sont pas connus et varient d'une image à la suivante. Le seul moyen d'extraire l'information de manière cohérente tout au long de la séquence vidéo consiste à s'appuyer sur la connaissance liée au problème et en particulier sur une modélisation géométrique des éléments à acquérir. Une composante importante de ce thème est la recherche de régions d'intérêt et d'éléments caractéristiques (amers) dans ces régions.

Lorsqu'il ne s'agit plus de séquences d'images, mais d'acquisitions sur une même scène par des capteurs de types différents, alors l'intégration des informations en provenance des ces différentes images requiert un recalage des données acquises. Il s'agit là de l'un des problèmes fondamentaux de la fusion de données image et l'équipe SimGraph y consacre une part son activité de recherche.

L'ergonomie cognitive concerne l'étude et l'analyse des processus cognitifs pour la gestion de la communication homme-machine, aussi bien pour ce qui est de « présenter l'information » qu'en ce qui concerne l'interaction entre l'homme et la machine, et ce, en fonction du « métier concerné ». Ces travaux sont en particulier au centre du développement d'un simulateur s'appuyant sur des données réelles (séquences vidéo) dans le cadre des sports collectifs (les travaux présentés dans ce mémoire s'inscrivent dans cette thématique). Au final, ces travaux débouchent sur la conception d'un outil parfaitement adapté à la formation des joueurs et des entraîneurs et à l'élaboration d'une méthodologie de transmission des connaissances via les technologies de l'image (cf. SimulFoot Entraînement).

Les modèles de vraisemblance de contour et les travaux associés à cette thématique constituent le prolongement d'une étude initiée en 2001 portant sur l'influence du bruit sur la détection de contour subpixel. Deux voies ont été suivies :

- l'élaboration d'un modèle de vraisemblance s'appuyant sur deux algorithmes classiques en détection de contour (le passage par zéro du Laplacien et la méthode de suppression des non maxima locaux du gradient) ;
- la définition d'un modèle de vraisemblance du contour subpixel intégré à un algorithme de contour actif. Deux modèles sont proposés : un basé sur les B-splines et l'autre échantillonné.

L'étude de l'influence du bruit en analyse de séries temporelles concerne les effets du bruit sur divers invariants utilisés pour caractériser la nature chaotique complexe ou stochastique des séries temporelles. Les premiers travaux ont permis de prédire théoriquement le comportement de l'algorithme des faux voisins moyennés (AFN : *Averaged False neighbors*) en cas de bruit pur. À partir de ces résultats, une version améliorée de l'algorithme AFN a été proposée. Ces travaux se sont ensuite poursuivis par l'étude de l'effet du bruit sur l'algorithme de l'entropie échantillonnée (*Sample Entropy*). Une utilisation originale de cet algorithme a été proposée pour l'analyse posturale en l'appliquant à des signaux du centre de pression.

De plus, un ensemble de collaborations, tant avec le milieu socioprofessionnel qu'avec le milieu institutionnel et associatif, dynamise et valorise la composante fondamentale des travaux effectués par l'équipe SimGraph. Certains sont achevés (comme, par exemple, un partenariat avec Eurocopter, qui a donné lieu à un dépôt de brevet) et d'autres sont en cours comme la réalisation de Simulfoot (un simulateur pour l'analyse de séquences de football présenté dans ce mémoire), le projet Virtù'Art (simulation de l'éclairage d'une cabine de pilotage en « Réalité Virtuelle »), le projet CISA (analyse de scènes à travers une « Caméra Intelligente pour la Surveillance Autonome »), etc.

### **1.1.2 SII, agence d'Aix-en-Provence**

SII<sup>20</sup> (Société pour l'Informatique Industrielle) est une société de services en ingénierie informatique créée par B. Huvet en 1979 et actuellement sous la direction de P. Demay, E. Matteucci et J. P. Chevée. SII compte un effectif de 2884 collaborateurs, répartis dans diverses agences. Ces dernières sont disséminées sur l'ensemble du territoire, mais également dans le reste du monde.

---

<sup>20</sup> <http://www.sii.fr/>

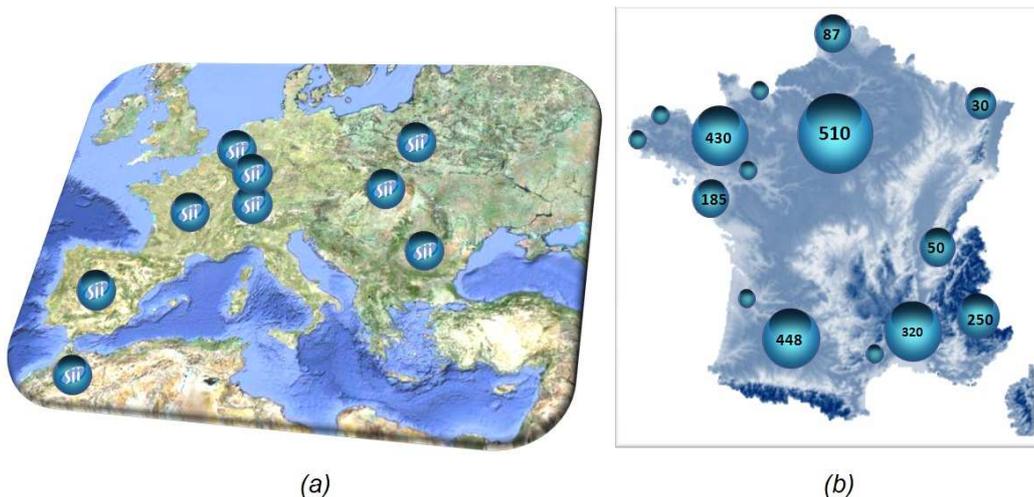


Figure 47 – Le groupe SII

La Figure 47 est une illustration de l'effectif du groupe SII ainsi que de son implantation géographique notamment en France (2335 collaborateurs), en Espagne (302), en Pologne (152) ou plus récemment au Maroc (5).

Les principaux secteurs d'activités de SII, et les clients pour lesquels SII a fourni des prestations, sont : l'industrie (les sociétés GMA CGM Systems, Thales, EADS et la DCNS), les télécoms et les médias (France Telecom, Orange et la Française des jeux), les banques et les assurances (la Société Générale, le groupe Caisse d'Épargne ou encore le Crédit Agricole) ainsi que les transports (Autoroute du Sud<sup>21</sup> et Eurocopter<sup>22</sup>). En effet, les métiers de SII sont essentiellement le traitement de l'information : systèmes d'information, systèmes embarqués et télécoms/réseaux. Mais aussi l'assistance, la maîtrise d'ouvrage et l'ingénierie : mécanique (structure, thermique, fluide, etc.), calcul scientifique, environnement (Iso 14000, DARPE, ICPE, etc.), sûreté de fonctionnement (dossier de sûreté, AMDEC, etc.) ou encore l'électronique.



Figure 48 – Secteurs d'activité et métiers des agences SII

<sup>21</sup> Une compagnie de VINCI : concessions et services

<sup>22</sup> Une compagnie de EADS

Chacune des agences a des spécificités propres, et notamment celle d'Aix-en-Provence qui s'inscrit depuis 2004 dans une démarche R&D via la création d'un pôle 3D (instigué par J. P. Chevé et Y. Hallakoun). Aussi, cette dernière a contribué à divers projets de simulation et de visualisation 3D et notamment : le SGA (*Scenario Generator Animator*), la création d'un module de « mer dynamique » et le CEV (Centre des Essais en Vol).

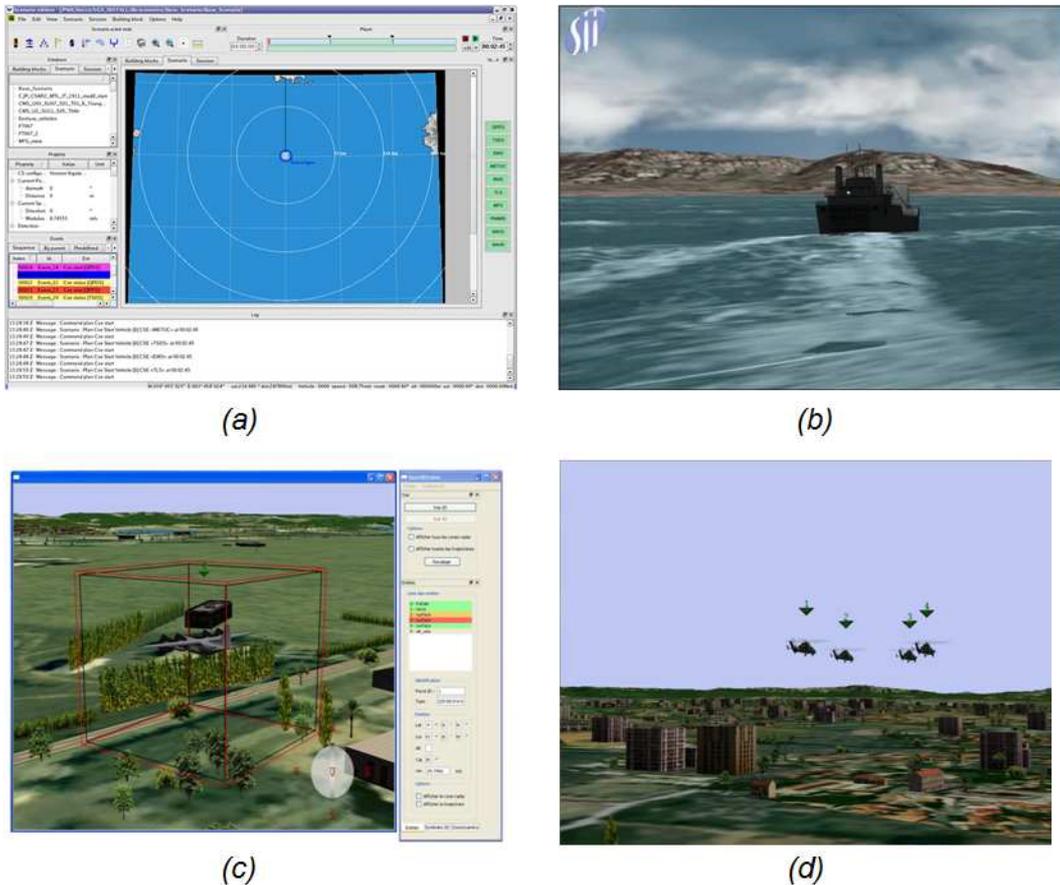


Figure 49 – SII et la simulation

La Figure 49 est une illustration du SGA, IHM (a) et mer dynamique (b), mais aussi du CEV, IHM (c) et visualisation 3D (d).

Le SGA est un système de génération et d'animation temps réel de situations tactiques permettant de définir sur une cartographie : des mobiles, auxquels sont associés des équipements de navigation et de détection, des interactions senseurs entre les mobiles, des actions planifiées associées aux mobiles ou à leurs équipements, mais aussi lancer une simulation distribuée, en émettant l'ensemble des entités du scénario à une chaîne de simulateurs (douze simulateurs actuellement).

Le CEV est un logiciel de représentation tactique qui propose la visualisation et la navigation dans un espace 2D / 3D. Cet environnement virtuel est obtenu à partir d'une base de données terrain, i.e un ensemble de données décrivant précisément le sol en deux (cartographie) ou trois dimensions (notion de relief). De plus, ce logiciel intègre divers objets statiques (bâtiments) ou dynamiques (véhicules).

L'agence d'Aix-en-Provence a également participé aux projets SICAT-BOA (DCN CMS), visuel 3D Nevis (EADS / NETCOS), portage KALAOS sous OpenSceneGraph (PIXIM), système stéréoscopique de

visualisation de fonds bathymétrique (DGA / IRENAV), visualisation du trafic sous Google Earth (CETE Méditerranée).

Cette agence vise à la réalisation de projets innovants, à fortes connotations technologiques. Cette ambition est principalement motivée par une attente client et une stratégie de communication. En effet, divers clients de SII ont exprimé le besoin d'avoir des collaborateurs et notamment des sous-traitants qui réalisent de la R&D. De plus, trouver un projet fédérateur est également une attente forte de SII en termes de communication et de recrutement. Aussi, SimulFoot est un projet qui englobe de la simulation, de la visualisation 3D et du football ; projet qui correspond parfaitement aux besoins de SII.

Aussi, le laboratoire LSIS a souhaité inscrire le projet SimulFoot dans une démarche industrielle, or, n'ayant pas vocation à développer et industrialiser du produit, le laboratoire s'est naturellement tourné vers les différents acteurs régionaux.

Le financement d'une bourse CIFRE (convention industrielle de formation par la recherche), contractée par le laboratoire LSIS (projet structurant SimGraph) et l'agence SII d'Aix-en-Provence a permis d'officialiser le partenariat entre ces deux « entités » avec comme principaux objectifs d'inscrire le projet SimulFoot dans un processus industriel – pérenniser et valoriser les travaux réalisés en amont par le laboratoire LSIS – mais aussi définir de nouveaux verrous technologiques (dans le cadre de ce projet) et y apporter des solutions. Pour cela, il a été nécessaire d'effectuer le transfert technologique du savoir-faire scientifique de l'équipe SimGraph via des compétences industrielles propres à l'agence SII.

## **1.2 Transfert technologique**

### **1.2.1 Etude de l'existant**

Le projet SimulFoot a été source de nombreux travaux (Mavromatis, et al., 2003) (Ripoll, et al., 2003) (Ripoll, et al., 2004) (Mavromatis, et al., 2007) notamment ceux de A. Le Troter (Le Troter, 2006). Ce dernier a de plus développé un outil de reconstruction 3D de scènes de football, nommée P2S. Aussi, l'étude de ces travaux et la prise en main de cet outil est une étape primordiale du transfert technologique.

Le logiciel P2S, proposé par A. Le Troter, offre la possibilité de reconstruire une scène statique ; reconstruction qui s'effectue à partir d'une image et de façon supervisée. Pour cela, P2S procède en deux temps :

- le recalage du terrain, via le calcul de l'homographie et de la matrice de transformation 3D,
- le placement des joueurs (position, orientation, posture, etc.)

Le calcul de l'homographie est réalisé à partir de l'appariement de quatre points de l'image (des amers) et de leurs correspondants dans le modèle (le terrain de football). De même, le calcul de la matrice de transformation 3D est accompli à partir de la sélection de quatre points dans l'image par l'utilisateur ; points choisis de telle sorte que deux d'entre eux n'appartiennent pas au plan terrain et que les deux autres soient leurs « projetés » sur le plan.

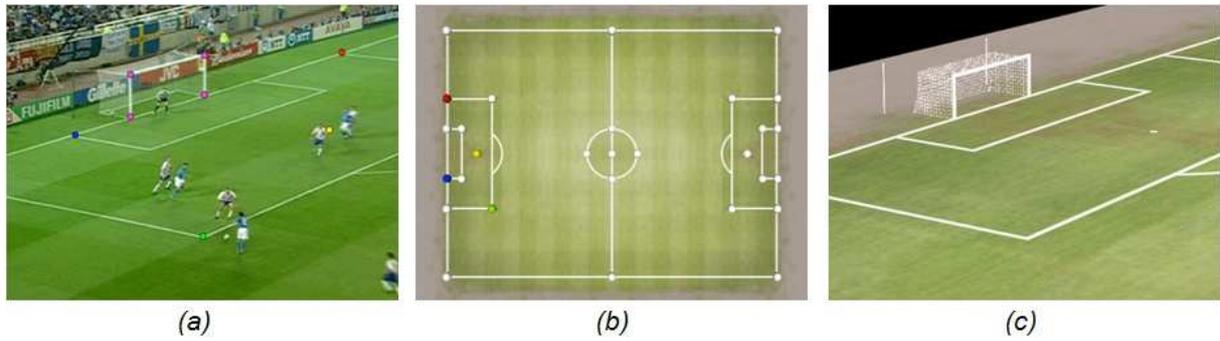


Figure 50 – Recalage du terrain

La Figure 50 représente les huit points sélectionnés dans une image source (a), les points correspondants dans le modèle (b), la vue reconstruite de la scène (c).

Les joueurs sont ensuite placés manuellement, via une interface dédiée (Figure 51) qui offre la possibilité de positionner et orienter les joueurs dans l'image ; leurs positions et leurs orientations dans la scène 3D sont automatiquement déduites à partir de la transformation précédemment calculée. De plus, il est possible d'attribuer une posture à chacun de ces joueurs. Pour faciliter cette reconstruction, l'opérateur est guidé par différents algorithmes de traitement d'images et notamment par :

- la détection de la ROI (*Region Of Interest*), la région de l'image qui correspond au terrain de football ; celle-ci est extraite par une segmentation couleur de l'image à partir de sa représentation dans l'espace couleur HLS<sup>23</sup>,
- la détection des primitives (droites et ellipses) de l'image ; étape réalisée par une approche originale de type transformée de *Hough*.

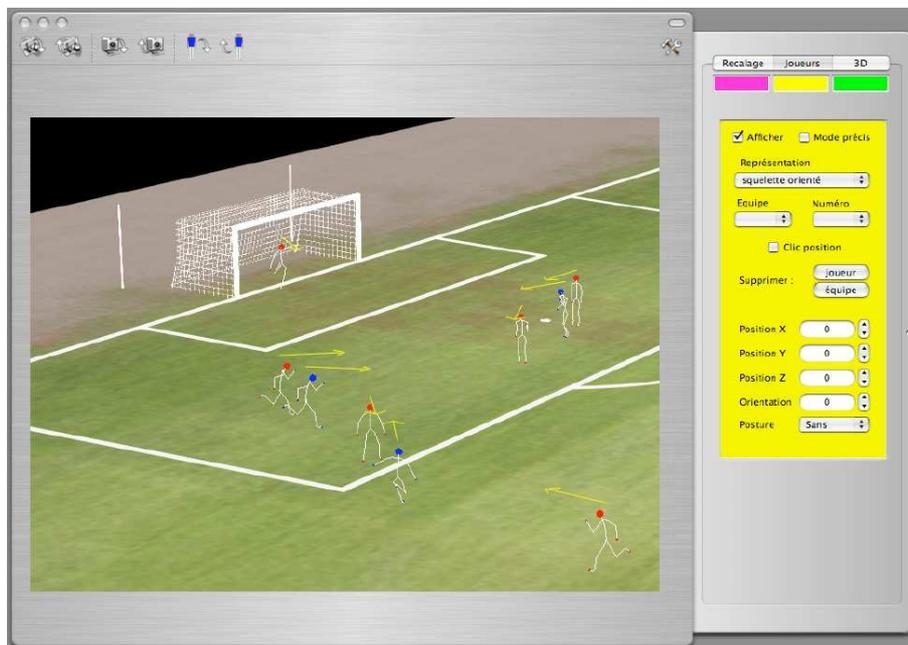


Figure 51 – Placement des joueurs

<sup>23</sup> Format de codage des couleurs : teinte (*Hue*), saturation (*Saturation*), lumière (*Lightness*).

La Figure 51 est une illustration de l'Interface Homme-Machine utilisée pour positionner et orienter les joueurs.

Aussi, le transfert technologique du savoir-faire de l'équipe SimGraph, et de leurs précédents travaux concernant SimulFoot, est pleinement réalisé par l'élaboration d'un prototype qui inclut les algorithmes présentés ci-dessus. De plus, celui-ci est employé comme plateforme de test pour intégrer les travaux de recherche présentés dans la partie A.

Dans la suite de ce mémoire, nous détaillons les deux principales étapes qui sont indissociables à l'élaboration de ce prototype : le choix d'un ensemble d'outils adaptés à sa réalisation et le « développement logiciel ».

## **1.2.2 Réalisation d'un prototype**

Le développement du prototype qui inclut les travaux menés en amont par l'équipe SimGraph nécessite dans un premier temps la sélection d'un ensemble d'outils qui sont :

- l'environnement et le langage de programmation,
- la conception UML,
- l'Interface Homme-Machine (IHM),
- la visualisation et le traitement d'images.

### **1.2.2.1 Choix des outils**

Les outils utilisés pour le transfert technologique sont fonction de divers critères tels que leurs pérennités ou encore leurs simplicités de déploiement (documentation précise, communauté active, etc.). Aussi, être multiplateforme (Windows, MAC OS, Linux) et libres de droit sont deux critères pris en considération.

#### **Environnement et langage de programmation**

Ce prototype est programmé en C++ ; choix principalement lié à la pertinence de la représentation objets et de l'utilisation de « patrons de conception » (singleton, modèle-vue, etc.). De plus, l'environnement de travail choisi correspond à Visual Studio<sup>24</sup> pour sa facilité d'intégration avec les outils retenus pour la conception, l'Interface Homme-Machine et la visualisation.

#### **Conception UML**

La conception UML a été réalisée sous *Enterprise Architect*<sup>25</sup> (logiciel qui s'intègre naturellement avec Visual Studio). Ce logiciel propose une interface dédiée à la création des différentes classes et de leurs interactions pour ensuite générer le code associé. De plus, *Enterprise Architect* permet l'emploi de techniques de rétro-ingénierie.

---

<sup>24</sup> <http://msdn.microsoft.com/>

<sup>25</sup> <http://www.sparxsystems.com.au/>

## Interface Homme-Machine

L'Interface Homme-Machine a été réalisée sous Qt<sup>26</sup> (Trolltech, Nokia©) ; probablement le logiciel d'interface le plus abouti. En effet, Qt est un logiciel multiplateforme fortement orienté objet qui propose de nombreux outils tels que le *designer* (un outil graphique pour créer des interfaces et qui s'intègre facilement dans Visual Studio), mais aussi les *signals & slots* (principe très intuitif qui permet le « dialogue » entre les différentes entités d'un même programme).

## Visualisation et traitement d'images

La visualisation est obtenue par le logiciel Ogre3D<sup>27</sup>, un moteur de rendu de type « graphe de scènes » qui a pour avantage d'être *open source*, multiplateforme, multi API 3D (OpenGL et DirectX) et qui s'intègre relativement facilement dans Visual Studio et Qt. De plus, la communauté Ogre3D est parmi les plus actives (sites internet, wiki, forums, etc.). Enfin, la lecture de vidéos et le traitement d'images sont réalisés via OpenCV ; une bibliothèque graphique libre spécialisée dans le traitement d'images temps réel.

### 1.2.2.2 Développement

Dans un second temps, le développement du prototype de faisabilité nécessite la conception des classes et de leurs diverses interactions ainsi que la réalisation d'une bibliothèque d'analyse d'images. Cette dernière permet de résoudre les différents problèmes liés à la reconstruction, notamment les problèmes de recalage et de calibrage de la caméra. De plus, ce prototype requiert la création d'une Interface Homme-Machine ; principalement composée d'une visualisation 3D de la reconstruction d'une scène de football et d'un assistant qui permet de guider pas à pas cette reconstruction.

### IHM principale

L'Interface Homme-Machine (Figure 52) du prototype est composée d'une barre de lancement rapide (affiche un assistant qui guide pas à pas l'utilisateur dans sa reconstruction 3D), d'une barre d'outils (qui permet d'effectuer un certain nombre d'actions : sélection, translation, rotation, ajout d'éléments, etc.) et de diverses « boîtes de dialogues » pour insérer des éléments dans la 3D et les modifier (zones circulaires, flèches droites et courbes, lignes de hors-jeux, etc.).

---

<sup>26</sup> <http://qt.nokia.com/>

<sup>27</sup> <http://www.ogre3d.org/>



Figure 52 – Interface Homme-Machine du prototype de validation

### L'assistant

La reconstruction 3D d'une scène de football est guidée par un assistant (Figure 53) et celui-ci se décompose en trois étapes : la sélection du média, puis le calibrage et enfin l'insertion d'entités.

L'étape de sélection du média consiste à choisir, à partir d'un flux vidéo, l'image que l'on souhaite reconstruire. Pour cela, le prototype de validation propose un lecteur multimédia classique, i.e la sélection, la visualisation et la navigation dans une vidéo.

L'étape de calibrage de caméra correspond au calcul de la transformation liant l'image et le modèle (le terrain de football), mais aussi au calcul des paramètres de la caméra ayant acquis cette image. Pour ce faire, l'utilisateur doit choisir quatre amers dans l'image – un amer correspond à l'intersection de deux droites (tracés du terrain) – puis indiquer leurs correspondants dans le modèle.

L'étape d'insertion des entités consiste à placer les joueurs dans la scène reconstruite. Pour cela, l'utilisateur visualise l'image source et peut positionner et orienter les joueurs dans cette image. Ainsi, la position et l'orientation des joueurs dans la 3D sont automatiquement calculées à partir de la transformation déterminée dans la précédente étape.

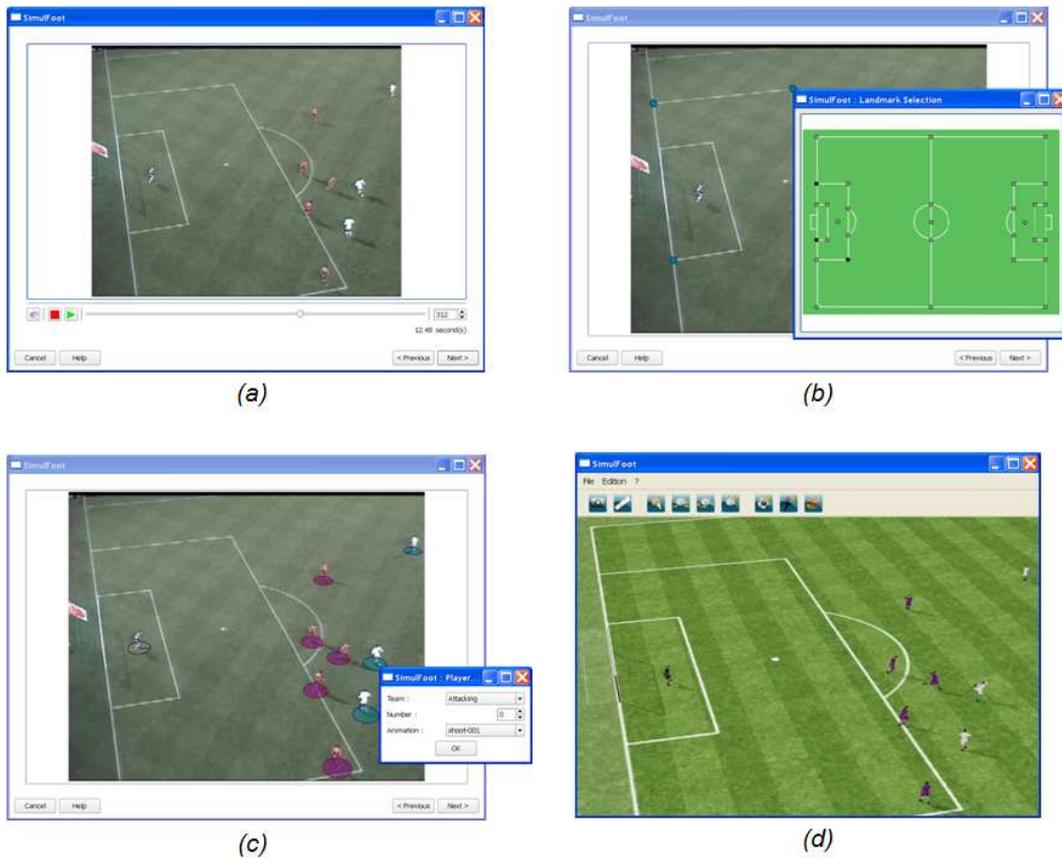


Figure 53 – Illustration de l'assistant

La Figure 53 est une représentation des étapes de l'assistant : la sélection du média (a), le calibrage (b), l'insertion des entités (c) et enfin la visualisation de la scène reconstruite (d).

### Reconstruction 3D et visualisation

Une fois la reconstruction effectuée, il est possible de « naviguer » dans la scène reconstruite, de l'annoter, de faire des captures d'écran ou encore de créer des mouvements de caméras pour générer des séquences vidéo animées.



Figure 54 – Résultats

La Figure 54 représente la reconstruction 3D d'une scène de football et les avantages liés à celle-ci, notamment : changer de point de vue (a) ou encore se « mettre » à la place d'un joueur (b).

Le transfert de technologies a été réalisé avec succès et a été concrétisé par le développement d'un prototype de faisabilité. Ce dernier a également été employé comme « plateforme d'accueil » aux avancées technologiques détaillées dans la partie A. de ce mémoire, mais il a aussi servi de support lors du 3<sup>ième</sup> Colloque International « FOOTBALL & RECHERCHE » organisé à Valenciennes (Ripoll, et al., 2009). Cette communication nous a conforté dans nos choix et a permis de recueillir les attentes des praticiens du sport. Aussi, l'intérêt manifesté par ces derniers a légitimé l'industrialisation de ce prototype ; opération rendue possible par le financement octroyé par l'organisme valorPACA.

Dans la suite de ce mémoire, nous exposons les principales étapes nécessaires à la réalisation du logiciel SimulFoot Entraînement (SFE) : une veille technologique, une étude d'ergonomie cognitive relative aux images et le développement informatique de SFE.

## Chapitre II. Maturation

### 2.1 Veille technologique

La réalisation d'une veille technologique des différents outils existants dans le domaine de l'analyse sportive (le football et plus généralement les sports collectifs) permet de définir les marchés potentiels ; marchés qui sont principalement la didactique et le commentaire d'actions sportives (club, TV), la TV 3D (sans droits), l'archivage et la transmission de séquences en vectoriel (mobiles), mais aussi la publicité virtuelle. Cette veille technologique fait émerger deux types de technologies qui sont actuellement déployées : les logiciels orientés formation (pauvres en informations rajoutées et en 3D mais très accessibles) et les logiciels orientés « média » (pouvant être riches en 3D mais très complexes dans leur utilisation).

#### 2.1.1 Logiciels orientés formation

De nombreux outils de formation sont disponibles sur le marché, notamment les logiciels Winner, TechFoot et TactFoot, Soccer-Trainer et X-Simul ; logiciels « agréés » par la FFF<sup>28</sup> (Fédération Française de Football).

Winner<sup>29</sup> est un logiciel qui permet de gérer une équipe de football à travers le suivi et la construction de séances d'entraînement, la gestion des rencontres et la planification de la compétition. Ce logiciel fournit un fichier de plus de 350 exercices et un outil pour créer ses exercices personnalisés. De plus, un module permet de gérer le championnat et d'avoir ainsi des classements selon différents critères.

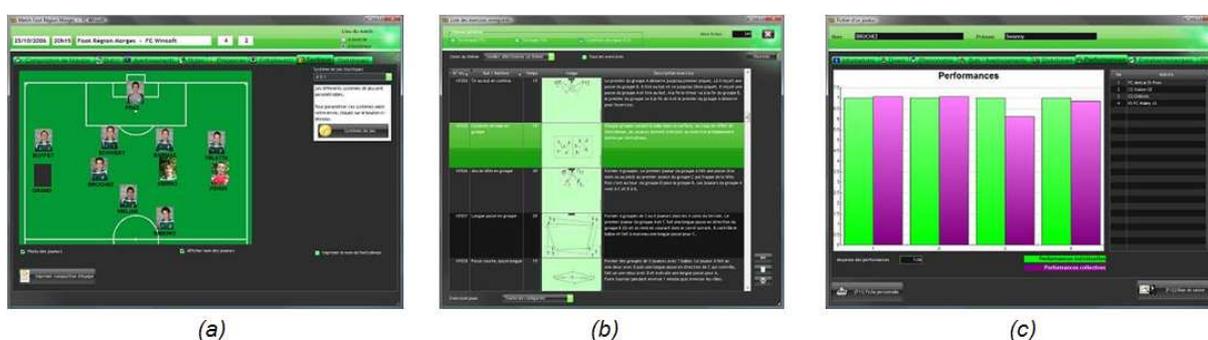


Figure 55 – Illustration du logiciel Winner

TechFoot et TactFoot sont des logiciels déclinés par FootSuitTrainer<sup>30</sup> pour aider et optimiser le travail de l'entraîneur. TechFoot est un logiciel de suivi de l'effectif d'une équipe sur une ou plusieurs saisons (suivi des joueurs, des équipes, des rencontres, des entraînements, des tests physiques et techniques). En effet, cet outil permet de planifier un programme d'entraînement, de quantifier les aptitudes physiques et tactiques de l'équipe et ainsi de gagner du temps dans l'organisation des séances d'entraînement, de créer des exercices personnalisés et d'interroger, en

<sup>28</sup> <http://www.fff.fr/>

<sup>29</sup> <http://www.winnerfoot.com/fr/>

<sup>30</sup> <http://www.footsuittrainer.com/>

fonction de différents critères, les informations stockées. TactFoot correspond au tableau noir informatique ; logiciel d’animation qui permet de recréer des situations tactiques à travers des dispositifs différents, d’effectuer un travail pédagogique notamment au niveau du positionnement (création de séances d’entraînement animées).

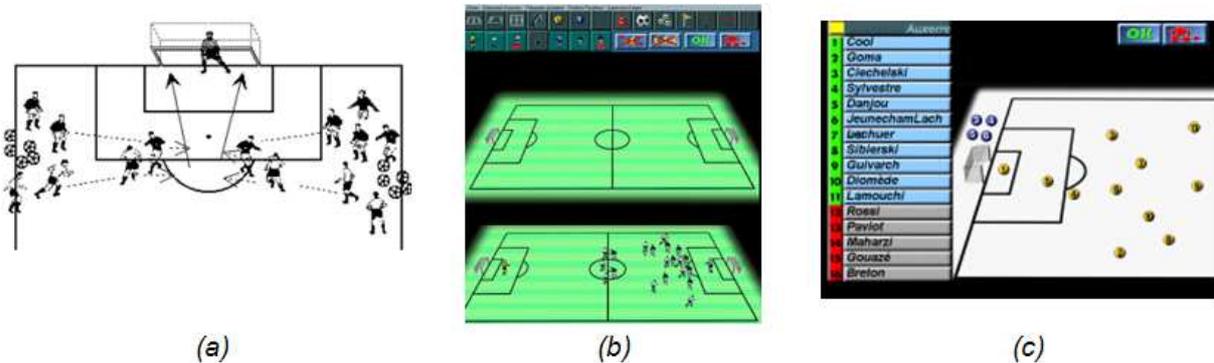


Figure 56 – Illustration des outils TechFoot & TactFoot

Soccer-Trainer<sup>31</sup> est un logiciel de création de séances d’entraînement ; 350 exercices classés par objectifs et catégories d’âge sont à disposition pour permettre à l’entraîneur de créer des séances et des programmes performants et adaptés à son équipe via des exercices et des jeux animés (un éditeur d’exercices permet la création et l’intégration d’exercices personnalisés).

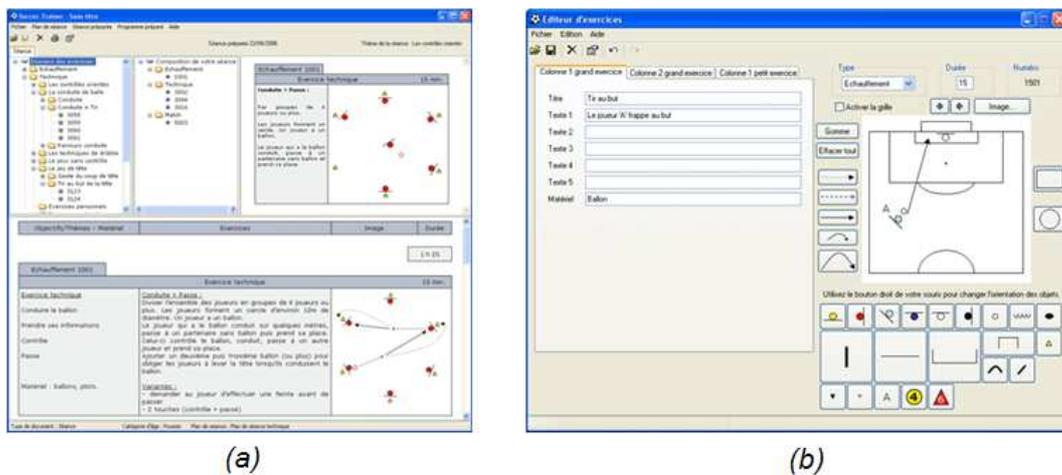


Figure 57 – Illustration de l'IHM du logiciel Soccer-Trainer

X-Simul est un logiciel de préparation et de simulation multisport (football, basket-ball, handball, rugby, volley-ball, etc.) ayant pour objectif la création d’animations pour simuler les tactiques de placements sur le terrain ; ces animations peuvent alors être projetées ou exportées sur Internet (en Flash).

<sup>31</sup> <http://www.soccer-trainer.fr/>

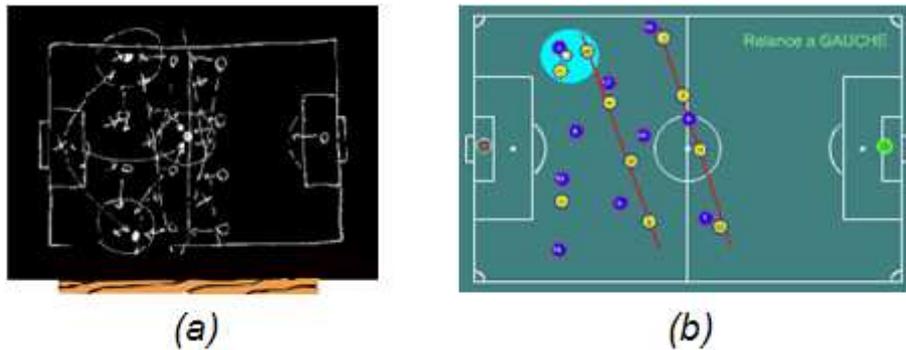


Figure 58 – Illustration du logiciel X-Simul

### 2.1.2 Logiciels orientés « média »

Il existe deux types de logiciels d'analyse de vidéos sportives :

- ceux qui fournissent des statistiques, pendant et après le déroulement d'un match (distances parcourues par les joueurs, nombre de ballons joués, etc.) tel que Amisco Media (société Sport Universal), Scanfoot (société Scanball), CyberSport (société ORAD) ou encore les outils développés par la société SportTec<sup>32</sup> ;
- ceux qui proposent une reconstruction des actions marquantes d'un match.

Ces derniers nous intéressent plus particulièrement et on peut citer les solutions les plus abouties qui sont : Virtual Replay (société SportFlashBack<sup>33</sup>), 3DReplay (société ORAD<sup>34</sup>), PIERO<sup>35</sup> (société RED BEE) ou encore le projet R&D TRICTRAC<sup>36</sup>.

Les trois premiers logiciels proposent des solutions intéressantes pour effectuer la reconstruction 3D de scènes sportives, aussi bien statiques que dynamiques, mais ces derniers sont fortement supervisés alors que le projet R&D TRICTRAC, bien que non commercialisé, propose des solutions pour tendre vers l'automatisation de la reconstruction.

Le logiciel Virtual Replay permet la reconstruction de scènes dynamiques et se décline sous différentes formes pour répondre à des marchés plus spécifiques tels qu'internet (avec *Tactics Animator*) ou encore la téléphonie. Toutefois, peu automatisé, il nécessite un recours non négligeable de l'utilisateur.

<sup>32</sup> <http://www.sportstec.com/>

<sup>33</sup> <http://sportflashback.com/>

<sup>34</sup> <http://www.orad.tv/>

<sup>35</sup> <http://www.redbeemedia.com/piero/>

<sup>36</sup> <http://www.multitel.be/trictrac/>



Figure 59 – Illustration du logiciel Virtual Replay

La Figure 59 est une illustration de la gamme de produits proposés par la société SportFlashBack : le logiciel de reconstruction 3D dynamique Virtual Replay (a), son application internet orientée tactique (b) et son application mobile (c).

3DReplay est un logiciel qui permet quant à lui la reconstruction supervisée de scènes statiques. Il nécessite néanmoins une mise en œuvre coûteuse qui comprend entre autres l'achat de plusieurs caméras, d'un serveur vidéo et d'une station graphique.

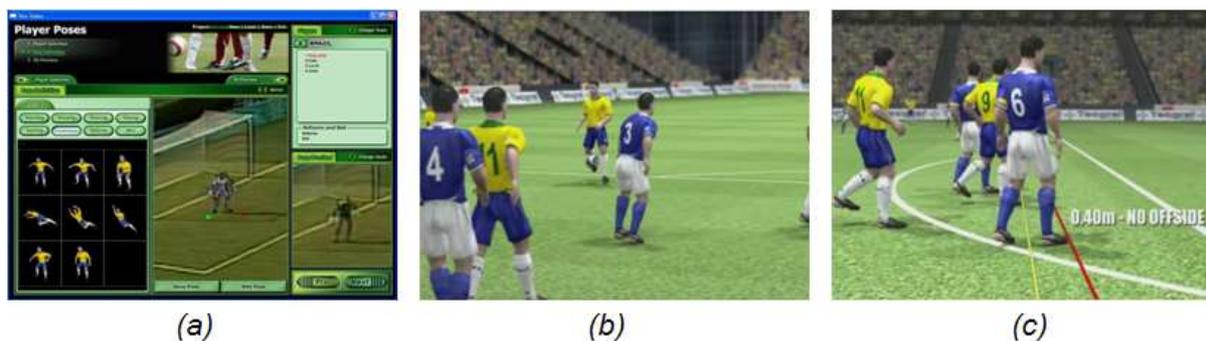


Figure 60 – Illustration du logiciel 3DReplay

La Figure 60 représente l'Interface Homme-Machine du logiciel 3DReplay (a) et différents résultats de reconstruction 3D (b) (c).

PIERO est probablement le logiciel le plus abouti actuellement disponible sur le marché. En effet, ce dernier permet la reconstruction statique et dynamique de scènes de football, notamment pour des applications télévisuelles. Ce logiciel est d'ailleurs utilisé par des chaînes telles que la BBC et Canal+.

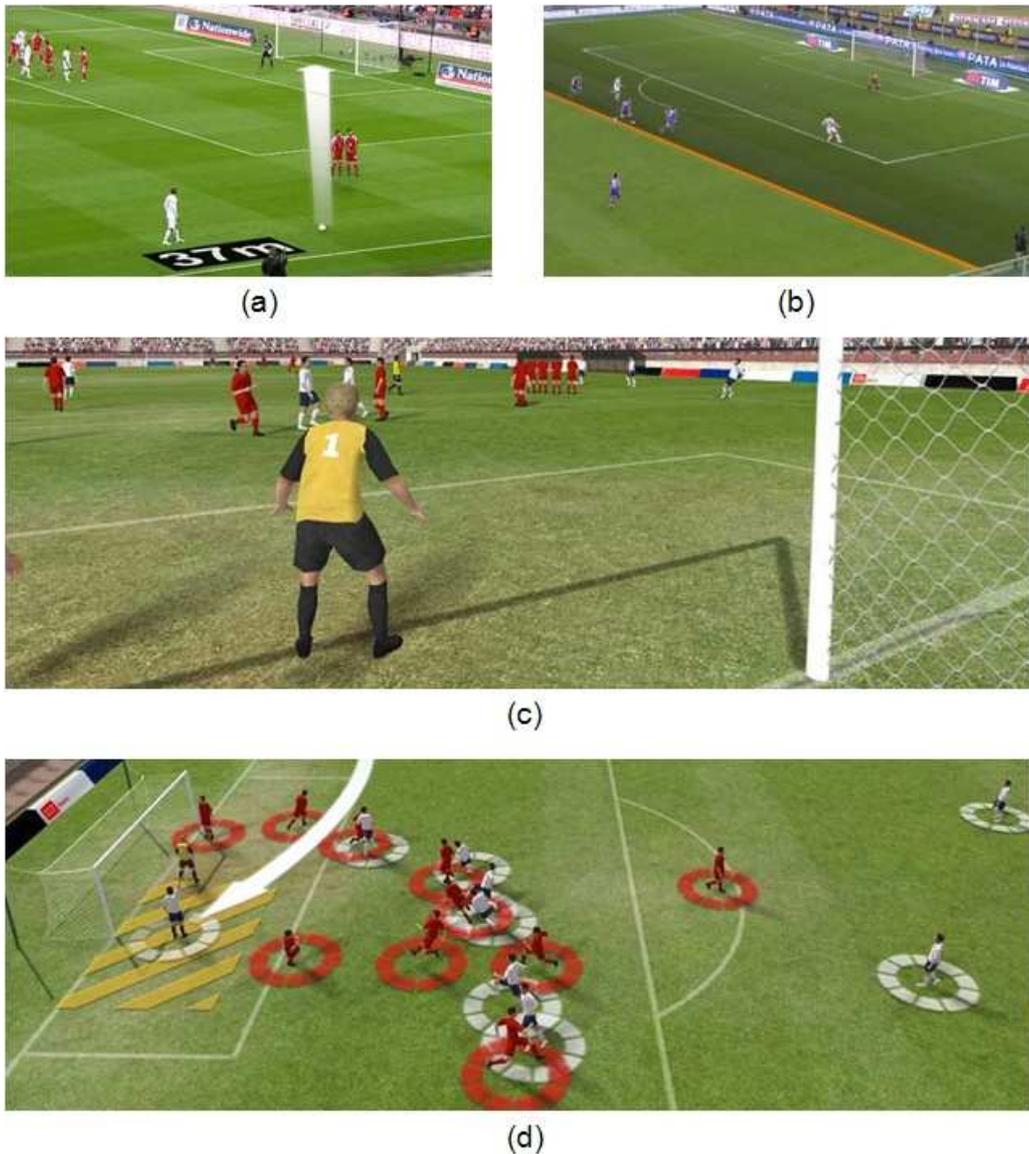


Figure 61 – Illustration du logiciel PIERO

La Figure 61 est une illustration des différentes possibilités offertes par le logiciel PIERO, notamment les « effets sur le terrain » : distance d'un coup franc (a), ligne de hors jeu (b), et les « effets » entièrement 3D : la vue d'un joueur (c), vue tactique (d).

TRICTRAC est un projet de R&D particulièrement proche de notre cadre applicatif. En effet, il a pour objectif la reconstruction automatique de scènes dynamiques de football qui nécessite l'application de méthodes de suivi de joueurs en temps réel à partir d'images issues d'une ou de plusieurs caméras. TRICTRAC<sup>37</sup> est un projet mené en collaboration avec le groupe INTELSIG *Montefiore Institute* de l'Université de Liège, le groupe TELE de l'Université Catholique de Louvain et le groupe de recherche Multitel.

<sup>37</sup> Le projet TRICTRAC a été fondé par WIST research et financé par la DGTE, the Walloon Region Direction of Technology, Research and Energy



Figure 62 – Reconstruction et visualisation 3D (sous Ogre3D) d'une séquence de jeu

Nous avons présenté les deux types de logiciels d'analyse sportive actuellement disponibles sur le marché : ceux orientés formation, pauvres en informations rajoutées et en 3D, et ceux orientés média, riches en 3D mais très complexes d'utilisation. Ainsi, l'analyse de ces divers outils permet l'émergence d'un marché lié à un besoin nouveau : celui de la réalité virtuelle – 3D – orientée pour la formation. Or, aucune validation de ces outils n'a été proposée en ce qui concerne leur réel apport sur l'aide à l'apprentissage (pertinence des images, des points de vues, etc.). Il est donc primordial de s'intéresser à cette question pour proposer un outil adapté à la demande.

## 2.2 Etude ergonomique

### 2.2.1 Introduction

De nombreuses recherches sur l'apprentissage des sports collectifs montrent que les connaissances d'un expert sont beaucoup plus précises, plus riches et mieux organisées que celles d'un novice (Zoudji, et al., 2003) (Zoudji, et al., 2005) (Laurent, et al., 2006) (Petit, et al., 2008) (Ripoll, 2008) (Laurent, et al., 2009). Celles-ci sont de deux sortes : procédurales – en lien direct avec l'action – et déclaratives ; reposant sur des systèmes de représentation de l'action. De plus, la pertinence et la vitesse des décisions prises par un joueur sur le terrain sont fonction de la richesse de ce répertoire de connaissances. Or, les études de V. Ferrari *et al.* (Ferrari, et al., 2009) confirment que les joueurs autodidactes, qui ne bénéficient pas d'un encadrement pédagogique, n'accèdent qu'en partie à la logique du jeu de football. Ces derniers acquièrent des savoir-faire techniques, i.e des connaissances de type procédurale (analyse de phases de jeu, utilisation de l'espace, conduite de balle, tir au but, etc.), mais ne développent que partiellement les formes plus sophistiquées de jeu, i.e les connaissances de type déclarative (combinaisons tactiques, schémas tactiques, systèmes de jeu, etc.). Il est donc primordial pour l'entraîneur (ou l'éducateur) d'améliorer et d'enrichir la pensée tactique de ses joueurs. Ces intervenants utilisent souvent des images : de façon abstraite à partir d'un tableau et en manipulant des symboles (pions, flèches, etc.) ou de façon réaliste via des vidéos. Cette utilisation est alors accompagnée de consignes (commentaires, descriptifs) pour expliciter les positions, les déplacements et replacements relatifs des joueurs (tableau) et illustrer la façon dont le jeu à été mis en œuvre par les joueurs (vidéos). Néanmoins, il est primordial de comprendre les mécanismes cognitifs impliqués dans le cadre de l'apprentissage afin d'utiliser le plus efficacement possible ces images.

### 2.2.2 De la représentation imagée à l'action

Les images donnent des informations abstraites aux joueurs concernant l'action à effectuer et lui permettent ainsi de se représenter l'action en situation réelle. Aussi, le passage entre l'image perçue

et la situation de jeu (réelle) est un processus complexe qui implique plusieurs opérations mentales : le joueur doit se projeter d'un espace 2D vers un espace 3D, d'une représentation de scène statique à une représentation dynamique et d'un espace vu de l'extérieur à un espace vécu de l'intérieur. Aussi, les images simulent davantage la réalité à un expert qu'à un non-expert (Poplu, et al., 2003) (Poplu, et al., 2008) car elles font davantage écho à un vécu sur le terrain. Cette sensibilité perceptive permet au joueur de développer (via ces images) un ensemble de connaissances utilisable sur le terrain (prise de décisions adaptées à un problème donné tel que passer la balle ou la garder) (Ripoll, 1988). De plus, les images proposées à un joueur sont statiques et donc en contradiction avec une scène (réelle) dynamique. Cependant, les études de A. Didierjean et E. Marmèche (Didierjean, et al., 2005) établissent que le joueur expert planifie mentalement et sans effort de réflexion la suite logique d'une situation. L'ensemble de ces études montre donc que :

- la pratique du football en autodidacte (sans encadrement) entraîne un répertoire d'image appauvri ;
- les représentations imagées, de type tableau et vidéo, accompagnées d'explications et de consignes enrichissent ce répertoire ;
- la qualité de jeu est en relation avec la richesse de ce répertoire.

Ainsi, l'entraîneur doit se poser la question suivante : comment agrandir le répertoire d'un joueur non expert (ou même expert) et à quelles conditions ?

### **2.2.3 L'apprentissage par l'image**

Un joueur expert active des schémas de façon automatique pour réagir à une situation de jeu durant le match. La vitesse et la pertinence de cette prise de décision dépendent de la vitesse de récupération des connaissances, déclaratives et procédurales, stockées en mémoire à long terme. Cela n'est efficace que si ces dernières ont été mises en correspondance au cours de l'entraînement, facilitant ainsi l'encodage en mémoire à long terme. En effet, H. Ripoll (Ripoll, 1996) (Ripoll, 2004) postule que la mise en relation d'images (tableau et/ou vidéos) avec les connaissances pratiques (sur le terrain) permet un *multiréférencage* qui favorise alors la récupération des connaissances et leur activation visant à une prise de décision très pertinente et souvent optimale.

Afin que la formation par des images soit efficace, l'entraîneur doit prendre en considération trois critères : l'âge des joueurs, leur niveau d'expertise et la clarté des consignes. En effet, la compréhension des images et des symboles n'est accessible que tardivement. Les études sur le développement cognitif (Piaget, 1946) montrent qu'un enfant de moins de onze ans, n'ayant pas encore accès à la pensée symbolique, ne peut comprendre l'usage de symboles et d'images. Des études appliquées aux sports vont également dans ce sens (Ripoll, 1997) (Menaut, 1998). En ce qui concerne le niveau d'expertise, nous avons exposé précédemment que les joueurs experts présentent une meilleure compréhension des images que des joueurs débutants ; les études de H. Ripoll (Ripoll, 1979) confirment également cela. Il est donc fondamental que l'entraîneur permette à ses joueurs de comprendre l'image et les codes qui y sont associés. La clarté des consignes est également essentielle et elle consiste à mettre en relation la représentation symbolique (ou imagée) avec la mise en œuvre sur le terrain via, notamment, des procédures d'autocontrôle et de rétrocontrôle (l'évaluation d'exercices par exemple).

## 2.2.4 Conclusion

Dans la partie précédente, nous avons clairement établi la pertinence de l'utilisation d'images en vue de l'apprentissage des sports collectifs. Ce constat a conduit un certain nombre de chercheurs à développer l'outil SimulFoot destiné à l'entraînement (SFE). Ce dernier est un simulateur de football qui permet à l'entraîneur de créer, à l'aide d'une interface graphique d'usage simple, des situations de jeu de façon analogue au tableau magnétique (Figure 63) et d'animer ces situations comme en vidéo (Figure 64). Parmi les nouveaux modes de représentation mis à disposition de l'entraîneur à travers l'usage de SFE, cet outil propose la possibilité :

- de varier les points de vue pour une séquence de jeu donnée et notamment, se placer en vue subjective ;
- de représenter les joueurs de manière symbolique (croix, cylindre, etc.) ou réaliste.

Le principal apport de SFE est donc de simuler la mise en relation entre une situation abstraite et une situation réelle et ainsi de faciliter la compréhension des consignes données par l'entraîneur et leur assimilation en mémoire à long terme. Le répertoire de connaissances des joueurs s'en trouve alors enrichi et consolidé.

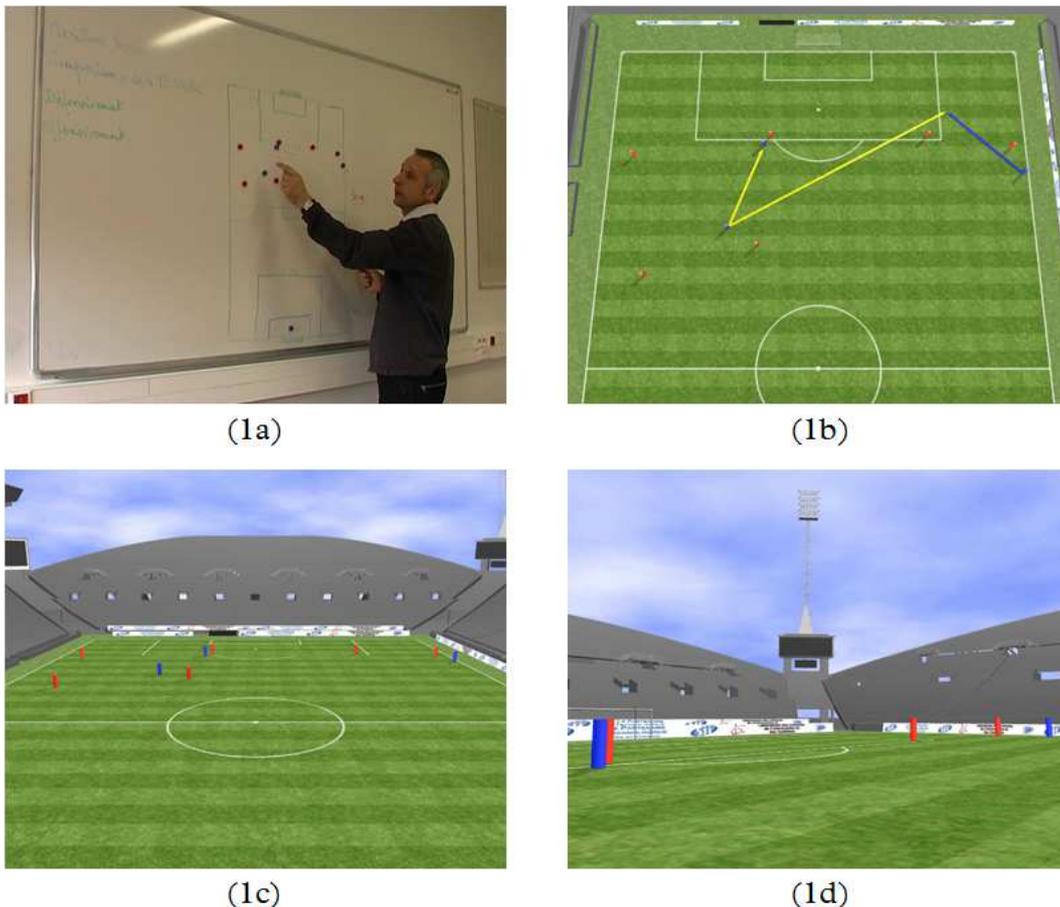


Figure 63 – Situation imaginée

La Figure 63 est une séquence de jeu imaginée par l'entraîneur (1a) reproduite virtuellement ; les joueurs et les actions de jeu sont schématisés, respectivement par des cylindres et des flèches

colorés, et sont positionnés comme sur le tableau (1b). La scène peut alors être visionnée soit de l'extérieur (1c) soit de l'intérieur, i.e du point de vue des joueurs (1d).

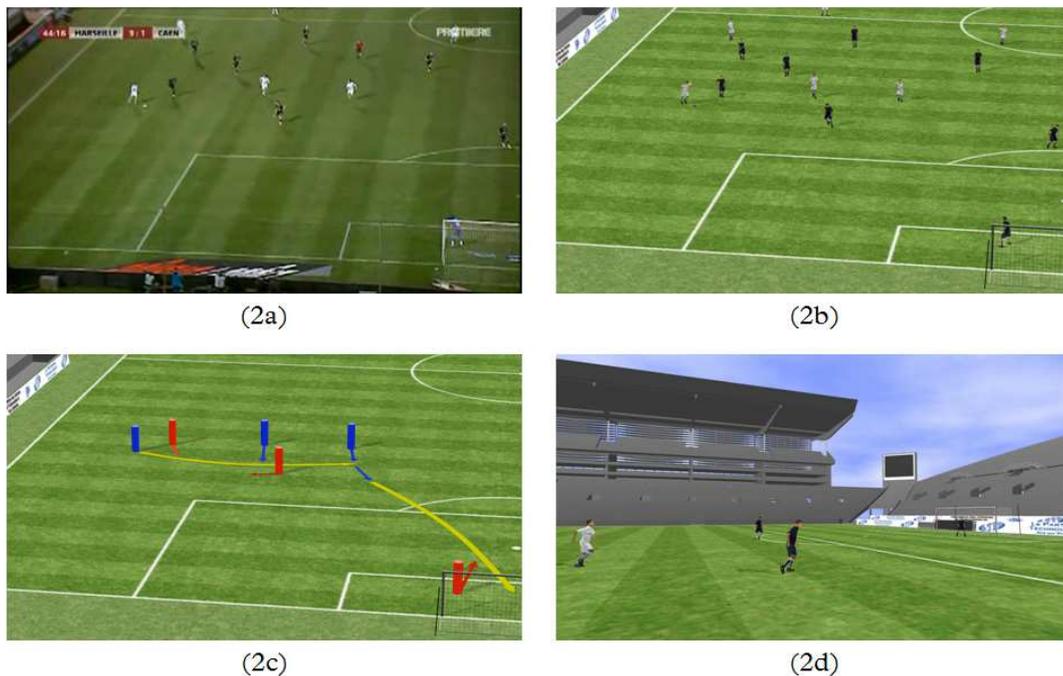


Figure 64 – Situation réelle

La Figure 64 est une illustration d'une situation réelle « traitée » par le logiciel SimulFoot Entraînement : une image est sélectionnée à partir d'une vidéo (2a), puis elle est reconstruite de façon réaliste (2b) et symbolique (2c). Cette représentation virtuelle peut alors être visionnée de différents points de vue, tels que ceux des joueurs (2d). Elle peut également être rejouée telle qu'elle s'est déroulée ou avec une issue différente.

### 2.3 SimulFoot Entraînement

Le développement du logiciel SimulFoot Entraînement (SFE) a été financé par l'organisme ValorPACA, via une démarche de maturation. En effet, ValorPACA est le dispositif mutualisé de valorisation de la recherche publique de la région Provence Alpes Côte d'Azur. Ce dispositif a pour mission de développer la valorisation de la recherche publique et de fédérer le transfert technologique entre la recherche et les entreprises. Il intervient tout au long de la chaîne du transfert en étroite collaboration avec les structures de valorisation de ses membres et avec les incubateurs régionaux. Ces maturations permettent de mieux transférer les résultats de recherche, en les rapprochant de besoins industriels, notamment par le développement de prototypes. Les partenaires financiers de ValorPACA sont le ministre de la Recherche, le Conseil Régional PACA, l'Europe au travers du FEDER, les départements du Vaucluse, des Bouches-du-Rhône, du Var et des Alpes Maritimes, la Ville de Marseille, OSEO et l'INPI.

La réalisation du logiciel SFE a été encadrée par S. Perrotte (chef de projet SII) et développée avec la participation de E. Pillant, S. Bronner et J. Achard (ingénieurs SII).

### 2.3.1 Scénarios dynamiques

SFE est un outil de reconstruction 3D de scènes de football pouvant représenter une scène statique (le moment clé d'une action sportive), mais il propose surtout la création de scènes dynamiques via une interface adaptée. En effet, l'information liée aux mouvements des joueurs est primordiale ; indiquer les placements d'une équipe en fonction des mouvements de l'équipe adverse est un processus classique en tactique sportive. Pour cela, l'entraîneur dispose généralement d'un tableau noir, ou magnétique, sur lequel il symbolise les déplacements des équipes. Il représente alors une séquence de jeu de manière morcelée. Par exemple, un joueur se déplace (dans un premier temps), il reçoit le ballon (dans un second temps) et enfin il se déplace avec ce ballon (dans un troisième temps). La notion de temps est représentée, de façon implicite, en « blocs ». Aussi, pour rester le plus fidèle à cette modélisation, nous utilisons la notion de scénario et d'étape pour définir notre relation au temps dans la scène reconstruite.

Un scénario se définit par un ensemble d'étapes, elles-mêmes constituées d'un ensemble d'actions. Ces dernières sont produites par les entités de la scène (les joueurs) et correspondent généralement à un déplacement ou encore à une passe. Aussi, une entité ne peut pas faire plus d'une action par étape. Pour résumer, les étapes et les différentes actions qui les composent forment un scénario, i.e une suite d'évènements inscrits dans le temps qui crée alors une scène dynamique. Aussi, un scénario est sauvegardé et peut ensuite être rejoué à tout moment.

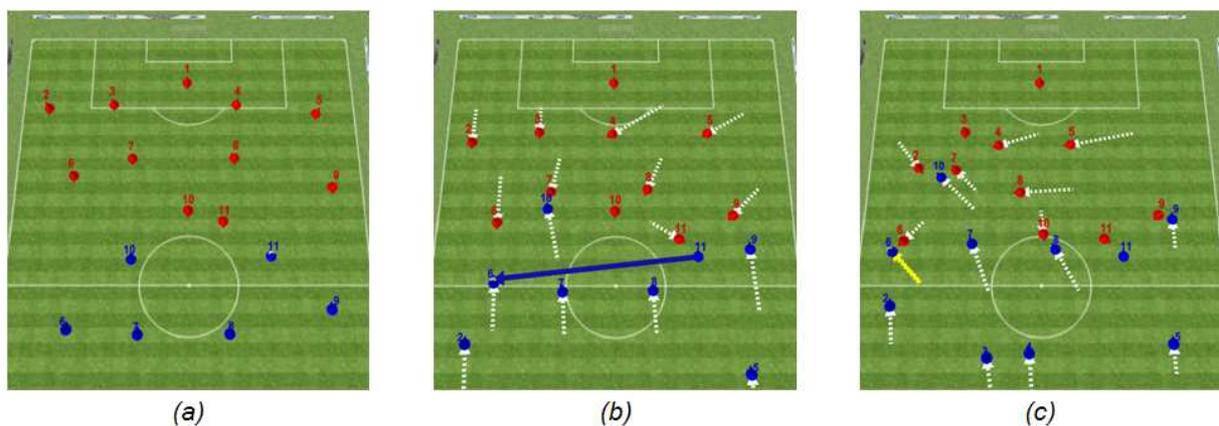


Figure 65 – Un scénario composé d'une étape initiale  $t_0$  (a) et de deux autres étapes  $t_1$  (b) et  $t_2$  (c)

De plus, il est indispensable d'offrir une animation cohérente, en particulier pour le mode de représentation « réaliste ». Pour cela, nous proposons trois types d'animations : marche, course et course rapide. L'animation à affecter à un joueur est déduite à partir de la distance parcourue et du temps d'une étape. La durée d'une étape n'étant pas connue, nous supposons que les joueurs se déplacent à une vitesse moyenne (proche de la course). Ainsi, après avoir calculé la moyenne des déplacements d'une étape il est facile de déterminer la durée de cette dernière. Cette hypothèse permet d'obtenir un comportement global satisfaisant.



Figure 66 – Illustration des différentes animations (de gauche à droite) : marche, course rapide et course

### 2.3.2 Un outil adapté à l'entraîneur

Le support ordinateur et plus précisément son interface (le duo clavier/souris) est un frein à l'utilisation du logiciel SFE ; notamment pour les professionnels du sport (temps d'adaptation, logistique, etc.). Aussi, proposer SimulFoot sur un support adapté au besoin de ces derniers est primordial.

Pour cela, nous avons choisi de remplacer l'ordinateur classique par une tablette PC, essentiellement pour son interface tactile et sa portabilité évidente. Celle qui a été retenue est la tablette PC J3400 ; développée par *motion computing*<sup>38</sup>. En effet, cette dernière a pour intérêt principal d'être un appareil « tout terrain » de par ses spécificités :

- un système d'affichage extérieur qui garantit une excellente lisibilité en cas d'exposition directe à la lumière du soleil ;
- relativement léger (1.6 kg) ;
- un revêtement antichoc renforcé avec du caoutchouc ; chute d'une hauteur d'environ un mètre (conforme à la norme MIL-STD810F) ;
- étanche et facile à nettoyer.

Aussi, adapter l'ergonomie logicielle de SFE au support tablette est capital et consiste à ajuster l'Interface Homme-Machine en fonction de l'écran de la tablette PC et de son contrôleur (le stylet). Il est donc nécessaire de définir plus précisément les boutons (taille et agencement), le contrôle de la caméra (proposer des caméras prédéfinies), etc.

De plus, proposer une symbolique identique à celle utilisée par l'entraîneur permet de faciliter l'adaptation et la transition du tableau noir (ou magnétique) vers le logiciel SFE (cf. 2.2 Etude ergonomique). Pour cela, nous représentons :

- le déplacement par une flèche en pointillés (coloris de l'équipe),
- le déplacement du porteur de la balle par une flèche « ondulée » (colorisée en jaune),
- la passe par une flèche (coloris de l'équipe),

<sup>38</sup> <http://www.motioncomputing.fr/>



Figure 67 – Illustration du logiciel SFE

La Figure 67 est une illustration du logiciel SFE : la tablette PC (a) et l'Interface Homme-Machine (b) composée (de haut en bas) d'une barre d'outils (choix de l'équipe, palette d'outils, mode de représentation, caméras, etc.), d'une « minicarte » pour choisir la position de la caméra, d'une palette d'outils liés à un joueur et les actions associées, d'une palette d'outils (flèches de déplacements, zones de hors-jeu, cercles et zones rectangulaires), d'une barre de temps (*time line*) constituée par les étapes et les actions associées.

L'intérêt du dispositif SimulFoot est justifié par les travaux de H. Ripoll et B. Zoudji (Ripoll, et al., 2009) et une démonstration a été proposée lors du 4<sup>ième</sup> Colloque International « FOOTBALL & RECHERCHE » organisé à Clairefontaine. Cette dernière a permis de recueillir l'approbation des praticiens du sport quant au déploiement d'un tel dispositif.

La solution SimulFoot, en plus d'un logiciel à l'ergonomie adaptée aux entraîneurs, intègre une tablette PC permettant de créer des scénarii de jeux dans des conditions extérieures ou intérieures. Elle permet de remplacer le tableau de l'entraîneur, mais aussi d'enrichir des contenus multimédia de situations de matchs par de la 3D. Cette solution apporte une forte contribution à la didactique du sport, que ce soit pour les entraîneurs, les joueurs ou le grand public.

Dans cette partie nous avons fait état de la réussite du transfert technologique impliqué par le contexte industriel de cette thèse. De plus, nous avons exposé la démarche visant à satisfaire un besoin émergent des praticiens du sport et qui se résume par : la réalisation d'une veille technologique, une étude d'ergonomie cognitive et enfin la réalisation du logiciel SFE. Ce dernier a été testé par l'équipe de Valenciennes dont les retours positifs ont motivé la poursuite de ce projet.

## Conclusion et perspectives

Nous avons présenté dans ce mémoire un travail qui a permis la réalisation d'un prototype logiciel sur une « tablette PC ». Et ce prototype, qui est utilisé à présent par des entraîneurs de football, a nécessité la résolution de problèmes scientifiques du domaine de l'imagerie numérique. Les solutions proposées ont permis de débloquent certains verrous et de faire fonctionner ce simulateur dans des conditions satisfaisantes.

Toutefois, il est intéressant de noter qu'il existe un décalage important entre les objectifs initiaux et ceux qui ont été redéfinis par la suite. Par exemple, alors que le suivi de joueurs (« *tracking* ») était le thème de recherche fort au début de la thèse, il est passé en objectif secondaire par la suite (nous l'avons conservé dans le titre du mémoire, car notre travail s'inscrit néanmoins dans cette ligne, avec cet objectif à atteindre ultérieurement). Cette remarque vient en illustration de l'intérêt des travaux décrits en deuxième partie : en tant que chercheurs en informatique, nous nous étions focalisés sur la résolution de problèmes dits « difficiles » et qui nous paraissaient prioritaires ; les échanges avec les utilisateurs sur fond scientifique en ergonomie cognitive (avec une composante théorique importante) nous ont permis de recadrer efficacement ces objectifs.

Il n'en demeure pas moins que ces questions seront à aborder ultérieurement pour pouvoir proposer un « produit » de meilleure qualité. Parmi les problèmes à traiter, il y aura le suivi des joueurs dans des situations difficiles (regroupement de joueurs dans une zone, croisement de plusieurs joueurs) et l'animation des scènes 3D reconstruite, sachant qu'on ne veut surtout pas produire des images réalistes, mais des séquences qui transmettent l'information pertinente aux entraîneurs et aux joueurs.

## Annexe I. Soustraction de fond

Pour déterminer le mouvement des objets dans une séquence d'images, une approche fortement utilisée de par sa simplicité s'appuie sur l'extraction du fond de l'image. Les objets en mouvement sont alors détectés par la différence de l'image avec le fond. Ces approches ont été classées en différentes catégories par M. Piccardi (Piccardi, 2004) et sont le plus souvent basées sur le filtre médian temporel, la moyenne gaussienne ou encore le mélange gaussien. D'autres types d'approches – par exemple celles basées sur les noyaux de densité, celles qui utilisent les cooccurrences de variation ou encore les valeurs propres – sont également utilisées et, pour plus de détails, on se référera aux travaux de M. Piccardi (Piccardi, 2004).

### 1.1 Filtre médian temporel

B. P. Lo et S. A. Velastin (Lo, et al., 2001) ont utilisé les valeurs médianes des  $n$  dernières images pour modéliser l'arrière-plan (Figure 68). Ce type d'approche nécessite l'historique des valeurs d'intensités des pixels de l'image pour déterminer leur valeur médiane, ce qui peut être problématique. De plus, la valeur à attribuer au seuil reste une partie délicate de cette approche.

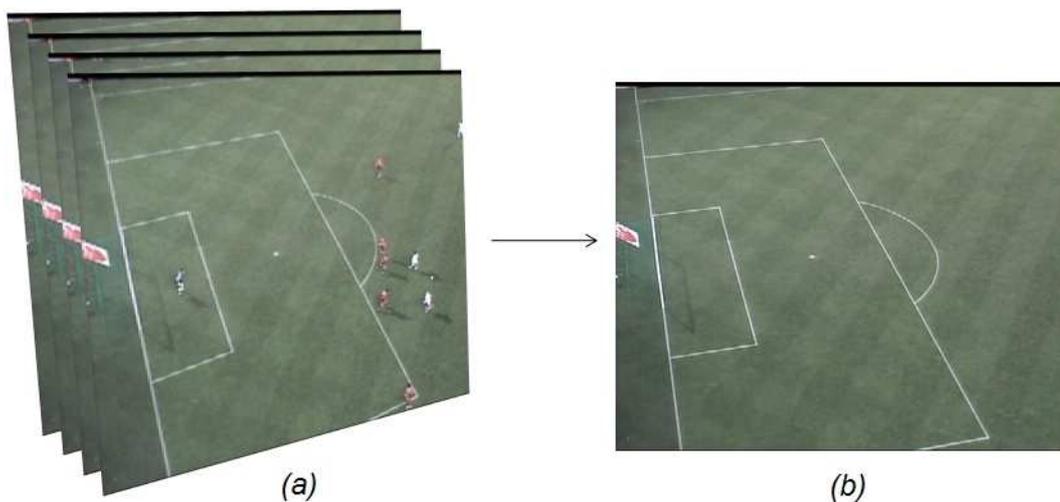


Figure 68 – Extraction de l'arrière-plan par un filtre médian temporel

La Figure 68 représente  $n$  images extraites d'une séquence vidéo (a) et l'arrière-plan obtenu par un filtre médian temporel.

### 1.2 Moyenne gaussienne

C. Wren et al. (Wren, et al., 1997) ont proposé un modèle d'arrière-plan qui s'appuie sur une fonction de densité de probabilité gaussienne. Ils modélisent l'arrière-plan d'une image indépendamment pour chaque position  $\mathbf{x} = (x, y)$  de l'image. De plus, pour ne pas avoir à calculer la fonction densité de probabilité à chaque nouvelle image, ils calculent  $\mu(\mathbf{x}, t)$  et  $\sigma(\mathbf{x}, t)$  respectivement la moyenne et l'écart type :

$$\mu(\mathbf{x}, t) = \alpha I(\mathbf{x}, t) + (1 - \alpha)\mu(\mathbf{x}, t - 1)$$

$$\sigma^2(x, t) = \alpha(I(x, t) - \mu(x, t))^2 + (1 - \alpha) (\sigma^2(x, t - 1) + (\mu(x, t) - \mu(x, t - 1))^2)$$

Avec  $\alpha$  un facteur de pondération. Ainsi, un point de l'image appartient au premier plan si :

$$|I(x, t) - \mu(x, t)| > \theta$$

Le seuil  $\theta$  est déterminé en fonction de  $\sigma$  et correspond alors à  $\theta = K\sigma$ .

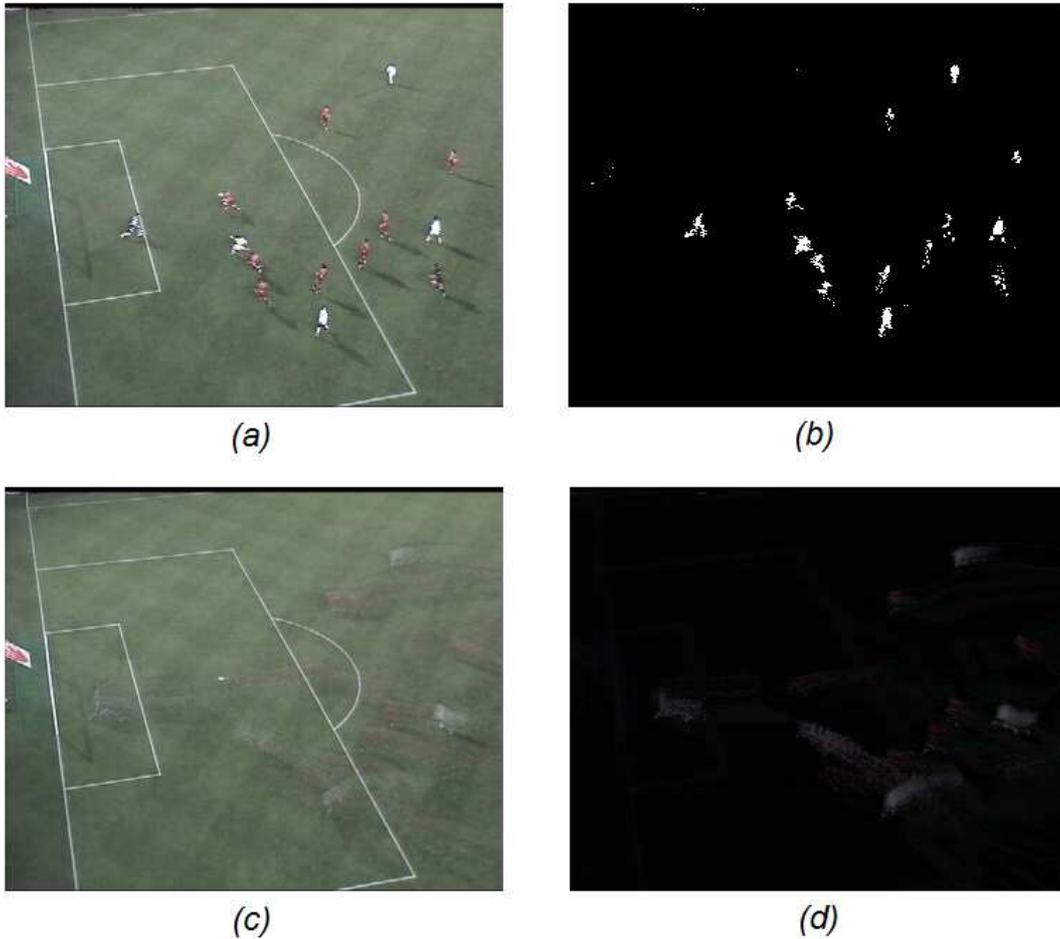


Figure 69 – Soustraction d'arrière-plan à l'aide d'une approche de type moyenne gaussienne

La Figure 69 représente une image extraite d'une séquence vidéo (a), le premier plan de cette image (b), la moyenne gaussienne des  $n$  images précédentes (c) et la variance des  $n$  images précédentes (d) avec  $\alpha = 0.05$  et  $K = 2$ .

### 1.3 Mélange gaussien

C. Stauffer et W. E. L. Grimson (Stauffer, et al., 1999) utilisent quant à eux un mélange gaussien pour caractériser l'arrière-plan et le premier plan d'une image. Ils décrivent la probabilité d'observer une certaine valeur pour un élément  $x$  de l'image au temps  $t$  :

$$P(x_t) = \sum_{i=1}^K \omega_{i,t} \eta(x_i - \mu_{i,t}, \Sigma_{i,t})$$

Avec  $K$  distributions gaussiennes (généralement  $K = 3$  ou  $5$ ).

Pour définir un modèle d'arrière-plan, C. Stauffer et W. E. L. Grimson (Stauffer, et al., 1999) classent les distributions, obtenues avec l'équation ci-dessus, en fonction du ratio entre l'amplitude  $\omega_i$  et l'écart type  $\sigma_i$ . Ils font l'hypothèse qu'au plus cette distribution est haute et compacte, au plus la probabilité d'appartenir à l'arrière-plan est grande. La première distribution d'ordre  $B$  qui vérifie l'équation ci-dessous est alors considérée comme l'arrière-plan (avec  $\theta$  un seuil préalablement fixé) :

$$\sum_{i=1}^B \omega_i > \theta$$

## Annexe II. Calcul du flot optique

S. S. Beauchemin et J. L. Barron (Barron, et al., 1994) (Beauchemin, et al., 1995) ont classé les différentes méthodes de calcul du flot optique en trois principales catégories : les méthodes différentielles, celles basées sur la corrélation et celles basées sur la fréquence. Ces dernières extraient le mouvement principal contenu dans une image – représentée dans le domaine fréquentiel par une sinusoïde complexe – à partir de filtres basés sur l'énergie (Adelson, et al., 1984) ou encore sur la phase (Flette, et al., 1990). Dans cette annexe, nous détaillons les deux premières méthodes (différentielles et basées sur la corrélation) et, pour plus de détails sur les méthodes fréquentielles nous nous référons aux travaux de S. S. Beauchemin et J. L. Barron (Barron, et al., 1994) (Beauchemin, et al., 1995). De plus, l'hypothèse de la cohérence temporelle est souvent utilisée pour affiner le calcul du flot optique et nous montrons comment celle-ci est généralement employée.

### 2.1 Méthodes différentielles

Le calcul du flot optique par des méthodes différentielles se base sur l'hypothèse de « conservation de l'intensité lumineuse » (conservation des données), formulée pour la première fois par C. L. Fennema (Fennema, et al., 1979). Cette hypothèse s'exprime mathématiquement sous la forme de l'équation suivante :

$$I(\mathbf{x}, t) \approx I(\mathbf{x} + \delta\mathbf{x}, t + \delta t)$$

avec  $I(\mathbf{x}, t)$  l'intensité de l'image à la coordonnée  $\mathbf{x} = (x, y)$  et  $\delta\mathbf{x}$  le déplacement subit par le point  $\mathbf{x}$  après un temps  $\delta t$ . Le développement en série de Taylor de cette équation permet, pour un  $\delta\mathbf{x}$  petit, d'obtenir :

$$I(\mathbf{x}, t) = I(\mathbf{x}, t) + \nabla I \delta\mathbf{x} + \delta t I_t + \varepsilon$$

Avec  $\nabla I = (I_x, I_y)$  et  $I_t$  les dérivées partielles de  $I(\mathbf{x}, t)$  et  $\varepsilon$  les termes d'ordres supérieurs à un. Après soustraction par  $I(\mathbf{x}, t)$  de parts et d'autres de l'équation, en négligeant  $\varepsilon$  et en divisant par  $\delta t$  on obtient :

$$\nabla I \cdot \mathbf{v} + I_t = 0$$

Avec  $\mathbf{v} = (u, v)$  la vitesse et  $u = \frac{\delta x}{\delta t}$ ,  $v = \frac{\delta y}{\delta t}$ . Cette équation est appelée « équation de contrainte du flot optique ». Or, cette seule équation ne permet pas la détermination unique de  $\mathbf{v}$  (Figure 70).

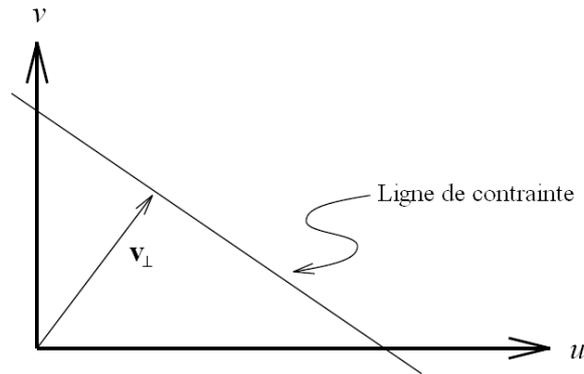


Figure 70 – Ligne de contrainte

Le problème est dit « mal posé<sup>39</sup> ». Seule  $v_{\perp}$  la composante du mouvement dans la direction du gradient peut être estimée. En effet, le vecteur qui vérifie l'équation du flot optique et qui a la plus petite amplitude est appelé « vitesse normale » et correspond alors à :

$$v_{\perp} = \frac{-I_t \nabla I}{\|\nabla I\|_2^2}$$

Ce phénomène est connu sous le nom de « problème de l'ouverture » et seul le mouvement aux points caractéristiques (courbure Gaussienne) peut être complètement estimé avec l'équation de contrainte du flot optique (Figure 71).

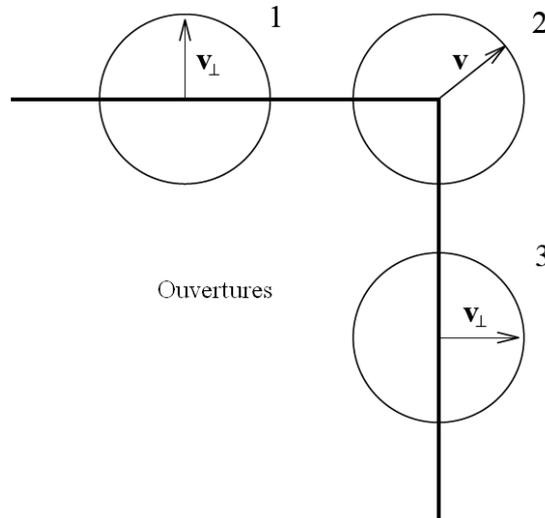


Figure 71 – Problème de l'ouverture

### 2.1.1 Contrainte globale

Une technique classique pour retirer l'indétermination de l'équation du flot optique consiste à utiliser un terme de régularisation. B. K. P. Horn et B. G. Schunck (Horn, et al., 1981) ont été les

<sup>39</sup> L'équation de contrainte du flot optique est une équation linéaire à deux inconnues.

premiers à introduire une contrainte globale, dite « contrainte de lissage », qui s'appuie sur l'hypothèse que le champ de vélocité est régulier (jusqu'au second ordre). Le calcul de l'équation du flot optique est ainsi ramené à un problème d'optimisation défini par :

$$\iint_D (\nabla I \cdot \mathbf{v} + I_t) + \lambda \operatorname{tr}((\nabla \mathbf{v})^T (\nabla \mathbf{v})) \, dx$$

Avec  $D$  le domaine d'intérêt et  $\lambda$  un coefficient qui pondère l'importance du lissage. La solution  $\mathbf{v}$  est donnée par un schéma itératif de type Gauss-Siedel. D'autres travaux s'appuient sur une contrainte de lissage similaire. Nous pouvons citer entre autres les travaux de H. H. Nagel (Nagel, 1983) qui prend en considération le second ordre lors du développement en série de Taylor et utilise une contrainte de lissage supplémentaire (sur l'intensité du gradient). Mais, l'utilisation d'une contrainte globale entraîne des erreurs aux discontinuités spatiales. Celles-ci sont liées à l'utilisation de l'hypothèse de cohérence spatiale dans le cas où elle n'est pas vérifiée. Cette méthode (Nagel, 1983) permet néanmoins « d'adoucir » le lissage dans ces zones de discontinuités.

### 2.1.2 Contrainte locale

B. Lucas et T. Kanade (Lucas, et al., 1981) (Lucas, 1985) supposent que  $\mathbf{v}$  peut être considéré comme localement constant (cohérence spatiale). Ils déterminent un voisinage  $\Omega$  pour lequel  $\mathbf{v}$  est constant et résolvent par une méthode de moindre carré l'équation du flot optique :

$$\sum_{x \in \Omega} (W^2(x) (\nabla I \cdot \mathbf{v} + I_t)^2)$$

avec  $W(x)$  une fonction de pondération (fonction Gaussienne). Le problème d'ouverture peut également être résolu de façon analytique par différentiation de l'équation du flot optique. On obtient ainsi une équation au second degré des dérivées de l'intensité. Ce type de contrainte permet d'obtenir plusieurs équations et, lorsqu'elles sont non singulières (systèmes non sous déterminés), peuvent être utilisées pour obtenir complètement le flot optique. Supposer le flot optique localement constant entraîne naturellement des erreurs aux discontinuités spatiales.

### 2.1.3 Contrainte de contour et de surface

L'équation du flot optique, exprimée sous la forme différentielle, peut être contrainte par les contours contenus dans une image. Le principe de cette technique est d'extraire ces contours, d'obtenir ainsi l'ensemble des vélocités normales le long de ces contours pour ensuite estimer la vélocité à partir de cet ensemble de vecteurs (Figure 72).

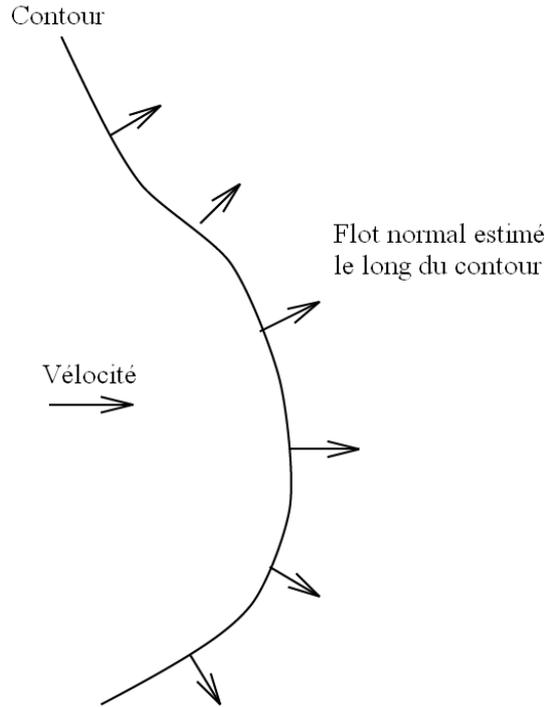


Figure 72 – Vitesse estimée à l'aide d'une contrainte de contour

Cette contrainte a été formulée par E. C. Hildreth (Hildreth, 1983). Il cherche à déterminer la vitesse telle qu'elle minimise, le long d'un contour  $s$  donné, la variation entre les différentes vitesses normales :

$$\int \frac{\delta v}{\delta s} ds$$

En pratique, E. C. Hildreth (Hildreth, 1983) minimise la fonction :

$$\int \left( \frac{\delta v}{\delta s} \right)^2 + \alpha (v \cdot \hat{n} - \|v_{\perp}\|_2)^2 ds$$

avec  $\hat{n}$  le vecteur unité de direction  $v_{\perp}$  et  $\alpha$  un facteur de pondération. M. Brady et S. Gong (Gong, et al., 1990) formulent une contrainte similaire et minimisent la variation avec les vitesses tangentes :

$$\int \left( \frac{\delta v}{\delta s} \right)^2 + \alpha (v \cdot \hat{n} - \|v_{\perp}\|_2)^2 + \beta (v \cdot \hat{t} - \|v_{\parallel}\|_2)^2 ds$$

avec  $\hat{t}$  le vecteur unité perpendiculaire au vecteur  $\hat{n}$  et  $\beta$  un coefficient de pondération.

#### 2.1.4 Contrainte multiple

Le principe des techniques de contraintes multiples est d'utiliser plusieurs instances de l'équation du flot optique pour déterminer sans ambiguïtés le vecteur de vitesse. V. Markandey et B. E. Flinchbaugh (Markandey, et al., 1990) ont mis en œuvre une approche multispectrale pour calculer le flot optique. Ainsi, pour chacune des composantes  $E$  de l'image, l'équation de contrainte du flot optique est formulée par :

$$\nabla E \cdot \mathbf{v} + I_t = 0$$

P. Golland et A. M. Bruckstein (Golland, et al., 1997) ont proposé une approche similaire pour calculer le flot optique d'une image couleur. Leur méthode s'appuie sur l'hypothèse de « conservation de la couleur ». Ils s'appuient sur le modèle RGB<sup>40</sup> et considèrent que l'hypothèse de conservation de la luminosité est valide sur chacune des composantes de la couleur. Ils obtiennent ainsi un système à trois équations :

$$\begin{aligned}\nabla R \cdot \mathbf{v} + R_t &= 0 \\ \nabla G \cdot \mathbf{v} + G_t &= 0 \\ \nabla B \cdot \mathbf{v} + B_t &= 0\end{aligned}$$

Il est ainsi possible de résoudre ce système surdéterminé en tenant compte seulement de deux équations, ou de façon plus « élégante », en utilisant une résolution par moindre carré. J. L. Barron et R. Klette (Barron, et al., 2002) ont comparé les différentes techniques de calcul du flot optique utilisant des contraintes couleur.

## 2.2 Méthode par corrélation

Ces méthodes définissent le flot optique comme le décalage  $\mathbf{d} = (d_x, d_y)$  que subit une région  $R$  dans le temps (Figure 73).

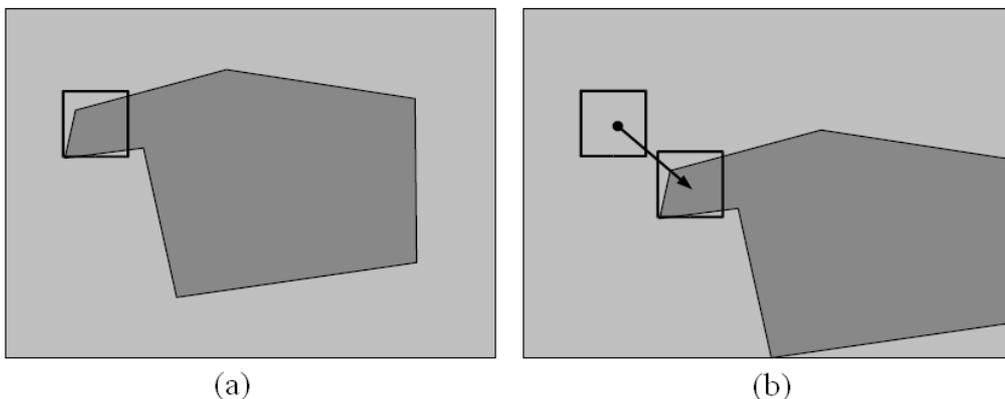


Figure 73 – Décalage détecté par une méthode par corrélation

La Figure 73 représente une image au temps  $t$  (a), une image au temps  $t + 1$  et le déplacement d'une zone donnée (rectangulaire).

L'approche la plus classique pour faire correspondre deux régions est de maximiser une mesure de similitude (ou de minimiser une mesure de dissimilitude). On peut citer l'exemple de l'utilisation de la mesure SSD comme mesure de dissimilitude et qui se formalise, appliquée à une séquence d'images, par :

$$\sum_R (I(x, t) - I(x + \mathbf{d}, t + 1))^2$$

<sup>40</sup> Composante Rouge, Verte(Green), Bleue.

Appliquer cette méthode à l'ensemble de l'image demande un temps de calcul important. En pratique, on définit un voisinage de recherche et on suppose que le décalage  $d$  est inférieur à un déplacement  $D$  maximum (Figure 74).

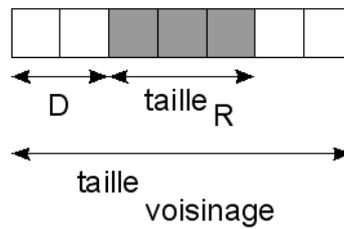


Figure 74 – Voisinage de recherche

La Figure 74 est un exemple 1D de voisinage de recherche : la région  $R$  est une région de trois pixels et  $D$ , le déplacement maximum, est supposé égal à deux pixels.

Contrairement aux méthodes différentielles ou encore fréquentielles, les méthodes par corrélation font intervenir un vecteur de déplacement  $d$  dont les coordonnées sont des entiers. Les méthodes par corrélation « classiques » ne permettent pas de déterminer les déplacements de façon sub-pixellique (Figure 75).

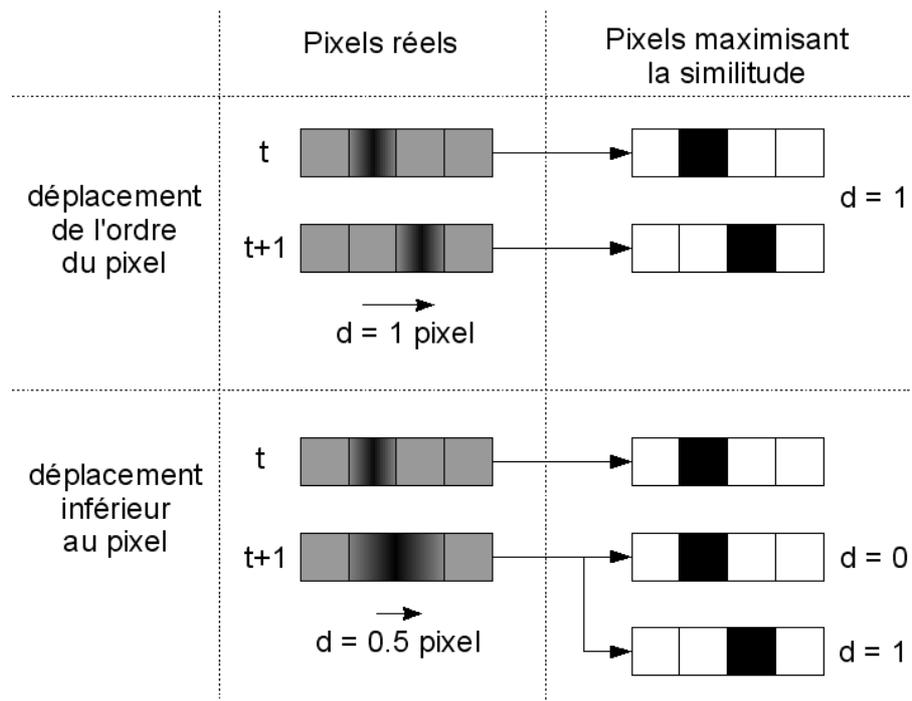


Figure 75 – Exemple 1D du problème lié aux déplacements inférieurs au pixel

## 2.3 Affinement du flot optique

### 2.3.1 Procédé hiérarchique

Traditionnellement, le flot optique d'une image est calculé à partir d'une résolution unique, directement liée au capteur utilisé lors de l'acquisition, et entraîne certaines difficultés pour la

détection de grands mouvements dans l'image. Le sous-échantillonnage ou les effets de crénelages rendent inappropriée l'utilisation de l'équation du flot optique. Une manière classique pour surmonter ce problème est d'appliquer un procédé hiérarchique aux techniques de détermination du flot optique.

Ce procédé décompose dans un premier temps les images selon différentes résolutions puis calcule le vecteur de déplacement, en fonction de chaque niveau de résolution, par une méthode « en cascade » (Figure 76).

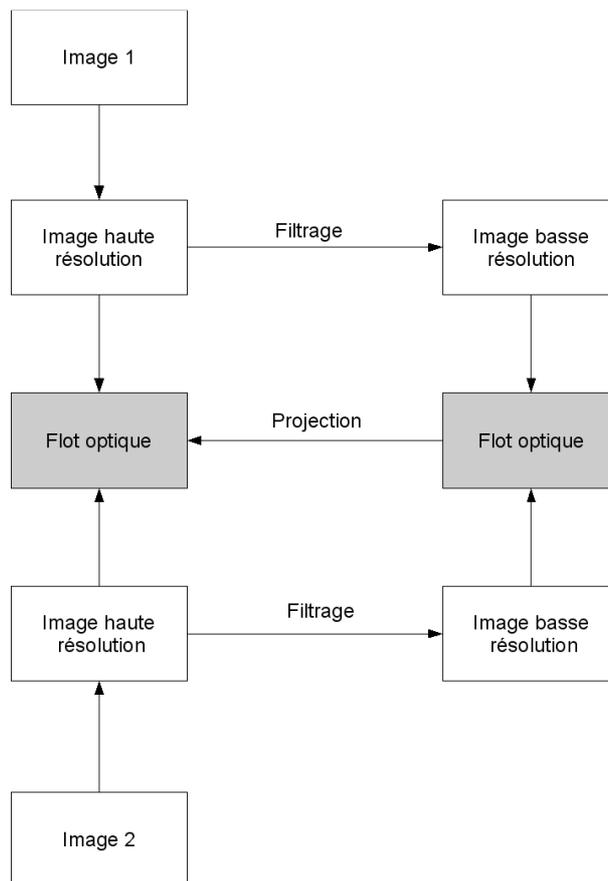


Figure 76 – Procédé hiérarchique

Le vecteur de déplacement est estimé à partir du niveau le plus bas. Il est ensuite projeté aux niveaux supérieurs de résolution, puis remis à jour. Ce procédé est ainsi répété jusqu'à l'obtention du vecteur de déplacement à la plus haute résolution. Il s'applique à la plupart des techniques de détermination du flot optique. De nombreux travaux ont été entrepris pour calculer le flot optique par un procédé hiérarchique, notamment par des méthodes différentielles (Glazer, 1987), fréquentielles (Heeger, 1987) et de corrélation (Anandan, 1989).

### 2.3.2 Raffinement temporel

La plupart des méthodes de calcul de flot optique présentées auparavant ne tiennent pas compte des flots optiques précédemment calculés. L'ajout de l'hypothèse de continuité temporelle permet d'obtenir le flot optique de façon incrémentielle.

L'indétermination de l'équation du flot optique est due au manque d'informations quant à la direction du gradient spatial. Dans de nombreux cas, cette indétermination peut être levée par l'ajout d'une contrainte de cohérence temporelle. Ainsi, T. M. Chin *et al.* (Chin, et al., 1994) ont proposé une extension de la méthode de B. Horn et B. Shunck (Horn, et al., 1981). Ils obtiennent l'équation suivante :

$$\int_0^\tau \iint_D (\nabla I \cdot \mathbf{v} + I_t)^2 + \lambda_1 \text{tr}((\nabla \mathbf{v})^T (\nabla \mathbf{v})) + \lambda_2 \left( \frac{\delta}{\delta t} \mathbf{v} \right)^2 dxdt$$

Avec  $[0; \tau]$  l'intervalle de temps considéré et  $\lambda_1, \lambda_2$  des coefficients de pondération. Cette méthode permet de lisser le champ de vélocité dans l'espace, mais aussi dans le temps et elle favorise ainsi les transitions « douces ».

La cohérence temporelle peut également être utilisée via des approches probabilistes telles que les filtres de Kalman (Kalman, 1960) (Welch, et al., 1995) (Welch, et al., 1997) ou encore les filtres particuliers (Handschin, 1969) (Handshin, 1970) (Kitagawa, 1996) (Gordon, et al., 1993) (Gordon, et al., 1995) (Isard, et al., 1996) (Isard, et al., 1998). Ces approches sont particulièrement robustes, mais elles impliquent une mise en œuvre lourde qui n'est pas aisée (applications « temps réel » par exemple).

De plus, l'hypothèse de continuité temporelle nécessite la formulation d'un certain nombre de conditions, notamment sur la scène et sur les objets (apparences, mouvements, etc.), qui ne sont pas toujours remplies. Par exemple, les comportements de joueurs de football – plus précisément leurs déplacements – sont erratiques et par conséquent difficilement prédictibles.

## Bibliographie

**Aanaes, H., et al. 2008.** Model-based satellite image fusion. *IEEE Transactions on Geoscience and Remote Sensing*. May 2008. Vol. 46, 5, pp. 1336-1346.

**Adelson, E. H., et al. 1984.** Pyramid methods in image processing. *RCA Engineer*. 1984.

**Aggarwal, J. K. et Cai, Q. 1999.** Human motion analysis : a review. *CVIU*. 1999. Vol. 74, 3, pp. 428-440.

**Almansa, A., Desolneux, A. et Vamech, S. 2003.** Vanishing points detection without any a priori information. *IEEE Trans. on PAMI*. 2003. Vol. 25(4), pp. 502-507.

**Anandan, P. 1989.** A computational framework and an algorithm for the measurement of visual motion. *IJCV*. January 1989. Vol. 2, 3, pp. 283-310.

**Asada, H. et Brady, M. 1986.** The curvature primal sketch. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Washington, DC, USA : IEEE Computer Society, January 1986. Vol. 8, 1, pp. 2-14. ISSN:0162-8828.

**Barnard, S. T. 1983.** Interpreting perspective images. *Artificial Intelligence*. 1983. Vol. 21, pp. 435-462.

**Barron, J. L. et Klette, R. 2002.** Quantitative color optical flow. *ICPR*. 2002. pp. 251-255.

**Barron, J. L., et al. 1994.** Performance of optical flow techniques. *International Journal of Computer Vision*. 1994. Vol. 12, pp. 43-77.

**Beauchemin, S. S. et Barron, J. L. 1995.** The computation of optical flow. *ACM Comput. Surv.* New York, NY, USA : ACM, 1995. Vol. 27, 3, pp. 433-466.

**Beaudet, P. R. 1978.** Rotationally invariant image operators. *International Conference on Pattern Recognition*. Kyoto, Japan : s.n., 1978. pp. 579-583.

**Black, M. J. 1992.** Robust incremental optical flow. *PhD thesis, Yale University*. 1992.

**Brillault-O'Mahony, B. 1991.** New method for vanishing point detection. *Computer Vision, Graphics and Image Processing*. 1991. Vol. 54(2), pp. 289-300.

**Brown, L. G. 1992.** A survey of image registration techniques. *ACM Computing Surveys (CSUR)*. New York, USA : s.n., December 1992. Vol. 24, 4, pp. 325-376.

**Canny, J. 1986.** Computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*. 1986. Vol. 8, pp. 679-714.

**Chambon, S. 2005.** Mise en correspondance stéréoscopique d'images couleur en présence d'occultation. [Thèse de doctorat, Université Paul Sabatier]. Toulouse, France : s.n., Décembre 2005.

**Chin, R. T. et Dyer, C. R. 1986.** Model-based recognition in robot vision. *ACM Comput. Surv.* New York, NY, USA : ACM, 1986. Vol. 18, 1, pp. 67-108.

- Chin, T. M., Karl, W. C. et Willsky, A. S. 1994.** Probabilistic and sequential computation of optical flow using temporal coherence. *IP*. November 1994. Vol. 3, 6, pp. 773-788.
- Davies, E. R. 1997.** Machine vision : theory, algorithms, practicalities. *Academoc Press*. 1997.
- Debevec, P. E. 2005.** Making "The Parthenon". *6th International Symposium on Virtual Reality, Archaeology, and Cultural Heritage*. Pisa, Italy : s.n., December 2005.
- . **1996.** Modeling and rendering architecture from photographs. *Ph.D. Thesis*. s.l. : UC Berkley, December 1996.
- Debevec, P. E., Borshukov, G. et Yu, Y. 1998.** Efficient view-dependent image-based rendering with projective texture-mapping. *9th Eurographics Rendering Workshop*. s.l., Vienna : Austria, June 1998.
- Debevec, P. E., Taylor, C. J. et Malik, J. 1996.** Modeling and rendering architecture from photographs : a hybrid geometry - and image - based approach. *SIGGRAPH'96*. August 1996.
- Demeris, A. M., et al. 2001.** Piste : mixed reality for sports TV. *International Workshop on Very Low Bitrate Video Coding (VLBV)*. Athens, Greece : s.n., 2001.
- Dempster, A., Laird, N. et Rubin, D. 19977.** Maximum likelihood from incomplete data via the EM algorithm. *Series B Journal of the Royal Statistical Society*. 19977. Vol. 39, 1, pp. 1-38.
- Deriche, R. et Faugeras, O. 1990.** 2-D curve matching using high curvature points : application to stereo vision. *IEEE Proceedings Pattern Recognition*. Atlantic City, NJ, USA : s.n., June 1990. Vol. 1, pp. 240-242. ISBN: 0-8186-2062-5.
- Deriche, R. et Giraudon, G. 1990.** Accurate corner detection : an analytical study. *Proceedinfs of the 3rd International Conference on Computer Vision*. 1990. pp. 66-70.
- Deriche, R. 1987.** Using canny's criteria to derive a recursively implemented optimal edge detector. *Int. J. Computer Vision*. April 1987. Vol. 1, pp. 167-187.
- Didierjean, A. et Marmèche, E. 2005.** Anticipatory representation of visual basketball scenes by novice and expert players. *Visual Cognition*. 2005. Vol. 12, 2, pp. 265-283.
- Dreschler, L. et Nagel, H. H. 1982.** On the selection of critical points and local curvature extrema of region boundaries for interframe matching. *Proceedings of ICPR*. 1982. pp. 542-556.
- Duda, R. O. et Hart, P. E. 1972.** Use of the hough transform to detect lines and curves in pictures. *Comm. ACM*. January 1972. Vol. 15, pp. 11-15.
- Fennema, C.L. et Thompson, W.B. 1979.** Velocity determination in scenes containing several moving objects. *CGIP*. April 1979. Vol. 9, 4, pp. 301-315.
- Ferrari, V., Ripoll, H. et Sebbane, M. 2009.** What is the nature of perceptual knowledge of experts in collective games? *Symposium of ISSP world scientific congress*. Marrakech : s.n., 17-21 Juin 2009.
- Figuroa, P., et al. 2004.** Tracking soccer players using the graph representation. *17th International Conference on Pattern Recognition (ICPR'04)*. Cambridge, UK : s.n., August 2004. Vol. 4. ISBN 0-7695-2128-2.

- Fischler, M. A. et Bolles, R. C. 1981.** Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*. 1981. Vol. 24, 6, pp. 381-395.
- Fitzgibbon, A. W. et Zisserman, A. 1998.** Automatic 3D model acquisition and generation of new images from video sequences. *Proceedings of European Signal Processing Conference (EUSIPCO'98)*. Rhodes, Greece : s.n., 1998. pp. 1261-1269.
- Flette, D. J. et Jepson, A. D. 1990.** Computation of component image velocity from local phase information. *Int. Journal of Computer Vision*. 1990.
- Gabriel, P. F., et al. 2003.** The state of the art in multiple object tracking under occlusion in video sequences. in *Proceedings of the Advanced Concepts for Intelligent Vision System (ACIVS'03)*. September 2003. pp. 166-173.
- Gabriel, P., et al. 2005.** Object tracking using color interest points. in *Proc. of the IEEE Int. Conf. on Advanced Video and Signal based Surveillance (AVSS'05)*. 2005.
- Gavrila, D. M. 1999.** The visual analysis of human movement : a survey. *CVIU*. 1999. Vol. 73, 1, pp. 82-98.
- Glazer, F. 1987.** Hierarchical gradient-based motion detection. *DARPA*. 1987. pp. 733-748.
- Gohlke, M. 2009.** A new approach for model-based recognition using color regions. 2009.
- Golland, P. et Bruckstein, A. M. 1997.** Motion from color. *Comput. Vis. Image Underst.* New York, NY, USA : Elsevier Science Inc., 1997. Vol. 68, 3, pp. 346-362.
- Gong, S. et Brady, M. 1990.** Parallel computation of optic flow. *Proceedings of the first european conference on Computer vision (ECCV)*. Antibes, France : Springer-Verlag New York, Inc., 1990. pp. 124-133.
- Gordon, N. et Salmond, D. 1995.** Bayesian state estimation for tracking and guidance using the bootstrap filter. *Journal of Guidance, Control and Dynamics*. 1995. Vol. 18, 6, pp. 1434-1442.
- Gordon, N., Salmond, D. et Smiths, A. 1993.** Novel approach to nonlinear/non-Gaussian Bayesian state estimator. *IEE Proc. F*. 1993. Vol. 140, 2, pp. 107-113.
- Gouet, V. et Boujemaa, N. 2002.** About optimal use of color points of interest for content-base retrieval. *Rapport de recherche de l'INRIA*. April 2002.
- Grau, O., et al. 2006.** A free-viewpoint video system for visualisation of sport scenes. *Proc. of IBC 2006*. Amsterdam, Netherlands : s.n., September 2006.
- Grau, O., Prior-Jones, M. et Thomas, G. 2005.** 3D modelling and rendering of studio and sport scenes for TV applications. *WIAMIS 2005*. Montreux, Switzerland : s.n., April 2005.
- Handschin, J. 1969.** Monte Carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering. *International Journal of Control*. 1969. Vol. 9, 5, pp. 543-559.

- Handshin, J. 1970.** Monte Carlo techniques for prediction and filtering of non-linear stochastic processes. *Automatica*. 1970. Vol. 6, pp. 555-562.
- Harris, C. et Stephens, M. 1988.** A combined corner and edge detector. *Proceeding of the 4th Alvey Vision Conference*. 1988. pp. 147-151.
- Hartigan, J. A. et Wong, M. A. 1979.** A k-means clustering algorithm. *Applied Statistics*. 1979. Vol. 28, 1, pp. 100-108.
- Hayet, J. B., Piater, J. H. et Verly, J. G. 2005.** Fast 2D model-to-image registration using vanishing points for sports video analysis. *ICIP05*. 2005. III, pp. 417-420.
- Heeger, D. J. 1987.** Optical flow using spatiotemporal filters. *ICCV*. 1987. pp. 181-190.
- Heikkila, J. et Silven, O. 1997.** A four-step camera calibration procedure with implicit image correction. *Proc. of IEEE Computer Vision and Pattern Recognition*. 1997. pp. 1106-1112.
- Hildreth, E.C. 1983.** The computation of the velocity field. *MIT AI Memo*. 1983.
- Horn, B. K. P. et Schunck, B. G. 1981.** Determining optical flow. *Artificial Intelligence*. 1981. Vol. 17, pp. 185-203.
- Huseby, R. B., Halck, O. M. et Solberg, R. 2005.** A model-based approach for geometrical correction of optical satellite images. *International Journal of Remote Sensing*. August 2005. Vol. 26, 15, pp. 3205-3223.
- Irani, M. et Anandan, P. 1999.** About direct methods. *ICCV Workshop on Vision Algorithms*. 1999. pp. 267-277.
- Isard, M. et Blake, A. 1998.** A smoothing filter for Condensation. *Proc. 5th European Conf. Computer Vision*. s.l., Germany : Springer Verlag, 1998. Vol. 1, pp. 767-781.
- Isard, M. et Blake, A. 1996.** Visual tracking by stochastic propagation of conditional density. *Proc. 4th European Conf. Computer Vision*. s.l., Cambridge : Springer Verlag, 1996. pp. 343-356.
- Iwase, S. et Saito, H. 2002.** Tracking soccer player using multiple views. *Proc. IAPR Work Mach. Vis. Appl.* 2002. pp. 102-105.
- **2003.** Tracking soccer players based on homography among multiple views. *Proc. SPIE*. 2003. pp. 283-292.
- Joshi, M. V., Bruzzone, L. et Chaudhuri, S. 2006.** A model-based approach to multiresolution fusion in remotely sensed images. *IEEE Transactions on Geoscience and Remote Sensing*. September 2006. Vol. 44, 9, pp. 2549-2562.
- Kalantari, M., Jung, F., Guédon, J. P. et Papanoditis, N. 2008.** Détection automatique des points de fuite et calcul de leur incertitude à l'aide de la géométrie projective. *Reconnaissance des Formes et Intelligence Artificielle (RFIA)*. Amiens, France : s.n., janvier 2008.
- Kalman, R. E. 1960.** A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering*. 1960. Vol. 82(Series D), pp. 35-45.

- Kass, M., Witkin, A. et Terzopoulos, D. 1987.** Snakes : active contour models. *International Journal of Computer Vision*. 1987. pp. 259-268.
- Keysers, D., et al. 2003.** Statistical framework for model-based image retrieval in medical applications. January 2003. Vol. 12, 1, pp. 59-68.
- Kitagawa, G. 1996.** Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computer and Graphical Statistics*. 1996. Vol. 5, 1, pp. 1-25.
- Kitchen, L. et Rosenfeld, A. 1982.** Grey-level corner detection. *Pattern Recogn. Lett.* 1982. pp. 95-102.
- Kühnapfel, U. et Neisius, B. 1993.** CAS-based graphical simulation in endoscopic surgery. *Endoscopic Surgery and Allied Technologies*. s.l. : Georg Thiee Verlag, 1993. Vol. 1, 3, pp. 181-184.
- Kühnapfel, U., Cakmak, H. K. et Maass, H. 1999.** 3D modeling for endoscopic surgery. *IEEE Symposium on Simulation, Delft University*. Delft, NL : s.n., 13 October 1999. Vol. 22-32. ISBN: 90-804551-7-2.
- . **2000.** Endoscopic surgery training using virtual reality and deformable tissue simulation. *Computer & Graphics* 24. s.l. : Elsevier, 2000. pp. 671-682.
- Laurent, E. et Ripoll, H. 2009.** Extending the rather unnoticed gibsonian view that 'perception is cognitive': Development of the enactive approach to perceptual-cognitive expertise. [éd.] H. Ripoll & M. Raab D. Araujo. *Perspectives on cognition and action in sport*. New York : Nova Science Publishers, 2009.
- Laurent, E., et al. 2006.** Expertise in basketball modifies perceptual discrimination abilities, underlying cognitive processes, and visual behaviours. *Visual Cognition*. 2006. Vol. 13, 2, pp. 247-271.
- Le Troter, A. 2006.** Mise en correspondance d'images et de modèles : application à la reconstruction 3D de scènes sportives. *thèse de doctorat, Université Aix-Marseille II*. september 2006.
- Lo, B.P.L. et Velastin, S.A. 2001.** Automatic congestion detection system for underground platforms. *Proceedings of International Symposium on Intelligent Multimedia, Video and Speech Processing*. 2001. pp. 158-161.
- Lucas, B. D. et Kanade, T. 1981.** An iterative image registration technique with an application to stereo vision. *IJCAI*. 1981. pp. 674-679.
- Lucas, B.D. 1985.** Generalized image matching by the method of differences. *CMU-CS-TR*. 1985.
- Lutton, E., Ma Itre, H. et Lopez-Krahe, J. 1994.** Contribution to the determination of vanishing points using Hough transform. *IEEE Trans on Pattern Analysis and Machine intelligence*. 1994. Vol. 16(4), pp. 430-438.
- M., Coughlan J. et Yuille, A. L. 1999.** Manhattan world: compass direction from single image by Bayesian inference. *Proceedings of the 11th International Conference on Computer Vision (ICCV'99)*. Kerkyra, Greece : s.n., September 1999. pp. 941-947.

- Magee, M. J. et Aggarwal, J. K. 1984.** Determining vanishing points from perspective images. *Computer Vision, Graphics and Image Processing*. 1984. Vol. 26, pp. 256-267.
- Maintz, J. et Viergever, M. 1998.** A survey of medical image registration. 1998. Vol. 2, 1, pp. 1-36.
- Makela, T. 2003.** A 3-D model-based registration approach for the PET, MR and MCG cardiac data fusion. *Medical Image Analysis*. September 2003. Vol. 7, 3, pp. 377-389.
- Markandey, V. et Flinchbaugh, B. E. 1990.** Multispectral constraints for optical flow computation. *ICCV*. 1990. pp. 38-41.
- Mavromatis, M. et Dias, P., Sequeira, J. 2007.** 3D reconstruction of soccer sequences using non calibrated video cameras. *ICAR'07*. Montréal : s.n., Août 2007.
- Mavromatis, S., Baratgin, J. et Sequeira, J. 2003.** Analyzing team sport strategies by means of graphical simulation. *International Conference on Image and Signal Processing (ICISP'03)*. Agadir, Maroc : s.n., Juin 2003.
- Medioni, G. et Yasumoto, Y. 1987.** Corner detection and curve representation using cubic B-spline. *Computer Vision, Graphics, and Image Processing*. San Diego, CA, USA : Academic Press Professional, Inc., September 1987. Vol. 39, 3, pp. 267-278. ISSN:0734-189X.
- Menaut, A. 1998.** Le réel et le possible dans la pensée tactique. *Presses Universitaires de Bordeaux*. 1998.
- Moeslund, T. B. et Granum, E. 2001.** A survey of computer vision-based human motion capture. *CVIU*. 2001. Vol. 81, 3, pp. 231-268.
- Moeslund, T., Hilton, A. et Krüger, V. 2006.** A survey of advances in vision-based human motion capture and analysis. *CVIU*. 2006. Vol. 104, pp. 90-126.
- Mokhtarian, F. et Mackworth, A. 1986.** Scale-based description and recognition of planar curves and two-dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Washington, DC, USA : IEEE Computer Society, January 1986. Vol. 8, 1, pp. 34-43. ISSN:0162-8828.
- Montesinos, P., Gouet, V. et Deriche, R. 1998.** Differential invariants for color images. *Proceedings of the 14th International Conference on Pattern Recognition (ICPR)*. Washington, DC, USA : IEEE Computer Society, 1998. p. 838. ISBN: 0-8186-8512-3.
- Morevec, H. P. 1977.** Towards automatic visual obstacle avoidance. *Proceedings of the 5th International Joint Conference on Artificial Intelligence(IJCAI)*. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1977. Vol. 2, p. 584.
- Mundy, J. L. et Zisserman, A. 1992.** Geometric invariance in computer vision. *MIT Press*. 1992.
- Nagel, H. H. 1983.** Displacement vectors derived from second-order intensity variations in image sequences. *Comput. Vision, Graphics, Image processing (CVGIP)*. January 1983. Vol. 21, 1, pp. 85-117.
- Noble, J. A. 1988.** Finding corners. *Image and Vision Computing*. Newton, MA, USA : Butterworth-Heinemann, May 1988. Vol. 6, 2, pp. 121-128. ISBN:0262-8858.

- Okuma, K., Little, J. J. et Lowe, D. G. 2004.** Automatic rectification of long image sequences. *in Proc. of the Asian Conf. on Computer Vision (ACCV'04)*. 2004.
- Petit, J. P. et Ripoll, H. 2008.** Scene perception and decision making in sport simulation : a masked priming investigation. *International Journal of Sport Psychology*. 2008. Vol. 39, 1, pp. 1-19.
- Piaget, J. 1946.** Le développement de la notion de temps chez l'enfant. 1946.
- Piccardi, M. 2004.** Background subtraction techniques: a review. *IEEE International Conference on Systems, Man and Cybernetics* . 2004. Vol. 4, pp. 3099-3104.
- Pollefeys, M. 2000.** 3D modelling from images - tutorial notes. *conjunction with ECCV 2000*. Dublin, Ireland : s.n., June 2000.
- Poplu, G., et al. 2008.** How do expert soccer players encode visual information in order to make decisions in simulated game situations? *Research Quarterly for Exercise and Sport*. 2008. Vol. 79, 3, pp. 392-398.
- Poplu, G., et al. 2003.** What kind of processes underlie decision-making in soccer simulation? *International Journal of Sport and Exercise Psychology*. 2003. Vol. 1, 4, pp. 390-405.
- Quan, L. et Mohr, R. 1989.** Determining perspective structures using hierarchical Hough transform. *Pattern Recognition Letters*. 1989. Vol. 9, pp. 279-286.
- Ripoll, H. 1997.** Activités physiques et sportives et développement cognitif et sensori-moteur de l'enfant et de l'adolescent. [éd.] J. Cazorla & G. Robert. *L'enfant, l'adolescent et le sport*. [Actes du 4ème colloque international des cadres techniques et sportifs de la Guadeloupe]. 1997. pp. 96-103.
- **2008.** Le mental des champions : comprendre la réussite sportive. Paris : Payot, 2008.
- **1979.** Le traitement de l'information de données visuelles dans les situations tactiques en sport. L'exemple du basket-ball. *Travaux et Recherches en E.P.S.* 4. 1979. pp. 99-104.
- **2004.** Les opérations cognitives en contexte sportif. [éd.] J. Larue & H. Ripoll. *Manuel de Psychologie du Sport. Tome 1: Bases psychologiques de la performance sportive*. [Editions Revue EP.S, Coll. Recherche et Formation]. Paris : s.n., 2004. pp. 151-280.
- **1996.** Les recherches actuelles en neurosciences du sport : le niveau cognitif. Essai de formalisation des rapports théorie-pratique. *Recherche et Pratiques en APS*. [EPS N°28. Revue EP.S]. Paris : s.n., 1996. pp. 58-66.
- **1988.** Stratégies de prise d'informations visuelles dans les tâches de résolution de problèmes tactiques en sport. *Traitement des informations visuelles, prises de décision et réalisation de l'action en sport*. Paris : I.N.S.E.P : s.n., 1988. pp. 329-354.
- Ripoll, H., Aubert, C. et Mavromatis, S. 2003.** Mechanisms involved in the change of point of view into a 3D-image simulation sport. *FEPSAC-2003. Xith Europeen Congress of Sport Psychology*. Copenhague, Danemark : s.n., 2003.

- Ripoll, H., et al. 2004.** Un simulateur pour l'entraînement en football. *Communication présentée au 3ièmes journées internationales des sciences en football, INSEP*. Paris, France : s.n., 2004.
- Ripoll, H., Zoudji, B. et Lucia, L. 2009.** Du tableau au simulateur pour apprendre la tactique en Football. *Les évolutions actuelles du football : le joueur d'aujourd'hui et de demain*. [4ième Colloque International FOOTBALL & RECHERCHES]. Juin 2009.
- Rother, C. 2000.** A new approach for vanishing point detection in architectural environments. *BMVC 2000*. Bristol, GB : s.n., 2000. Vol. 11-14, pp. 382-391.
- Rowe, D. 2008.** Towards robust multiple-target tracking in unconstrained human-populated environments. *PhD thesis*. s.l., Spain : Universitat Autorna de Barcelona, 2008.
- Rowe, S. 1996.** Robust feature search for active tracking. *PhD thesis*. s.l. : University of Oxford, 1996.
- Shah, M. A. et Jain, R. 1984.** Detecting time-varying corners. *Computer Vision, Graphics and Image Processing*. 1984. Vol. 28, pp. 345-355.
- Shi, J. et Tomasi, C. 1994.** Good features to track. *IEEE Conference of Computer Vision and Pattern Recognition*. Seattle, WA : s.n., June 1994. pp. 593-600.
- Shufelt, J. A. 1999.** Performance evaluation and analysis of vanishing point detection techniques. *PAMI(21), n°3*. March 1999. pp. 282-288.
- Stauffer, C. et Grimson, W. E. L. 1999.** Adaptive background mixture models for real-time tracking. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 1999. Vol. 2, p. 252.
- Thomas, M. L. 1998.** A two-stage algorithm for model-based registration of medical images. *International Conference on Pattern Recognition*. 1998. Vol. 1, p. 344.
- Tomasi, C. et Kanade, T. 1991.** Detection and tracking of point features. *Technical Report CMU-CS-91-132*. Carnegie Mellon University, Pittsburgh, PA : s.n., April 1991.
- **1992.** Shape and motion from image streams under orthography : a factorization method. *Int. Journal of Computer Vision*. 1992. Vol. 9(2), pp. 137-157.
- Torr, P. H. S. et Zisserman, A. 1999.** Feature based methods for structure and motion estimation. *ICCV Workshop on Vision Algorithms*. 1999. pp. 278-294.
- Tsai, R. Y. 1986.** An efficient and accurate camera calibration technique for 3D machine vision. *Proceedings of IEEE Conference on COmputer Vision and Pattern Recognition*. Miami Beach, FL : s.n., 1986. pp. 364-374.
- **1987.** Metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*. Miami-Beach, FL : s.n., August 1987. Vol. 3, 34, pp. 364-374.
- Tuytelaars, T., et al. 1998.** The cascaded Hough transform as an aid in aerial image interpretation. *Proceedings of the 6th Internatioanl Conference on Computer Vision (ICCV'98)*. Bombay, India : s.n., January 1998. pp. 67-72.

- van den Heuvel, F. A. 1998.** Vanishing point detection for architectural photogrammetry. *International Archives of Photogrammetry and Remote Sensing*. 1998. Vol. XXXII part 5, pp. 652-659.
- Wang, L, Hu, W. et Tan, T. 2003.** Recent developments in human motion analysis. *Pattern Recognition*. 2003. Vol. 36, 3, pp. 585-601.
- Welch, G. et Bishop, G. 1995.** An introduction to the Kalman filter. *UNC-CH Computer Science Technical Report 95-041*. 1995. p. 14 pages.
- . **1997.** SCAAT : incremental tracking with incomplete information. *Computer Graphics*. [T. Whitted]. Los Angeles, CA, USA : ACM Press, Addison-Wesley, August 1997. pp. 333-344.
- Wren, C. R., et al. 1997.** Pfinder: real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1997. Vol. 19, 7, pp. 780-785.
- Zhang, Z. 2000.** A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2000. Vol. 22, 11, pp. 1330-1334.
- . **1996.** Epipolar geometry un stereo, motion and object recognition : a unified approach. *Kluxer Academic Publishers*. 1996. ISBN 0-7923-4199-6.
- Zitova, B. et Flusser, J. 2003.** Image registration methods: a survey. *Image and Vision Computing*. October 2003. Vol. 21, 11, pp. 977-1000.
- Zoudji, B. et Thon, B. 2003.** Expertise and implicit memory: differential repetition priming effects on decision making in experienced and a inexperienced soccer players. *International Journal of Sport Psychology*. 2003. Vol. 34, pp. 189-207.
- Zoudji, B., et al. 2005.** Mnemonic system efficiency of expert soccer players to temporal constraints in a simulated decision making task. [éd.] P. Terry, S. Gordon, S. Hanrahan, L. Ievleva, G. Kolt, and P. Tremayne T. Morris. *Promoting Health & Performance for Life*. [ISSP (CD Rom)]. Sydney : s.n., 2005.
- Zuniga, O. A. et Haralick, R. M. 1983.** Corner detection using the facet model. *Proc. Conf. Pattern Recognition and Image Processing*. 1983. pp. 30-37.