Localisation 3D basée sur une approche de suppléance multi-capteurs pour la Réalité Augmentée Mobile en Milieu Extérieur

Imane Zendjebil

▶ **To cite this version:**

Imane Zendjebil. Localisation 3D basée sur une approche de suppléance multi-capteurs pour la Réalité Augmentée Mobile en Milieu Extérieur. Interface homme-machine [cs.HC]. Université d'Evry-Val d'Essonne, 2010. Français. NNT : . tel-00541366

HAL Id: tel-00541366

https://theses.hal.science/tel-00541366

Submitted on 30 Nov 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

NNT : 2010 Evry 0024

UNIVERSITE D'EVRY-VAL D'ESSONNE

Laboratoire d'Informatique, Biologie Intégrative et Systèmes Complexes

T H E S E

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ D'ÉVRY

Spécialité : Sciences de l'Ingénieur

Localisation 3D basée sur une approche de suppléance multi-capteurs pour la Réalité Augmentée Mobile en Milieu Extérieur

présentée et soutenue publiquement par

Iman Mayssa ZENDJEBIL

Le 01 Octobre 2010

JURY

Mr. David FOFI	, Prof. Université de Bourgogne	, Rapporteur
Mr. Eric MARCHAND	, Prof. Université de Rennes 1	, Rapporteur
Mr. Pascal GUITTON	, Prof. Université de Bordeaux 1	, Examinateur
Mr. Fakhreddine ABABSA	, MdC Université d'Evry	, Encadrant
Mr. Jean-Yves DIDIER	, MdC Université d'Evry	, Co-encadrant
Mr. Malik MALLEM	, Prof. Université d'Evry	, Directeur de thèse

"le coeur le plus sur est le coeur d’une mère."

Table des matières

Table des matières i

Table des figures v

Liste des tableaux ix

1 Réalité Augmentée en Extérieur : un tour d'horizon 7
- 1.1 Réalité Augmentée : Définitions . . . 7
- 1.2 Système de réalité augmentée : descriptif et technologies . . . 9
  - 1.2.1 Base de connaissances . . . 10
  - 1.2.2 Capteurs de localisation . . . 12
    - 1.2.2.1 Global Positioning System (GPS) . . . 12
    - 1.2.2.2 Capteurs inertiels . . . 14
    - 1.2.2.3 La caméra . . . 16
  - 1.2.3 Dispositifs de restitution . . . 16
    - 1.2.3.1 Dispositifs basés moniteurs . . . 16
    - 1.2.3.2 Les casques de RV/RA . . . 18
- 1.3 Réalité augmentée en extérieur : applications . . . 19
  - 1.3.1 Applications pour la navigation . . . 19
  - 1.3.2 Applications pour l'accès à l'héritage culturel . . . 22
  - 1.3.3 Applications pour l'assistance au travail . . . 23
- 1.4 Synthèse . . . 26
- 1.5 RA en extérieur : Problématiques . . . 29
  - 1.5.1 Localisation . . . 29
  - 1.5.2 Visualisation . . . 30
  - 1.5.3 Interaction . . . 30
- 1.6 Objectifs de la thèse . . . 31
- 1.7 Conclusion . . . 32

2 Localisation basée vision 33
- 2.1 Taxonomie des méthodes d'estimation de pose . . . 33
  - 2.1.1 Approches avec connaissance *a priori* . . . 34
    - 2.1.1.1 Méthodes basées marqueurs . . . 34
    - 2.1.1.2 Méthodes sans marqueurs ou "markerless" . . . 35
  - 2.1.2 Approches sans connaissance *a priori* . . . 42
- 2.2 Discussion . . . 43
- 2.3 Méthode basée point d'intérêts : vue globale . . . 48

2.4 Initialisation semi-automatique : Appariement 2D/3D 50
2.5 Suivi basé points 53
2.6 Expérimentations et résultats 54
2.6.1 Performances de l'initialisation semi-automatique 54
2.6.2 Performances de l'estimation de pose 55
2.6.2.1 Erreur de reprojection 56
2.6.2.2 Erreur de localisation : Position 58
2.6.2.3 Erreur de localisation : Orientation 59
2.6.2.4 Temps d'exécution 61
2.6.2.5 Résultats de recalage 62
2.7 Conclusion 62

3 Systèmes de localisation multi-capteurs 65
3.1 Taxonomie des systèmes multi-capteurs 65
3.1.1 La fusion de données 66
3.1.1.1 You et al. 67
3.1.1.2 Hol et al. 68
3.1.1.3 Bleser et al. 69
3.1.1.4 Ababsa et al. 71
3.1.1.5 Reitmayr et Drummond 73
3.1.1.6 Schall et al. 75
3.1.2 La suppléance des données 77
3.1.2.1 Aron et al. 77
3.1.2.2 Maidi et al. 79
3.2 Synthèse et étude comparative 80
3.3 Proposition d'un système de localisation multi-capteurs 85
3.4 Enjeux et problématiques 86
3.4.1 Calibration du capteur hybride 87
3.4.2 Localisation et suivi basée vision 87
3.4.3 Prédiction d'erreur et correction 87
3.5 Conclusion 88

4 Modélisation et Calibration de Capteur Hybride 89
4.1 Calibration Inertiel/Caméra : état de l'art 90
4.1.1 You et al. 90
4.1.2 Alves et al. 91
4.1.3 Lang et Pinz 92
4.1.4 Hol et al. 92
4.1.5 Aron et al. 93
4.1.6 Reitmayr et Drummond 94
4.1.7 Maidi et al. 95
4.1.7.1 Première approche 95
4.1.7.2 Deuxième approche 96
4.1.8 Bleser et al. 97
4.2 Synthèse et étude comparative 98
4.3 Capteur Inertiel/Caméra : Modélisation et calibration 99
4.3.1 Centrale inertielle 100
4.3.2 Modélisation 101
4.3.3 Calibration 102
4.4 Capteur GPS/Caméra : modélisation et calibration 103

4.4.1 GPS . . . 103
4.4.2 Modélisation . . . 105
4.4.3 Calibration . . . 105
4.5 Expérimentations et résultats . . . 106
4.5.1 Caractérisation de la centrale inertielle . . . 107
4.5.2 Calibration Inertiel/Caméra : évaluation de la précision de l'estimation des rotations . . . 109
4.5.3 Caractérisation du récepteur GPS . . . 111
4.5.4 Calibration GPS/Caméra : évaluation de la précision de l'estimation de la position . . . 114
4.6 Conclusion . . . 116

5 Localisation basée suppléance multi-capteurs 119
5.1 Description du système . . . 120
5.2 Composants nécessaires à l'intégration des deux sous-systèmes . . . 121
5.2.1 Critères de validation . . . 121
5.2.1.1 Le nombre de points suivis . . . 122
5.2.1.2 L'erreur de reprojection . . . 122
5.2.1.3 Intervalles de confiance . . . 122
5.2.2 Prédiction et correction d'erreur . . . 123
5.2.2.1 La régression avec le processus Gaussien . . . 125
5.2.2.2 Application au système d'assistance de localisation . . . 126
5.2.3 Réinitialisation automatique . . . 127
5.3 Fonctionnement et réalisation du système . . . 129
5.3.1 ARCS : Augmented Reality Components System . . . 131
5.3.1.1 Composants . . . 131
5.3.1.2 Les feuilles (*sheets*) . . . 131
5.3.1.3 Automate . . . 132
5.4 Expérimentations et Résultats . . . 133
5.4.1 Evaluation des performances de l'approche de réinitialisation . . . 133
5.4.2 Apport de la prédiction/correction . . . 135
5.4.3 Evaluation du comportement du système de localisation . . . 137
5.4.3.1 Cas d'occultation partielle . . . 139
5.4.3.2 Cas d'occultation totale . . . 140
5.4.3.3 Cas de variations de luminosité . . . 140
5.4.3.4 Cas de mouvements brusques . . . 141
5.4.4 Bilan . . . 141
5.5 Application au projet RAXENV . . . 143
5.5.1 Matériel employé . . . 144
5.5.2 Architecture logicielle . . . 145
5.5.3 Description de la plate-forme RAXENV . . . 146
5.5.4 Résultats . . . 147
5.6 Conclusion . . . 150

Bibliographie 157

A Outils Mathématiques 165
A.1 Géométrie de la caméra . . . 165
A.1.1 Paramètres extrinsèques . . . 166
A.1.2 Paramètres intrinsèques . . . 166

A.2 Faugeras-Toscani : calibration de caméra . . . 167
A.3 Itération orthogonale . . . 168
A.4 Algorithme de RANSAC . . . 169
A.5 Homographie . . . 171

B Estimation de pose basée segments **173**
B.1 Vue globale . . . 173
B.2 Appariement 2D/3D basé segments . . . 175
B.3 Estimation de la pose basée segments . . . 175

C Estimation de pose basée crêtes de montagnes (contours) **177**
C.1 Extraction des crêtes de montagnes . . . 178
C.1.1 Segmentation basée HSV . . . 179
C.1.2 Filtrage . . . 180
C.1.3 Extraction des contours . . . 180
C.2 Résultats . . . 181

D GPS : fiabilité **183**

E Composants développés **185**

Table des figures

1.1 Le premier casque de réalité Augmentée [Sutherland, 1998] 8
1.2 Le Continuum de la réalité-virtualité [Milgram et Kishino, 1994] 8
1.3 Vue globale d'un système de Réalité Augmentée [Didier et al., 2009] 10
1.4 Exemple de données utilisées pour la reconstruction de modèles d'environnements exétrieur. 11
1.5 Exemple de reconstruction de ville avec CityEngine [Müller et al., 2006] 12
1.6 GPS : 24 Satellites placés dans l'orbite terrestre 13
1.7 Gyroscope mécanique 14
1.8 Schéma de principe d'un accéléromètre 15
1.9 Caméra : Configurations possible 16
1.10 Dispositif basé moniteur 17
1.11 Magnifying glass approach : Premier *Hand-held* [Rekimoto et Nagao, 1995] . . 17
1.12 Les dispositifs de visualisation Hand-held 17
1.13 Classification des casques HMD pour la RA selon [Azuma, 1997] 18
1.14 Exemples de casques HMD 19
1.15 La plate-forme **MARS** et son interface de navigation [Hollerer et al., 1999] . . 19
1.16 Le prototype **BARS** [Julier et al., 2000] 20
1.17 Les différentes versions de **Tinmith** [Piekarski et Thomas, 2001] 20
1.18 La plate forme **Studierstube** [Reitmayr et Schmalstieg, 2003] 21
1.19 Le système **ARCHEOGUIDE** [Gleue et Dähne, 2001] 22
1.20 Le système **Augurscope** [Schnadelbach et al., 2002] 23
1.21 La plate-forme **GEIST** [Holweg et Schneider, 2004] 23
1.22 Pour la visualisation dynamique des constructions 24
1.23 Le projet **Vidente** [Schall et al., 2007] 24
1.24 **IpCity** [Markus et Dieter, 2007] 25
1.25 **Timewarp** un scénario du projet **IpCity** 25
1.26 la plate-forme RAXENV : aspect matériel 26
1.27 Exemple de l'application Métro de Paris proposé par Apple pour les I-Phone . . 28
1.28 Système de Réalité Augmentée mobile : architecture 29

2.1 Exemple d'estimation de pose en utilisant **ARToolKit** [Kato et Billinghurst, 1999] 35
2.2 Exemple d'utilisation de marqueurs en extérieur [Piekarski et Thomas, 2002] . 35
2.3 Chaîne de traitement associée au calcul de pose 36
2.4 Etapes de l'approche utilisée dans [Reitmayr et Drummond, 2006] 36
2.5 Exemple d'augmentation d'un environnement extérieur [Comport et al., 2006] . 37
2.6 Méthode d'estimation de pose proposée dans [Lepetit et al., 2003] 38
2.7 Exemple du modèle utilisé et des images de références dans [Vacchetti et al., 2004] 39

2.8 Suivi avec d'images de référence [Stricker et Kettenbach, 2001] 40
2.9 Suivi planaire proposé dans [Simon et Berger, 2002] : (a) Définition du plan à suivre, extraction des points et suivi (b) Recalage obtenu 41
2.10 Estimation de pose basée suivi hybride [Vacchetti et al., 2004] 42
2.11 Suivi obtenu avec l'approche hybride décrite dans [Pressigout et Marchand, 2006] 42
2.12 Modèle de caméra sténopé 48
2.13 Principe général de fonctionnement en utilisant une approche basée points. . . . 49
2.14 (a) Rendu du modèle filaire (b) Alignement manuel du rendu avec la vue courante 51
2.15 ensemble de points extraits (en jaune) autour de la projection d'un point 3D (en rouge) 52
2.16 Résultat de l'appariement : correspondant 2D des points 3D visibles du modèle 52
2.17 Résultats d'appariement 2D/3D : (a) Projection des points 3D (en rouge) avec les coins extraits dans la zone de recherche (en jaune) (b) résultats de l'appariement avec le SURF (rouge) et résultats aprés élimination des données aberrantes (en jaune) 55
2.18 Tracé des erreurs de reprojection 56
2.19 Tracé des erreurs de généralisation 57
2.20 Positions caméra (ligne bleu) vs. positions de référence (ligne rouge) le long d'une droite 58
2.21 Positions caméra (en bleu) vs. positions de référence (en rouge) le long d'une droite 59
2.22 Erreurs d'orientation avec des rotations autour de l'axe X 60
2.23 Erreurs d'orientation avec des rotations autour de l'axe Y 60
2.24 Erreurs d'orientation avec des rotations autour de l'axe Z 61
2.25 Résultats de recalage d'un modèle filaire (rouge) sur la façade d'un bâtiment. . 63
2.26 Résultat de recalage d'un modèle 3D sur le château de Saumur 63

3.1 Descriptif du système multi-capteurs présenté dans [You et al., 1999] 67
3.2 Le modèle de prédiction de l'orientation [You et al., 1999] 67
3.3 Exemple d'annotations dans un environnement extérieur [You et al., 1999] . . . 68
3.4 Le modèle de fusion selon [Hol et al., 2006] 68
3.5 Exemple d'une scène avec les imagettes constituant le modèle 3D 69
3.6 Flot de données dans l'approche de localisation proposée dans [Bleser, 2009] . 70
3.7 Le système Hybride proposé dans [Ababsa et Mallem, 2007] : flot de données . 71
3.8 Un filtre complémentaire pour la fusion de capteurs [Ababsa et Mallem, 2007] . 72
3.9 Suivi de contours basé Inertiel/Vision [Reitmayr et Drummond, 2006] 73
3.10 Définition de la zone de recherche dans [Reitmayr et Drummond, 2007] 74
3.11 Système de RA proposé dans [Schall et al., 2009] 75
3.12 Architecture du système proposé dans [Schall et al., 2009] 76
3.13 Estimation de pose basée multi-capteur [Aron et al., 2007] 78
3.14 Descriptif du système de localisation [Maidi et al., 2009] 79
3.15 Résultas en situation de semi-occultation et occultation totale [Maidi et al., 2009] 80
3.16 Vue globale du système de localisation 86

4.1 Configuration des systèmes de coordonnées dans [You et al., 1999] 90
4.2 Inertiel/Camera observant la direction verticale [Alves et al., 2004] 91
4.3 Approche de calibration décrite par [Lang et Pinz, 2005] 92
4.4 L'approche de *Hol et al.* 93
4.5 Configuration utilisée dans [Aron et al., 2007] pour la calibration 94
4.6 Procédure de calibration avec un bras de robot [Maidi et al., 2005] 95

4.7 Capteur Hybride de [Maidi et al., 2009] et système de repères associé 96
4.8 Systèmes de coordonnées utilisées dans [Bleser, 2009] 97
4.9 Centrale Inertielle 100
4.10 Configuration des repères 101
4.11 Relation entre le repère R_G et le repère R_W 102
4.12 (a) repère défini par le système WGS84 (b) Coordonnées géographiques : longitude et latitude 104
4.13 (a) Projection conique (b) Projection conique conforme Lambert : cas de la France 104
4.14 Calcul de l'orientation entre les positions GPS et postion caméra 106
4.15 Les différentes configurations pour la précision des orientations de la centrale inertielle 107
4.16 Orientation centrale inertielle (rouge, vert et bleu) vs. orientation de référence (noir) 108
4.17 Variation du R_{GW} : en angle d'euler 109
4.18 orientation inertielle (en angle d'euler) (bleu) vs orientation caméra (rouge) . . 110
4.19 orientation inertielle avec correction (en angle d'Euler) (bleu) vs orientation caméra (rouge) 111
4.20 Comparaison entre le recalage obtenu avec les orientations de la centrale inertielle avec et sans correction 112
4.21 Positions exprimées en coordonnées géographique (à gauche) et cartographique (à droite) 113
4.22 Tracé de la trajectoire du GPS (vert) vs. caméra (magenta) vs. réelle (noir) sur une ligne droite 114
4.23 Positions GPS (vert) vs. caméra (magenta) vs. réelle (noir) 115
4.24 Résultat de recalage avec les positions GPS (vert) vs. la vision (rouge). 117

5.1 flux de données 121
5.2 Intervalle de confiance défini pour valider les positions estimées par la caméra. 123
5.3 (a) Les appariements des points SURF (b) Surpression des *outliers* avec le calcul de l'homographie 127
5.4 Réinitialisation automatique : schéma illustratif 128
5.5 (a) Image de référence (b) Projection des points avec l'homographie calculée . 129
5.6 fonctionnement du système hybride de localisation 130
5.7 Exemple de composition de composants pour la formation d'une feuille 131
5.8 Une application est la composition d'un automate et de plusieurs feuilles 132
5.9 Organisation d'un fichier XML pour la description d'une application avec ARCS 133
5.10 Descriptif du moteur d'exécution 133
5.11 Résultats de l'utilisation de la réinitialisation : Exemple de points coplanaires . 134
5.12 Résultats de l'utilisation de la réinitialisation : Exemple points non-coplanaires 135
5.13 Comparaison entre les erreurs prédites (en bleu) et les erreurs estimées (rouge) obtenue sur premier jeu de données 136
5.14 Comparaison entre les erreurs prédites (en bleu) et les erreurs estimées (rouge) obtenue sur un second jeu de données 137
5.15 Positions caméra (rouge) vs. positions GPS (bleu) vs. position GPS corrigée (noir) (obtenue sur le premier jeu de données) 138
5.16 Positions caméra (rouge) vs. positions GPS (bleu) vs. position GPS corrigée (noir) (obtenue sur le second jeu de données). 138
5.17 Résultats obtenus dans le cas d'occultation partielle 139
5.18 Résultats obtenus dans le cas d'occultation totale 140

5.19 Résultats obtenus dans le cas de variation de luminosité 141
5.20 Résultats obtenus dans un cas de mouvement brusque 142
5.21 L'ensemble de blocs composant le phidget 145
5.22 Aperçu de la plate-forme `Elkano` utilisée dans le projet RAXENV 146
5.23 Flot de données dans la plateforme RAXENV 146
5.24 Exemple de recalage sur le site du château de Saumur 147
5.25 Résultats de recalage avec différents degré de transparence 148
5.26 Exemple de manipulation 148
5.27 Exemple de visualisation de sondes de sondage 149
5.28 Exemple de coupe dans le sol 149
5.29 Exemple d'utilisation en mode sans recalage visuel 150

A.1 Modèle de projection perspective : modèle sténopé 165
A.2 Schéma représentant le principe de l'algorithme Itération orthogonale 169
A.3 (a) Ensemble de données (b) Ajustement de modèle de ligne avec détermination de ligne droite 170

B.1 Principe général de fonctionnement en utilisant une approche basée segment. . 174
B.2 Schéma descriptif de la phase d'appariement 2D/3D en utilisant les segments . 174
B.3 Schéma descriptif [Horaud et Monga, 1995] 175

C.1 Système multi-capteur pour la localisation en environnement panoramique . . . 177
C.2 Exemples de résultats d'extraction de contours avec le filtre de Deriche sur les images panormiques 178
C.3 Exemple de segmentation en région 179
C.4 Exemple d'application du filtrage pour l'élimination des régions parasites . . . 180
C.5 Exemple d'extraction des contours à partir des régions obtenues 181
C.6 Exemple d'extraction des contours à partir des régions obtenues 181

E.1 L'automate à états finis représentant le système de localisation 185
E.2 Les composants utilisés dans la feuille *initialisation* : chargement des paramètres, et initialisation des capteurs 186
E.3 Les composants utilisés dans la feuille *matching* : l'initialisation semi-automatique 187
E.4 Les composants utilisés dans la feuille *localization* représentant le fonctionnement du sous-système de vision 188
E.5 Les composants utilisés dans la feuille *Aidlocalization* représentant le fonctionnement du sous-système d'assistance à la localisation 189
E.6 Les composants utilisés dans la feuille *automatch* représentant la phase de réinitialisation 190

Liste des tableaux

1.1 Synthèse des plates-formes de réalité augmentée en milieu extérieur 27

2.1 Comparatif des méthodes sans marqueurs avec connaissance *a priori* 46
2.2 Paramètres de calibration de la uEye UI-2220R 54
2.3 Temps de calcul moyen par phase pour l'initialisation 55
2.4 Erreurs de reprojection 57
2.5 Erreurs généralisées 57
2.6 Localisation basée vision : Erreurs moyennes de position et écart types 59
2.7 Localisation basée vision : erreurs moyennes et écarts types des orientations . . 61
2.8 Temps de calcul par phase pour le calcul de pose 62

3.1 Systèmes Multi-capteurs basés fusion de données : Synthèse 82
3.2 Systèmes Multi-capteurs basés suppléance de données : Synthèse 83
3.3 Comparatif des performances des approches de localisation basées multi-capteurs 84

4.1 Comparatif des approches de Calibration Inertiel/Camera 98
4.2 Performances des orientations fournies par la centrale inertielle 108
4.3 Temps de calcul par phase pour le calcul de pose 115
4.4 Temps de calcul par phase pour le calcul de pose 116

5.1 Comparaison des temps d'exécution de la réinitialisation : avec et sans prédiction 136
5.2 Les erreurs de prédiction 137

Remerciements

Nous y voilà maintenant, après quatre années de dur labeur, devant le fait accompli. Il faut bien y passer un jour et clore ce chapitre de la vie. Si je voulais résumer cette période, je dirais que c'est une expérience pleine de bonheur, de stresse, de doute, de poussé d'adrénaline et d'angoisse. Mais heureusement que tout passe et qu'au final on ne garde que le meilleur. Avant de rentrer dans le vif du sujet, j'aimerais remercier tout les gens qui ont contribués à mon travail de prés ou de loin.

Tout d'abord, je souhaiterais remercier les membres du jury qui ont accepté de juger le fruit de mon travail. Je tiens à remercier **Pascan Guitton** pour m'avoir fait l'honneur de présider mon jury de thèse. Je remercie en particulier **David Fofi** et **Eric Marchand** pour avoir accepté de rapporter ma thèse et apporter une contribution critique à celle-ci ainsi que toutes les remarques positives adressées à la suite de leur lecture de mon rapport.

Mes remerciements vont à mon directeur de thèse **Malik Mallem** qui m'a accueilli au sein de son équipe et a dirigé mon travail de thèse avec son œil avisé et son expérience. Ma gratitude va à mes deux encadrants **Fakhreddine Ababsa** et **Jean-Yves Didier** qui ont veillés au bon déroulement de ma thèse. Je les remercie pour leurs conseilles, leurs disponibilité à tout moment leurs aide et leurs sens critique ainsi que leurs encouragements pour me surpasser.

Je n'oublie pas aussi de remercie les partenaires de l'aventure RAXENV **Jacques Vairon**, **Luc Frauciel**, **Pierre Thierry**, **Pascal Guitton**, **Bernard Rodière** et **Romuald Delmont**. Je remercie particulièrement **Joachim Poudreaux** avec qui j'ai eu à travailler à plusieurs reprises.

Bon je pense que j'ai cité tout le monde ...

Oups ... j'ai failli oublier mes acolytes. Ca été une belle expérience grâce à vous. Votre présence m'a permis de surmonter mes doutes, mes baisses de morales. **Mouna**, je te remercie pour ta présence, ton soutient et ton amitié. **Nader**, le RER D se souviendra de nos sujets de discussions et les débats lancés pour faire passer le temps durant le trajet. Je n'oublie pas **Pierre** et son jeu de l'œil, **Cris**, **Mahmoud** et **Christophe** qui ont animés avec nous la *Dream Team*. Je remercie aussi tout les membres du laboratoire.

Mes pensées et gratitudes vont à ma famille qui a toujours été là malgré la distance qui nous sépare pour m'encourager, me soutenir sans condition dans tout ce que j'entreprends.

Introduction générale

Par son principe à rehausser notre perception du monde réel avec des objets issus du monde numérique, la réalité augmentée (RA) permet d'envisager de plus en plus d'applications dans divers domaines. Ces applications peuvent constituer aussi bien un outil d'assistance (destiné à des experts issus de domaines tels que la médecine [State et al., 1996] ou la maintenance [Didier et al., 2005]) qu'un outil ludique et divertissant destiné au grand public.

Cependant, ce type d'applications a été longtemps restreint à des environnements intérieurs confinés. En effet, la majorité des applications qui ont été proposées étaient dédiées à des environnements à petite échelle. Ceci est dû d'une part à des problématiques de recalage et d'autre part aux performances des technologies utilisées qui limitaient le déploiement d'applications mobiles. Néanmoins, quelques travaux parus dans les années 90 avaient comme principal objectif de démontrer la faisabilité d'un système de réalité augmentée dans des environnements extérieurs tel que le projet MARS [Hollerer et al., 1999]. Cependant, depuis quelques temps, nous assistons à un engouement vers ce type d'application. Celui-ci est motivé, en premier lieu, par les avancées enregistrées du point de vue technologique sur les terminaux mobiles tels que les tablettes-PC, les PDAs et les téléphones cellulaires qui ont gagné en autonomie et en puissance de calcul. A cela s'ajoute le développement des capteurs qui sont plus précis et moins encombrants. En effet, certains d'entre eux sont maintenant intégrés dans des terminaux mobiles tels que les téléphones. Par exemple, nous retrouvons certaines marques de téléphone tel que le *iPhone* de *Apple* qui propose des applications ludiques employant la RA en utilisant un recalage primaire avec un récepteur GPS et des accéléromètres intégrés. En plus de l'aspect technologique, les avancées connues en termes de méthodologies principalement en méthodes de visualisation, d'interaction et de localisation ont permis d'envisager l'exportation de ces applications en milieu extérieur où l'utilisateur à peu de contrôle sur l'environnement.

S'inscrivant dans ce registre, notre thèse rentre dans le cadre d'un projet exploratoire, le projet RAXENV (Réalité Augmentée en eXtérieur appliquée au sciences de l'ENVironnement), qui propose une solution d'assistance à base de RA destinée à des métiers issus des sciences de la terre tel que la géologie. Ce projet a pour objectif de mettre au point un système mêlant des problématiques complémentaires de localisation, de visualisation et d'interaction sur terminaux mobiles. En effet, la localisation permet d'estimer la position et l'orientation du point de vue. Ces informations offre la possibilité de rajouter les données virtuelles en cohérence avec la vision du monde réel (recalage réel/virtuel) afin d'être visualisées par l'utilisateur. Ce dernier a la possibilité d'interagir avec ces objets virtuels. De ce fait, le processus de localisation représente le cœur d'un système de RA.

Dans le cadre de cette thèse, nous souhaitons étudier et proposer un système de localisation fonctionnant dans un environnement à grande échelle et en milieu extérieur dans un cadre de mobilité. Notre travail s'oriente vers une approche combinant plusieurs capteurs afin de permettre une localisation en continu et en temps réel sous différentes conditions de travail. La mise en œuvre de cette combinaison doit répondre à certaines exigences à savoir :

- **La précision de la localisation** : L'information de position et d'orientation doit être le plus précise possible pour avoir un bon recalage. Or, les données fournies par les différents capteurs utilisés ont des degrés de précision différents. De ce fait, il faut quantifier ces erreurs afin d'améliorer la précision globale du système ;
- **La robustesse et la réactivité du système de localisation** : Le système doit être robuste face aux conditions de travail. Il doit être capable de s'adapter aux différentes situations auxquelles il est confronté comme par exemple la défaillance d'un des capteurs et son incapacité à fournir une mesure cohérente. En effet, le système doit être réactif aux conditions extérieures pour ainsi pouvoir s'adapter à différentes situations et garantir un bon fonctionnement.

Dans nos travaux, nous nous focalisons sur les problématiques entourant le processus de localisation. Celles-ci concernent essentiellement les systèmes multi-capteurs. Elles constituent plusieurs verrous scientifiques et technologiques importants à lever. Ces problématiques seront présentées au fur à mesure en exposant les solutions que nous préconisons. Notre manuscrit comprend cinq chapitres.

Dans le premier chapitre (**Réalité augmentée en extérieur : un tour d'horizon**), nous présentons une déclinaison du paradigme de la réalité augmentée destiné à des systèmes évoluant dans des environnements extérieurs. La première partie du chapitre commence par définir la réalité augmentée telle qu'elle est présentée dans la communauté. Par la suite, nous exposons un aperçu général d'un système de réalité augmentée. Cette description concerne l'architecture des systèmes de RA. Nous nous préoccupons essentiellement des aspects technologiques composants ce type de systèmes. Après la description des travaux réalisés en réalité augmentée en extérieur, une synthèse permettra de voir l'évolution des applications sur différents points (technologies, méthodologiques et applicatives). Cette étude permettra de définir les problématiques qui entourent la mise en oeuvre d'une application de réalité augmentée complète en se focalisant sur leur déploiement en milieu extérieur. Enfin, nous allons présenter les objectifs poursuivis par nos travaux de thèse et qui seront traités tout au long du manuscrit. L'étude présenté dans ce chapitre a fait l'objet de publications dans une conférence [Zendjebil et al., 2008a] et une revue [Zendjebil et al., 2009].

Dans le second chapitre (**Localisation basée vision**), nous nous intéressons aux approches utilisant la vision pour calculer la position et l'orientation de l'utilisateur dans l'environnement. Ce chapitre comporte une brève étude des approches les plus utilisées. Nous nous restreignons aux approches utilisées en RA (en raison de la multiplication de travaux dans la vision par ordinateur en général). Cette étude a pour objectif de mettre en avant et confronter les approches les plus intéressantes pour des applications en extérieur. La seconde partie de ce chapitre présente l'approche que nous avons mise au point. Basée modèle, la méthode décrite utilise les points d'intérêts pour estimer la pose de la caméra. Elle se décompose en deux phases. La première phase représente la phase d'initialisation. Cette étape semi-automatique, qui nécessite peu d'intervention de l'utilisateur, cherche les appariements de points 2D/3D utiles pour le calcul de la pose. La seconde étape est constituée d'un suivi visuel qui aide à maintenir l'appariement 2D/3D obtenu. Des ex-

périmentations et des résultats accompagnent cette description. Ces expérimentations qui caractérisent les performances de l'approche que nous proposons sont obtenues sur des données réelles.

Dans le chapitre 3 (**Systèmes de localisation multi-capteurs pour la réalité augmentée mobile**), nous détaillons quelques systèmes de localisation proposés dans la communauté RA qui combinent plusieurs types de capteurs. Nous proposons une taxonomie qui se base sur la stratégie adoptée pour la combinaison des données issues des capteurs à savoir la fusion et la suppléance. La description de ces systèmes est suivie d'une étude comparative entre ces derniers. Enfin, nous esquissons notre système multi-capteurs ainsi que les différents problématiques et enjeux à relever lors de sa mise en œuvre.

Après avoir présenté brièvement le système de localisation ainsi que les enjeux à relever, le chapitre 4 (**Modélisation et calibration de capteur hybride**) comporte la description de la solution proposée pour la problématique liée à la calibration du capteur hybride. Dans ce chapitre, nous proposons deux procédures de calibration qui concernent respectivement le couplage Inertiel/Caméra et le couplage GPS/Caméra. Cette description comprend une modélisation de chaque couplage avec l'approche de calibration associée. Nous dressons un bref état de l'art sur les approches déjà proposées concernant la calibration Inertiel/Caméra. Par la suite, nous nous intéresserons aux parties expérimentales qui ont pour but de caractériser notre capteur hybride ainsi que les procédures de calibration. Les travaux décrits dans ce chapitre ont été présentés dans deux conférences [Zendjebil et al., 2008c] et [Zendjebil et al., 2008d] et paraitra prochainement dans [Zendjebil et al., 2010].

Dans le chapitre 5 (**Localisation basée suppléance multi-capteurs**), nous détaillons notre système de localisation. Cette description reprend l'architecture proposée pour notre système de localisation en y ajoutant les éléments connexes nécessaires à la mise en œuvre. Une fois toutes les parties de notre système présentées, nous allons nous intéresser au fonctionnement de ce système de localisation. La dernière partie concerne une batterie d'expérimentations pour tester le fonctionnement de notre système dans différentes situations réelles auxquelles il peut être confronté. Comme nous l'avons dit auparavant, notre travail de thèse rentre dans le cadre d'un projet ANR appelé RAXENV, acronyme de Réalité Augmentée en eXtérieur appliquée aux sciences de l'ENVironnement. Dans cette partie du chapitre, nous allons nous intéresser à ce qui a été effectué dans le cadre de ce projet. Afin de montrer l'application de nos travaux dans un cadre concret. Nous commençons par une description du projet, de ses objectifs et ses enjeux. Cette description sera suivie d'une présentation du matériel employé dans la conception de cette plate-forme. A cela, s'ajoute une brève exposition sur l'architecture logicielle utilisée. Ceci concerne essentiellement les outils utilisés pour le développement des différentes bibliothèques comprenant les différents traitements associés à la plate-forme. Nous présentons aussi quelques résultats obtenus dans le cadre du projet. Notre système de localisation a été présenté dans [Zendjebil et al., 2008b] et dans [Zendjebil et al., 2008d].

Enfin, nous terminons ce manuscrit par une synthèse sur les différents aspects traités avec les résultats obtenus.

Chapitre 1

Réalité Augmentée en Extérieur : un tour d'horizon

Depuis son apparition, la réalité augmentée (RA) ne cesse de séduire les utilisateurs en offrant de plus en plus d'applications et en repoussant les limites de l'imaginable. Longtemps restreinte à des environnements intérieurs, confinés et contrôlés, nous assistons depuis quelques années à une multiplication de travaux qui tentent d'exporter cette technologie dans des milieux extérieurs à grande échelle où l'être humain a un contrôle limité sur son environnement.

Ce chapitre présente un tour d'horizon sur les systèmes de réalité augmentée en extérieur (RAE). Il traite également des problématiques sous-jacentes permettant de cerner le contexte de cette thèse.

Dans un premier temps, nous allons passer en revue quelques définitions de la réalité augmentée. Dans un second temps, nous nous intéressons aux systèmes de réalité augmentée où nous décrirons ce qu'est un système de RA. Nous verrons aussi quelques dispositifs technologiques utilisés dans les systèmes de réalité augmentée particulièrement en fonctionnement extérieur. Puis, nous décrirons les systèmes de réalité augmentée en extérieur. Suivra une synthèse mettant en avant l'évolution des systèmes de RA en extérieur. Enfin, nous présentons les problématiques entourant la mise en œuvre de tels systèmes et nous exposons les objectifs fixés dans le cadre de cette thèse.

1.1 Réalité Augmentée : Définitions

Vers 1968, les travaux de [Sutherland, 1998] sur les casques de réalité virtuelle ont permis de mettre au point un casque semi-transparent "*see-through system*" (cf. fig.1.1) qui offrait la possibilité d'afficher de simples polygones. Utilisé pour la réalité virtuelle, ce nouveau concept permettait de pallier les inconvénients de la réalité virtuelle qui, d'une part, coupe son utilisateur de son monde réel et, d'autre part, rend difficile la reproduction réaliste du monde réel. Cette faculté de mélanger du réel et du virtuel a fait émerger un nouveau concept appelée **réalité augmentée** (en anglais "*Augmented Reality*"). En parcourant la littérature, nous avons constaté que diverses définitions ont été données à ce nouveau paradigme.

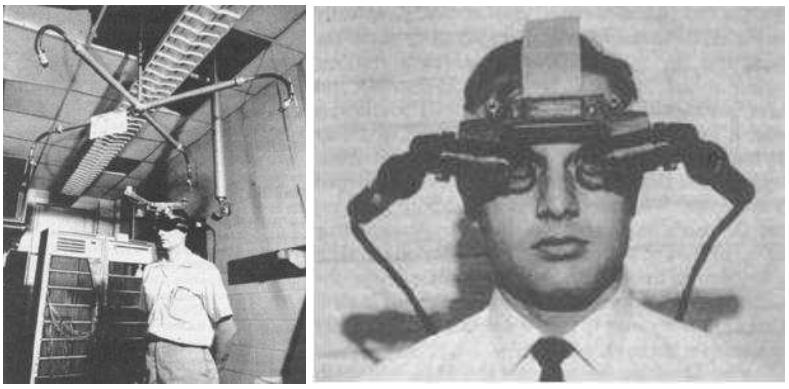


FIG. 1.1: Le premier casque de réalité Augmentée [Sutherland, 1998]

Dans [Milgram et Kishino, 1994], les auteurs introduisent la notion de réalité Mixée (*Mixed Reality*) regroupant l'ensemble des approches qui combinent, à différent degré, l'environnement virtuel et l'environnement réel. Milgram décrit un espace, appelé **Continuum de la réalité-virtualité** (cf. fig.1.2), dont les extrémités correspondent à la réalité et à la virtualité pures. Entre ces deux extrémités, il place deux grandes approches, à savoir :

1. **La virtualité augmentée** : elle consiste à intégrer des éléments réels extérieurs à la réalité virtuelle dans l'interface de simulation 3D, afin d'enrichir l'interaction de l'opérateur humain avec le monde virtuel au moyen d'outils et de tâches issues du monde réel.
2. **La réalité augmentée** : par opposition à la virtualité augmentée, elle consiste à incruster du virtuel dans un environnement réel dans le but d'enrichir l'interaction de l'utilisateur avec le monde physique grâce à des données et services offerts par le monde numérique (l'ordinateur).

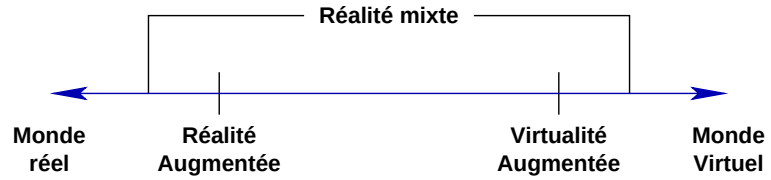


FIG. 1.2: Le Continuum de la réalité-virtualité [Milgram et Kishino, 1994]

Pour [Azuma, 1997], la réalité augmentée est caractéristique du mixage de la réalité virtuelle avec des éléments du monde réel et dont les traits communs sont :

1. la combinaison du réel et du virtuel ;
2. l'interaction en temps réel ;
3. l'alignement 3D (recalage).

Pour [Vallino, 1998], la réalité augmentée est un domaine émergeant de la réalité virtuelle, conséquence de la difficulté à recréer complètement l'environnement qui nous entoure. Les systèmes de réalité augmentée sont des systèmes qui offrent à l'utilisateur une vue d'une scène réelle rehaussée avec des objets virtuels générés par ordinateur. Ces informations additionnelles ont pour but d'enrichir la perception du monde.

Dans [Klinker, 1999], l'auteur définit la réalité augmentée comme étant la technologie par laquelle la vue du monde réel est augmentée avec des informations synthétisées par ordinateur.

Selon [Bérard, 1999], la réalité augmentée vise à réunir le meilleur des deux mondes, à savoir le monde réel et le monde virtuel. On fait appel au monde virtuel afin de réaliser des tâches qui ne peuvent être faites dans le réel mais sans pour autant exporter tout le monde réel dans le virtuel. Il suffit juste d'ajouter ces tâches à notre propre vision du monde.

Une autre définition avancée par [Fuchs et Moreau, 2003], présente la RA comme étant la technologie qui regroupe l'ensemble des techniques permettant d'associer le monde réel avec un monde virtuel. Elle utilise l'intégration d'images réelles (IR) avec des entités virtuelles (EV) : images de synthèses, objets virtuels, textes, symboles, graphiques, etc. Toutefois, ces enrichissements ne se restreignent pas qu'aux augmentations visuelles, mais peuvent être une augmentation d'un ou de plusieurs des cinq sens de l'être humain (la vue, le toucher, l'odorat, etc.).

Toutes ces définitions se rejoignent sur le principe que la réalité augmentée offre la capacité de rehausser et d'instrumenter notre perception du monde réel par l'ajout d'entités virtuelles. L'aspect visuel prédomine dans la plupart des cas car c'est le sens que nous mettons le plus à contribution dans la perception de notre environnement.

Durant ces dernières années, le paradigme de la réalité augmentée n'a cessé d'évoluer et ses champs d'applications de se diversifier notamment en milieu **extérieur**.

En effet, de nouveaux types de terminaux ont fait leurs apparitions telles que les tablettes-PC, PDAs, téléphones cellulaires, etc. Leur évolution en terme de capacités de calcul ont permis d'explorer la synergie entre la RA et l'informatique mobile en tirant parti de chacune d'elle. D'une part, la réalité augmentée, par son principe, offre la possibilité de rompre les frontières entre le monde réel et le monde numérique. D'autre part, l'informatique mobile offre à l'utilisateur une liberté de mouvement supplémentaire. De plus, elle permet de migrer de l'espace de travail traditionnel (bureau) vers d'autres environnements (en extérieur, par exemple) où l'accès aux données distantes est davantage contraint. De plus, les capteurs connaissent un développement en termes de précision et de miniaturisation. Cet essor a rendu possible l'exportation de la RA, longtemps confinée dans des environnements dit d'intérieur (*indoor*), préparés et contrôlables, vers des environnements à grandes échelles en extérieur (*outdoor*), non contrôlables et non préparés. Nous présentons dans la section 1.3 (page 19) les principaux travaux et projets menés dans ce sens.

1.2 Système de réalité augmentée : descriptif et technologies

Selon [Didier et al., 2009], un système de réalité augmentée comprend trois composantes essentielles (cf. fig 1.3) :

- une base de connaissances ;
- des capteurs ;
- et des dispositifs de restitutions.

La base de connaissances fournit une connaissance *a priori* sur l'environnement dans lequel évolue l'utilisateur. Ces connaissances se présentent sous forme de modèles CAO, re-

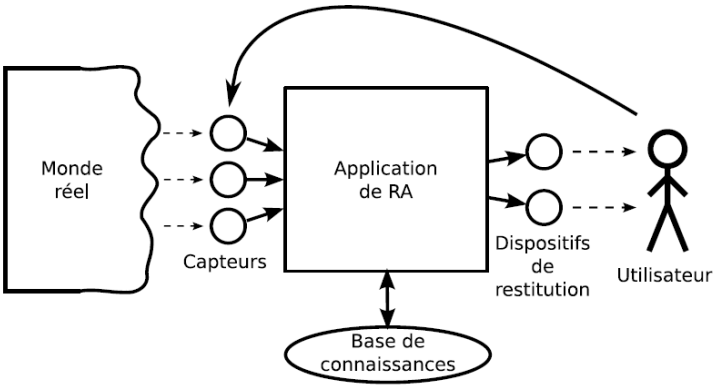


FIG. 1.3: Vue globale d'un système de Réalité Augmentée [Didier et al., 2009]

présentant une reproduction complète ou partielle de l'environnement, ou bien des systèmes d'informations géographiques (SIG) regroupant un ensemble de données géo-référencées tel que des plans, les réseaux d'assainissements, etc.

Les bases de données ne sont pas l'unique moyen de fournir une information sur l'environnement. En effet, un ensemble de capteurs peut être utilisé pour fournir une connaissance complémentaire de l'environnement de travail. Il s'agit généralement de capteurs de localisation qui permettent de connaître la position et/ou l'orientation de l'utilisateur dans l'environnement. Nous verrons par la suite que cette information est capitale pour les systèmes de RA.

A partir de ce que fournit la base de connaissance ainsi que les capteurs, l'application RA crée une vue augmentée. Cette dernière est retournée aux dispositifs de restitution appelés dispositifs de visualisation.

Dans ce qui suit, nous allons nous intéresser aux technologies utilisées dans les systèmes de RA en général et les applications en extérieur en particulier. Ceci concerne les approches utilisées pour la génération de base de connaissance, les capteurs et les dispositifs de restitution.

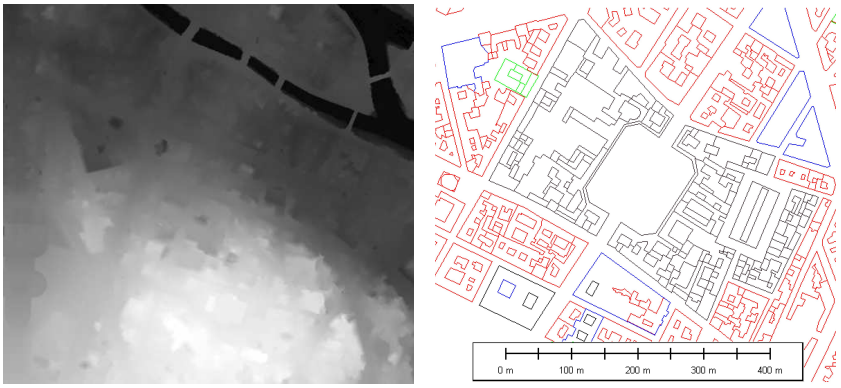
1.2.1 Base de connaissances

Certaines applications de RA ont besoin d'une représentation virtuelle 3D de l'environnement. Contrairement aux applications de réalité virtuelle ou l'immersion permet d'utiliser un environnement non-réaliste, les environnements utilisés en RA doivent être en adéquation avec l'environnement réel dans lequel l'utilisateur évolue.

Les modèles 3D sont très utiles pour l'estimation de la pose (position et orientation) à partir d'une mise en correspondance 2D/3D (entre les données image 2D et les données du modèle 3D). De plus, ces représentations servent comme des données virtuelles pour le recalage. La construction de modèles 3D simples et riches en information à grandes échelles est un enjeu majeur pour la RA mobile. En extérieur, la construction de ces modèles s'appuie sur des données "réelles" (mesurées) qui peuvent être :

1. **Photographie aérienne** : désigne des images acquises depuis le ciel (avion) ou l'espace (satellite).

2. **Modèle Numérique de Surface (MNS)** : est une représentation numérique de la topographie de l'environnement, c'est-à-dire des formes et détails visibles sur le terrain, qu'ils soient naturels, notamment le relief, ou artificiels (comme les bâtiments, les routes, etc.).
3. **Modèle Numérique de Terrain (MNT)** : est une représentation de la topographie naturelle (cf. fig.1.4-a). Il peut être défini comme un MNS sans sursol (construction, végétation, etc.).
4. **Données vectorielles** : peuvent être des emprises de bâtiments qui représentent la trace au sol (ou à une hauteur donnée) du bâtiment ou bien des éléments de faîtage qui forment une représentation de la géométrie du toit des bâtiments (cf. fig.1.4-b).
5. **Photos obliques** : à l'inverse des images ortho photo qui sont verticales, prises à altitude équivalente, ces images contiennent des informations très intéressantes pour la reconstruction notamment des façades.



(a) MNT (b) Données vectorielles

FIG. 1.4: Exemple de données utilisées pour la reconstruction de modèles d'environnements exétrieur.

Ce type de données est obtenu à partir d'une observation aérienne de l'environnement. Ces données ont été très utilisées vu qu'elles couvrent une large superficie. Toutefois, de nouveaux types de données plus précises et plus localisées sont utilisés. Ces nouvelles données sont obtenues par scanners, longuement utilisé pour la numérisation de petits objets, ou bien photos et vidéos terrestres.

Nous retrouvons sur le marché quelques logiciels utilisés pour la reconstruction des modèles mais qui ne sont pas adaptés à la reconstruction à grande échelle. Pour la reconstruction automatique de modèle à grande échelle, nous retrouvons l'approche décrite dans [Lafarge et al., 2005] et le laboratoire **MATIS** de l'IGN qui propose des méthodes pour reconstruire le modèle 3D à partir de données aériennes (photo, MNS, MNT). D'autres techniques se focalisent sur la reconstruction à partir des emprises de bâtiments. Ainsi dans [Müller et al., 2006], les auteurs proposent un système procédural nommé **CityEngine** basé sur une grammaire permettant une reconstruction paramétrable très détaillée d'environnements urbains de grande taille. Nous retrouvons dans la figure 1.5 un exemple de reconstruction d'une ville avec le système **CityEngine**. Le système **FastBuilder**, mis en place par la société **Archivideo**, reconstruit à partir de données géographiques des environnements urbains de grande taille. Il utilise des heuristiques afin de combler les blancs dans les données fournies. La reconstruction proprement dite utilise une bibliothèque de façades

adaptées à la ville (en moyenne 500 façades) et une classification des bâtiments selon leur style pour adapter le modèle. Les règles de reconstruction peuvent être modifiées à l'échelle de la ville, du quartier voire du bâtiment. Ce système permet une reconstruction rapide (quelques jours de calcul pour une ville telle que Paris avec 150000 bâtiments) et adaptable en fonction des besoins. Dans le cadre d'une navigation au niveau du sol, l'environnement comporte de nombreuses parties non visibles qu'il n'est donc pas nécessaire de reconstruire. Les modèles les plus utilisés dans les applications de réalité augmentée en extérieur sont les modèles CAO ainsi que les systèmes d'information géographiques (SIG). Le SIG est un système d'information qui englobe, gère et organise des données géo-référencées tel que des plans.

FIG. 1.5: Exemple de reconstruction de ville avec CityEngine [Müller et al., 2006]

Pour plus de détails, les lecteurs peuvent se référer à [Vairon et al., 2007a] où un état de l'art détaillé est présenté dans le chapitre 4 sur les approches de génération des environnements 3D à grandes échelles.

1.2.2 Capteurs de localisation

Pour les applications de réalité augmentée, l'information de localisation consiste à identifier la position et/ou l'orientation du point de vue. Ces données sont importantes pour connaitre la portion du monde perçu par l'utilisateur. Ceci permet essentiellement d'assurer par la suite une cohérence du mixage du réel et du virtuel.

Différents types de capteurs sont utilisés par la communauté de RA pour la localisation. Le calcul de la position et/ou de l'orientation d'un objet donné est effectué par rapport à un référentiel. En général, les capteurs peuvent avoir deux configurations possibles :

- Configuration dite **Outside-In** : où le récepteur est statique alors que l'émetteur est positionné sur la cible qui est en mouvement.
- Configuration dite **Inside-Out** : dans ce cas de figure, l'émetteur est statique et le récepteur est en mouvement.

Dans ce qui suit, nous allons présenter une taxonomie non exhaustive de ces capteurs, elle est basée sur les travaux présentés dans [Rolland et al., 2000] et [Auer, 2000].

1.2.2.1 Global Positioning System (GPS)

Plus connu sous le sigle GPS, ce système, conçu par le département américain de la Défense, permet de déterminer une position globale. Il se base sur un positionnement par satellites. Le GPS utilise le système géodésique WGS84, auquel se réfèrent les coordonnées

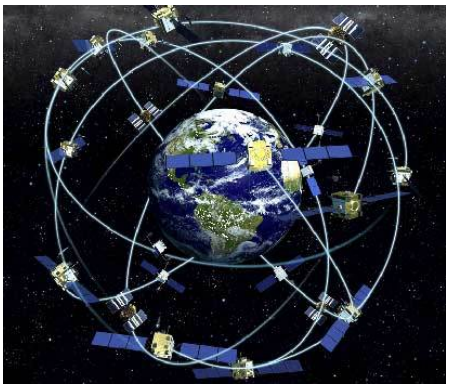


FIG. 1.6: GPS : 24 Satellites placés dans l'orbite terrestre

mesurées. A titre indicatif, le système est composé de 24 satellites placés en orbite terrestre (à une altitude de 20200 Km) et tournant autour de la terre en 24 heures. Six plans orbitaux sont définis, chacun disposant de 4 satellites. Chaque satellite est incliné de 55 degrés par rapport au plan équatorial. Cette configuration permet en moyenne d'avoir 5 à 8 satellites visibles à partir de n'importe quel point sur la terre. La figure 1.6 illustre la disposition de ces satellites autour de l'orbite terrestre.

Un récepteur GPS estime la position, le temps et la vitesse de l'opérateur à partir des différents signaux provenant des satellites. En effet, chaque satellite dispose d'une horloge atomique interne ainsi que des éphémérides * permettant le calcul des coordonnées prédites. Les satellites émettent continuellement un signal horodaté. La position est déduite à partir de la différence de temps entre les instants d'émission et de réception du signal.

En théorie, la position de l'utilisateur est déterminée à partir d'au moins 3 satellites visibles. Or, comme l'horloge du récepteur n'est pas précise et possède un décalage inconnu en plus des dérives des horloges atomiques des satellites, il faut au moins 4 satellites afin de déterminer la position de l'antenne réceptrice ainsi que le décalage de l'horloge. Ainsi, un récepteur GPS qui capte les signaux d'au moins quatre satellites peut, en mesurant les écarts relatifs des horloges, connaître sa distance par rapport aux satellites et, par trilatération †, de situer précisément en trois dimensions n'importe quel point courant sur la surface de la terre visible par les satellites.

La précision de ces systèmes est de l'ordre de 10 mètres pour les GPS grand public, et peut atteindre le mètre pour les GPS professionnels. Par ailleurs, il existe une variété de GPS qui possède une précision de l'ordre du centimètre. Il s'agit des systèmes DGPS pour *Differential* GPS. En plus des satellites, ce système s'appuie aussi sur des signaux émis

*Dans le langage courant, une éphéméride désigne ce qui se passe quotidiennement ; l'éphéméride du jour est la liste des évènements marquants de ce jour. Par extension, les éphémérides astronomiques désignent a priori une table journalière de positions de corps célestes mobiles (ceux du système solaire) ainsi que des phénomènes astronomiques ayant lieu ce jour telles les éclipses. Les éphémérides de positions sont donc avant tout la représentation d'un mouvement. Les éphémérides sous forme de tables de nombres sont les plus courantes et les plus anciennes, mais ce n'est pas la seule forme possible et, de nos jours, ce n'est plus la meilleure car il en existe maintenant d'autres beaucoup plus performantes. Wikipédia

†La trilatération est une méthode mathématique permettant de déterminer la position relative d'un point en utilisant la géométrie des triangles tout comme la triangulation. Mais contrairement à cette dernière, qui utilise les angles et les distances pour positionner un point, la trilatération utilise les distances entre un minimum de deux points de références. Wikipédia

à partir de stations terrestres dont les positions sont connues. Ces stations calculent séparément l'erreur de mesure pour chaque satellite et transmettent les données de correction. Les récepteurs DGPS fusionnent ces données avec leurs propres mesures pour fournir une position globale de la cible.

Généralement, la localisation en altitude est moins précise que la localisation en latitude et en longitude. De plus, différentes causes peuvent altérer le signal GPS. Pour plus d'information, ces causes sont décrites en annexe D (page 183).

1.2.2.2 Capteurs inertiels

Les centrales inertielles ou à inertie sont des systèmes de navigation comprenant des gyroscopes, des accéléromètres et des magnétomètres. Ils calculent, en temps réel, l'évolution des vecteurs vitesses et son attitude (roulis, tangage et lacet). Les capteurs inertiels fonctionnent selon le principe de conservation de l'axe de rotation. Ils différent des autres capteurs puisqu'ils ne sont pas composés d'une paire d'émetteur-récepteur. Voyons plus en détail les composants du capteur inertiel.

Les gyroscopes mécaniques : Ce sont des systèmes qui se basent sur le principe de conservation des moments angulaires. Ce principe suppose qu'un objet ayant une rotation avec une grande vitesse angulaire, en l'absence de moments externes, conserve ses moments angulaires. Équipé d'une roue, l'orientation de la cible est calculée à partir des angles reportés sur les encodeurs rationnels (cf. fig.1.7). Chaque gyroscope fournit un axe de référence. Au minimum deux gyroscopes suffisent pour recouvrer l'orientation d'un objet dans l'espace.

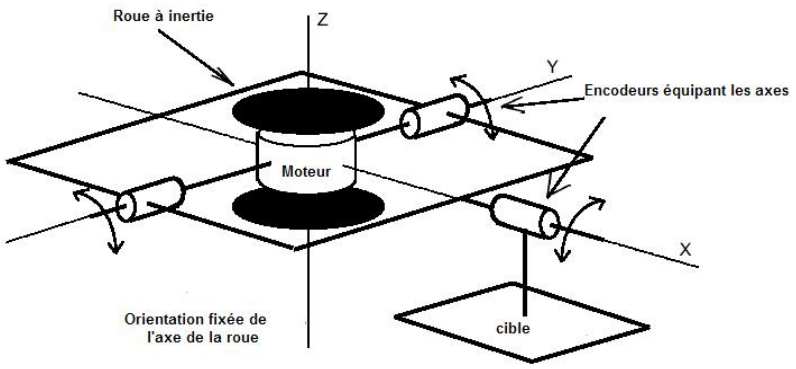


FIG. 1.7: Gyroscope mécanique

L'avantage principal de ce type de capteurs réside dans le fait qu'il n'a pas besoin d'un repère externe. En effet, les axes de rotation de la roue représentent le repère de référence. Les gyroscopes fournissent les vitesses angulaires. Les orientations peuvent être déduites à partir de ces données en effectuant une intégration. Toutefois, ces capteurs ont un problème : les moments angulaires de la roue ne sont pas toujours parallèles aux axes de rotation à cause des frictions minimes entre les axes de la roue et le roulement. De plus, le système présente une accumulation des erreurs.

Les accéléromètres : Ils permettent de mesurer les accélérations linéaires de l'objet auquel ils sont reliés selon la loi fondamentale de la dynamique. On distingue deux grandes

familles d'accéléromètres : les accéléromètres non asservis et les accéléromètres à asservissement.

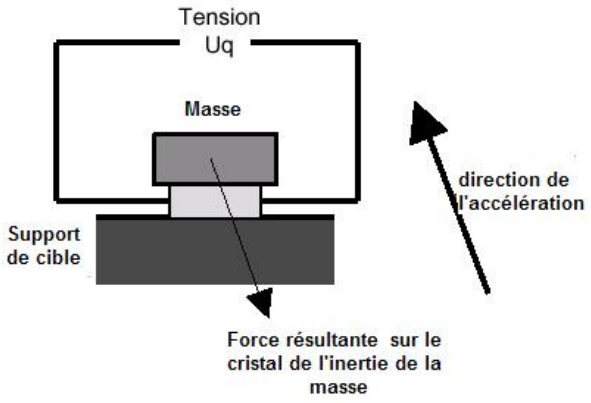


FIG. 1.8: Schéma de principe d'un accéléromètre

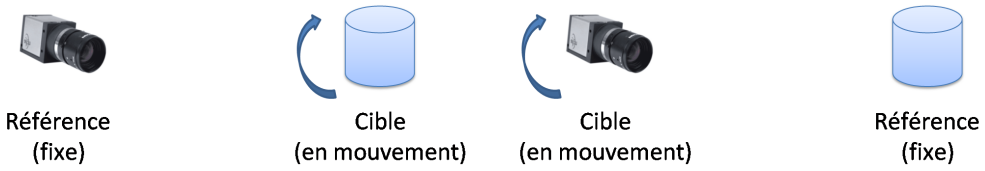
Brièvement, pour les capteurs de type non asservis (boucle ouverte), l'accélération est mesurée à partir de son image "directe", alors que pour les accéléromètres à asservissement, l'accélération est mesurée à la sortie d'une boucle de contre-réaction (asservissement) comportant un correcteur de type P.I. La figure 1.8 présente un schéma de principe d'un accéléromètre. La position peut être estimée à partir de ces données en effectuant une double intégration. La localisation est relative. Ceci implique la connaissance *a priori* d'une position de référence. Ce type de capteur est rapide, cependant le processus d'intégration tend à accumuler les erreurs au cours du temps ce qui produit une dérive (*drift*) entre la position estimée et la position réelle.

Les magnétomètres : C'est un capteur qui permet de mesurer le champ magnétique ambiant. Il est composé d'un liquide contenant des ions, placé dans une bobine. Ces ions oscillent sous l'effet du champ magnétique ambiant. En injectant un courant électrique intense dans la bobine ; un champ magnétique supérieur au champ qu'on veut mesurer est créé. Ceci force les ions à s'orienter selon ce nouveau champ. Une fois le courant coupé brusquement, les ions reviennent à leur place initiale en oscillant autour de cette position. Cette oscillation crée à son tour un courant dans la bobine dont la période est fonction du champ magnétique mesuré. Le magnétomètre peut s'utiliser pour mesurer le champ émis d'une base fixe ou le champ magnétique terrestre. Partant de là, il est alors possible de faire agir le capteur comme une boussole afin d'indiquer le nord magnétique. Le capteur fournit alors son orientation par rapport au Nord et, comme il ne nécessite pas de base magnétique, a un rayon d'action à l'échelle de la planète, ce qui en fait un capteur privilégié pour opérer en extérieur. L'inconvénient de ce type de système reste toutefois sa sensibilité aux perturbations électromagnétiques qui peuvent induire des erreurs angulaires dans les mesures de l'orientation.

Les capteurs inertiels mécaniques traditionnels demeurent encombrants. L'avènement des nanotechnologies a permis de miniaturiser ces dispositifs appelés MEMS (Micro-Electro-Mechanical Systems). La méthode consiste à intégrer des éléments mécaniques, capteurs, actionneurs et leur électronique sur un substrat commun en silicium par le biais des tech-

nologies de micro-fabrication. L'électronique est fabriquée en utilisant les méthodes classiques de production des circuits intégrés tandis que les éléments micromécaniques sont obtenus par le biais de processus compatibles qui vont découper et retirer des parties du silicium pour constituer les micro-mécanismes. Dans le cas des accéléromètres, ceci a permis de réduire les coûts de fabrication par 10, tout en miniaturisant et en augmentant la fiabilité de ces derniers.

1.2.2.3 La caméra



(a) Suivi Outside-in (b) Suivi Inside-out

FIG. 1.9: Caméra : Configurations possible

La caméra est le capteur le plus utilisé surtout dans les applications de RA et de robotiques mobiles. Le principe de fonctionnement de ce capteur consiste à analyser les projections 2D des caractéristiques images afin de déterminer la position et l'orientation de la cible. Plusieurs caméras peuvent être utilisées en même temps comme en stéréovision. Deux configurations sont possibles :

- Configuration *Outside-in* : le système de caméras est monté sur une position fixe et observe la scène. La relation entre la scène et la caméra, i.e. la pose, est fixe. Cependant, le mouvement effectué par les objets composant la scène peut être suivi. Cette configuration est illustrée dans la figure 1.9-a.
- Configuration *Inside-out* : dans ce cas là, la camera est en mouvement (ou peut être attaché à l'objet qui est en mouvement comme par exemple le bras d'un robot, ou un casque de visualisation). La position et l'orientation de la caméra sont estimées continuellement à partir des données extraites des images. En RA, c'est la configuration la plus utilisée. La figure 1.9-b schématise cette configuration. Il existe différentes approches dont quelques unes seront présentées dans le chapitre 2.

1.2.3 Dispositifs de restitution

Les applications de RA disposent d'une panoplie de dispositifs d'affichage qui permettent de visualiser les augmentations, rehaussant ainsi la perception du monde réel. Ces différents dispositifs peuvent être subdivisés, du point de vue technologique en se basant sur la taxonomie présentée par [Milgram et Kishino, 1994], en deux grandes classes qui sont : les dispositifs basés moniteurs et les casques de réalité virtuelle/augmentée.

1.2.3.1 Dispositifs basés moniteurs

Les systèmes basés moniteurs offrent à l'opérateur la possibilité d'observer le monde réel et les objets virtuels superposés à sa vue sans pour autant être équipé de lunettes spéciales. En laboratoire, ce type de dispositif offre la possibilité de tester rapidement les systèmes et de créer ainsi des démonstrateurs peu coûteux. L'opérateur est équipé d'une caméra, l'image capturée est rehaussée avec l'objet virtuel généré puis retournée sur l'écran

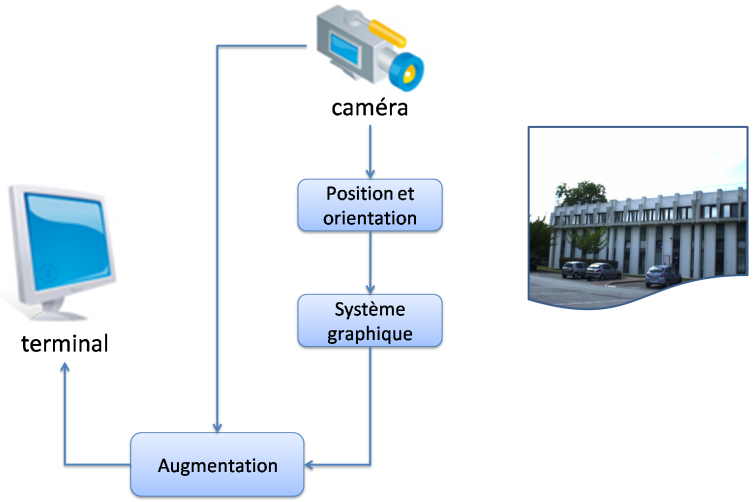


FIG. 1.10: Dispositif basé moniteur

d'affichage (cf. fig.1.10).

Sur le marché, nous trouvons une grande variété de ce type de dispositif. Le plus basique est bien évidemment l'écran de PC. Dans une perspective de mobilité, il existe une gamme de dispositifs portatifs dits (*Hand-held*). L'un des premiers est celui présenté par [Rekimoto et Nagao, 1995] (cf. fig.1.11). Le dispositif *magnifying glass* est une sorte d'écran qui tient dans une main. Ce dispositif est équipé d'une petite caméra qui fournit les images de la scène réelle.

FIG. 1.11: Magnifying glass approach : Premier *Hand-held* [Rekimoto et Nagao, 1995]

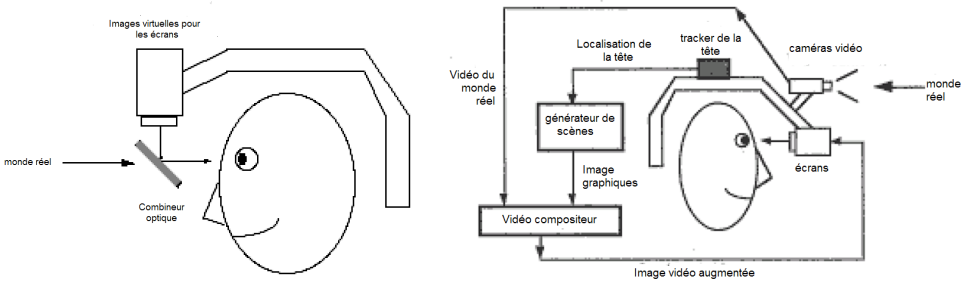
D'autres dispositifs *Hand-held* dits de nouvelle génération ont fait leur apparition comme interface de visualisation. Il s'agit bien évidemment des tablettes PC (cf. fig.1.12-a), PDA (Personal digital Assistants) (cf. fig.1.12-b) et téléphones cellulaires (cf. fig.1.12-c).

(a) Tablette PC (b) PDA (c) Téléphone cellulaire

FIG. 1.12: Les dispositifs de visualisation Hand-held

1.2.3.2 Les casques de RV/RA

Les casques de RV/RA, appelés aussi HMD (*Head Mounted Display*) représentent les dispositifs d'affichage les plus utilisés par la communauté de RA. Il existe deux technologies différentes de casques HMD : les casques de type optique (*optical see-through*) et les casques de type vidéo (*video see-through*). Les casques vidéo sont équipés de deux caméras à travers lesquelles l'environnement réel rehaussé de virtuel est observé. Dans les casques de type optique, l'affichage de l'augmentation est réalisé en utilisant un système de prismes qui permet de superposer les graphiques générés par ordinateur sur la vue directe de la scène réelle.



(a) Casque à vision directe (*optical see-through*) (b) Casque à vision indirecte (*video see-through*)

FIG. 1.13: Classification des casques HMD pour la RA selon [Azuma, 1997]

La figure 1.13-a représente un schéma descriptif du principe de fonctionnement de ce type de dispositif d'affichage. A partir du suivi du mouvement de la tête, la position et l'orientation du point de vue sont estimées et les entités virtuelles projetées sur la vue du monde réel. L'estimation du point de vue est sujette à plusieurs problèmes comme la précision, la calibration, les occultations, et le temps de latence. Certaines de ces difficultés sont contournées en remplaçant les dispositifs optiques par des casques de type vidéo (cf. fig.1.13-b). Ainsi, au lieu de superposer le virtuel directement sur la vision réelle du monde, la vue augmentée va correspondre à une image retournée. Les paramètres internes et externes (position et rotation) de la caméra sont estimés à partir de l'image prise de la scène réelle. Cela permet de réaliser un rendu de l'image virtuelle avec les mêmes caractéristiques que la caméra réelle. La vue réelle et l'image virtuelle sont fusionnées pour ainsi créer une vue augmentée retournée au module d'affichage (écran). Les casques HMD (optique ou vidéo) présentent quelques inconvénients :

- Une perte dans la résolution due aux limitations techniques des écrans utilisés. En ce qui concerne les casques optiques, c'est le module graphique qui souffre d'une faible résolution.
- Le champ de vision est réduit.
- Les dispositifs d'affichage requièrent une procédure de calibration et un suivi précis du point de vue afin d'aligner correctement les entités graphiques. Dans les capteurs de type vidéo, l'intégration est réalisée au niveau du pixel, toutefois le temps de traitement des images introduit un certain temps de latence.
- Présence de malaise du à l'immersion et aux mouvements brusques de la tête de l'opérateur.

Voici quelques exemples de casques existants (cf. fig.1.14). La figure 1.1 (page 8) représente une image de l'un des premiers casques conçu.

FIG. 1.14: Exemples de casques HMD

Après avoir vu de quoi est composé un système de réalité augmentée, nous allons voir quelques systèmes de réalité augmentée existants en particulier les systèmes dédiés à des environnements extérieur.

1.3 Réalité augmentée en extérieur : applications

Depuis quelques années, nous assistons à la multiplication de projets et de travaux exploitant le paradigme de la RA en milieu extérieur. Ces derniers touchent des domaines d'applications aussi divers que le domaine militaire, le génie civil ou l'héritage culturel. Les différentes applications peuvent être regroupées en trois catégories :

- les applications dédiées à la navigation qui fournissent à l'utilisateur des informations pertinentes pour l'aider à évoluer dans son environnement ;
- les applications pour l'accès à l'héritage culturel dans la perspective de constituer un nouveau concept interactif permettant de redonner vie aux vestiges du passé ;
- les applications offrant une assistance aux utilisateurs dans leurs tâches de travail menées dans des environnements extérieurs (par exemple : inspection, etc.).

1.3.1 Applications pour la navigation

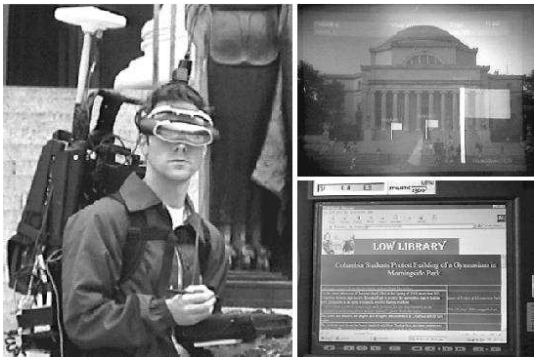
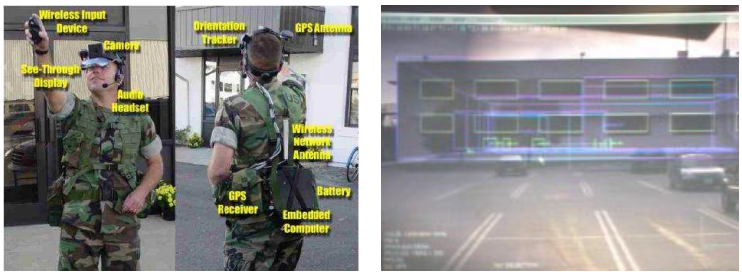


FIG. 1.15: La plate-forme **MARS** et son interface de navigation [Hollerer et al., 1999]

L'équipe de *Steven Feiner* (1996) est pionnière dans ce domaine avec le système **MARS** "*Mobile Augmented Reality System*" [Hollerer et al., 1999]. L'objectif principal était d'explorer la synergie entre le paradigme de la réalité augmentée et l'informatique mobile afin de démontrer la faisabilité d'un système de RA en milieu extérieur qui fournit des informations sur l'environnement. Ce projet devait permettre de créer "une machine de guide personnalisée" (cf. fig.1.15). À partir de la position obtenue avec un DGPS et de l'orientation du point de vue fournie par un capteur inertiel, l'information, correspondante à la vue

courante, est retournée à l'utilisateur via un écran tactile utilisé uniquement pour la visualisation et l'accès au moteur de recherche (l'unité de traitement étant un ordinateur portable porté sur le dos).



(a) (b)

FIG. 1.16: Le prototype **BARS** testé par un militaire [Julier et al., 2000] (a) Plate-forme matérielle (b) Exemple d'augmentation

Ce système a servi de modèle de base pour la mise en œuvre d'autres systèmes tel que le système **BARS** "*Battlefield Augmented Reality System*" (2000) [Livingston et al., 2006]. Le projet **BARS** avait pour objectif de proposer aux militaires un système de navigation et de localisation pour visualiser, en plein champ de bataille, des informations pertinentes comme la position des tireurs embusqués (cf. fig.1.16). À partir de la localisation fournie par un GPS couplé avec une centrale inertielle, le soldat pouvait visualiser les données à travers un casque semi-transparent "*see-through HMD*" et interagir avec elles à l'aide d'une souris ou d'une commande vocale.

(a)

(b) (c) (d)

FIG. 1.17: (a) Plat-forme **Tinmith** [Piekarski et Thomas, 2001] (b) Exemple de visualisation avec **Tinmith** (c) ARQuake [Piekarski et Thomas, 2002] (d) Exemple d'utilisation de la plate-forme pour l'architecture [Thomas et al., 1999]

La plate-forme **Tinmith** (1998) [Piekarski et Thomas, 2001] a permis de mettre au point une architecture matérielle et logicielle pour un système de navigation mobile utilisant la RA. L'architecture logicielle proposée se base sur des objets hiérarchisés, calquée sur le système de fichiers Unix. Le but est de gérer le flux de données issu des différents capteurs, les opérations de filtrage et de rendu. Du point de vue matériel, la plate-forme **Tinmith** est composée essentiellement de capteurs pour la localisation (récepteur GPS et capteur d'orientation), une caméra pour le retour vidéo, un casque pour la visualisation et une unité de traitement (cf. fig.1.17-a).

Le système développé a connu plusieurs déclinaisons pour différentes applications telles que le jeu **ARQuake** [Piekarski et Thomas, 2002] (cf. fig.1.17-c) ou la visualisation de plans architecturaux [Thomas et al., 1999] (cf. fig.1.17-d). De plus, la plate-forme **Tinmith** a été utilisée dans la conception du système **ARVino** [King et al., 2005] pour la visualisation des données géo-localisées en relation avec la viticulture. La RA fournit une assistance sur terrain, en permettant de superposer à la parcelle de viticulture les données la concernant.

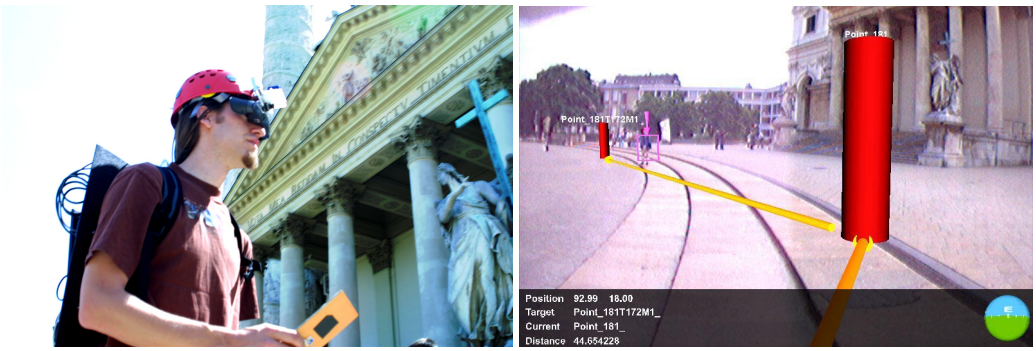


FIG. 1.18: La plate forme **Studierstube** [Reitmayr et Schmalstieg, 2003] (a) La plate-forme matérielle (b) Exemple d'augmentation

Autre exemple, nous citons la plate-forme collaborative développée dans le cadre du projet **Studierstube** [Reitmayr et Schmalstieg, 2003] qui permet à un ou plusieurs utilisateurs d'interagir lors d'une application de navigation (cf. fig.1.18-b). Le système se base sur un positionnement DGPS qui permet d'identifier les données 3D et les afficher via un casque de visualisation 3D stéréoscopique (cf. fig.1.18-a). Le système de navigation proposée utilise comme plate-forme logicielle le système **Studierstube** [Schmalstieg et al., 2002] sur Open Inventor (OIV) [Strauss et Carey, 1992]. Cette plate-forme offre un environnement multi-utilisateurs et multi-applications. Elle comprend plusieurs variétés de dispositifs de visualisation telle que les casques HMD stéréoscopique. Elle offre la possibilité d'interagir avec des objets virtuels ainsi qu'avec des éléments de l'interface utilisateur. L'utilisation d'Open Inventor permet de développer les applications en utilisant les graphes de scènes. Pour les besoins de la collaboration, les auteurs ont mis au point une extension à Open Inventor afin de partager la mémoire sémantique dans la structure de données du graphe de scène. Le processus de localisation est basé sur une centrale inertielle pour l'orientation et un récepteur GPS pour la position.

1.3.2 Applications pour l'accès à l'héritage culturel

Dans les applications dédiées à l'accès aux données sur l'héritage culturel, le premier système proposé est celui mis en œuvre dans le cadre du projet **ARCHEOGUIDE** "*Augmented Reality-based Cultural HEritage On-site GUIDE*" (2000) [Gleue et Dähne, 2001]. Ce projet étudiait plusieurs aspects liés à la RA mobile : la visualisation 3D, l'informatique mobile et les méthodes d'interaction multimodales. En fonction de la position du visiteur sur le site, **ARCHEOGUIDE** filtre les informations et n'affiche que celles qui correspondent aux monuments vus par l'utilisateur. De plus, l'interface développée permet de choisir entre plusieurs thèmes (en différentes langues) et supports multimédia. **ARCHEOGUIDE** offre également la possibilité de visualiser en 3D les parties endommagées des vestiges (cf. fig.1.19-b).

ARCHEOGUIDE combine une localisation grossière, avec un GPS et un compas électronique, et une localisation fine en utilisant une approche basée vision. En effet, dans [Stricker et Kettenbach, 2001], les auteurs proposent d'utiliser une base d'images panoramiques de référence. La localisation fournie par le couple GPS et compas permet de sélectionner l'ensemble d'images de la base les plus proches de la position identifiée. Puis lors de la phase de suivi, l'image courante est mise en correspondance avec les images de la base en utilisant une technique basée sur la transformée de Fourier. L'image de référence ayant le meilleur score est retenue. La transformation 2D entre cette image et l'image courante est ensuite estimée afin de déduire la pose de la caméra.

(a) (b)

FIG. 1.19: Architecture développée dans le projet **ARCHEOGUIDE** [Gleue et Dähne, 2001] (a) La plate-forme (b) Exemple de visualisation : Un temple grec reconstruit en 3D

Autre exemple, la plate-forme **Augurscope** [Schnadelbach et al., 2002], conçu pour créer un musée en plein air, est une interface portable de réalité mixée qui permet d'afficher des avatars virtuels dans le château de Nottingham. En effet tout en explorant le site, des visiteurs peuvent voir des modèles 3D qui peuvent être des reconstructions des lieux passés ou futurs et d'explorer cet environnement en s'aidant d'un contrôleur de point de vue qui offre la possibilité de déplacer, orienter et incliner le dispositif. La plate-forme est composée d'un dispositif d'affichage (écran) monté sur un trépied pivotant équipé d'une caméra (pour le retour vidéo), un récepteur GPS, des accéléromètres et des encodeurs pour la localisation du point de vue (cf. fig.1.20).

Destiné à un auditoire juvénile, le projet **Geist** [Holweg et Schneider, 2004], "fantôme"

FIG. 1.20: Le système **Augurscope** [Schnadelbach et al., 2002]

en français, tente de mettre au point un système de visualisation interactive dans un but éducatif. Equipés d'un casque (cf. fig.1.21-a), les participants à ce jeu éducatif se promènent dans le site (à savoir la cité de *Heidelberg*) et à certains endroits rencontrent des fantômes sous forme d'avatars qui leur expliquent l'histoire qui a façonnée le lieu (cf. fig.1.21-b).

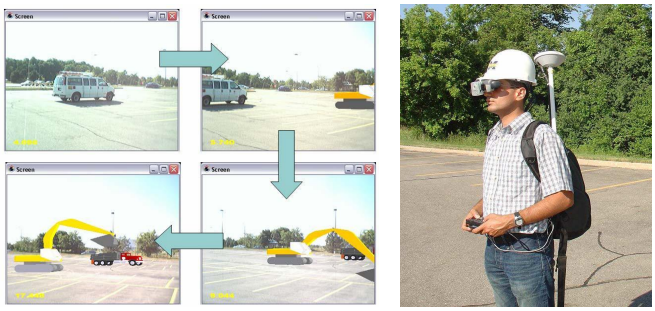
(a) (b)

FIG. 1.21: La plate-forme GEIST(a) équipement utilisé dans la plate-forme **GEIST** [Holweg et Schneider, 2004] (b) Exemple de fantômes intervenant pour raconter l'histoire

1.3.3 Applications pour l'assistance au travail

Pour l'assistance au travail, **ARVISCOPE** (*Augmented Reality Visualization of Simulated Construction Operations*) [Behzadan, 2008] propose une visualisation géo-référencée et dynamique des constructions. **ARVISCOPE** est un outil de visualisation équipé d'un moteur de création en temps réel de scènes animées et dynamiques pour les modèles de constructions (cf. fig.1.22-a).

Pour le calcul du point de vue, **ARVISCOPE** se base sur la plate-forme matérielle **UM-AR-GPS-ROVER** [Behzadan et Kama, 2005] qui est composée d'un casque de visualisation, un récepteur GPS couplé à un capteur inertiel, une caméra pour le retour visuel et un "*Touch Pad*" pour l'interaction (cf. fig.1.22.b).



(a) **ARVISCOPE** : exemple d'utilisation [Behzadan, 2008]

(b) Plate-forme matérielle **UM-AR-GPS-Rover** [Behzadan et Kama, 2005]

FIG. 1.22: Pour la visualisation dynamique des constructions

Vidente (2006) [Schall et al., 2008] est un nouveau système dédié à l'assistance des agents de travaux publics dans leur tâche de maintenance, planification et arpentage des infrastructures sous-terraines (cf. fig.1.23-b). Ce système propose un nouveau dispositif de visualisation et d'interaction spatiale qui est moins encombrant et plus facilement acceptable pour les utilisateurs finaux (cf. fig.1.23-a). Les motivations de ce travail sont dues essentiellement à la démocratisation de l'utilisation des Systèmes d'Information Géographique (SIG) dans la gestion des infrastructures. De plus, les plans papiers, habituellement utilisés, sont remplacés par des terminaux mobiles qui offrent un accès direct aux SIG. Le projet s'intéresse essentiellement aux techniques de visualisation et d'interaction avec les informations SIG et les dispositifs mobiles. Le recalage se base uniquement sur un positionnement primaire par GPS et l'orientation est fournie par une centrale inertielle. Ce qui aboutit à un recalage de faible précision.

FIG. 1.23: (a) Prototype de **Vidente** (b) Visualisation des tuyaux de canalisation [Schall et al., 2007]

IPCity "*Integrated Project on Interaction and Presence in Urban Environments*" (2006) [Markus et Dieter, 2007] est un projet européen dont l'objectif est de développer des techniques d'interaction qui permettent d'exporter la réalité mixte (RM) dans des environnements urbains (cf. fig.1.24). Ceci facilite la participation et la communication entre les différents acteurs d'un même projet (par exemple des élus, des citoyens ou des riverains, dans le cas d'un projet urbain) au moyen d'interfaces mobiles et légères. En effet, les différentes

parties peuvent, par exemple, visualiser des futures constructions dans l'environnement où elles vont être implantées et ainsi donner leur avis et suggestions pour d'éventuelles modifications.

FIG. 1.24: **IpCity** [Markus et Dieter, 2007]

Timewarp [Herbst et al., 2008] est un jeu utilisant la réalité mixée (RM) en extérieur pour explorer l'évolution des villes au fil du temps. Les enjeux ciblés par ce scénario sont essentiellement : l'implémentation d'un jeu RM interactif dans un environnement urbain non restreint, non contrôlable, et ainsi explorer un jeu éducatif. Ce système comprend un casque de visualisation pour visualiser les augmentations. Un UMPC (ultra mobile PC) de type PDA ou téléphone cellulaire (cf.fig.1.25) est utilisé pour fournir des informations sur l'environnement de jeu et les statues des joueurs. De plus, nous retrouvons des capteurs de localisation pour estimer le point de vue. Le système de localisation est composé d'un récepteur GPS, d'un capteur d'orientation 3 DDL et d'une webcam qui permet de réaliser un suivi visuel. Le système dispose d'une base de données hypermédia contenant des modèles 3D, des animations, des sons géo-référencés. L'objectif est aussi de concevoir différents types d'interactions pour les jeux exploitant la RA mobile telles que les commandes vocales ou les contrôleurs d'orientations de type *Wii*.

(a) (b)

FIG. 1.25: **IpCity** : **Timewarp** [Herbst et al., 2008] un jeu pour connaitre l'histoire de sa ville et son évolution [Markus et Dieter, 2007]

Le projet **RAXENV** (*Réalité Augmentée en eXtérieur appliquée aux sciences de l'ENVironnement*) (2007) [Vairon et al., 2007b] a pour but de démontrer la faisabilité d'un système basé sur le paradigme de la RA dédié aux sciences et techniques de l'environnement, comme par exemple la géologie. Ce projet se focalise sur différents aspects techniques de

la RA ainsi que son adoption par les utilisateurs finaux.

(a) Tablette-PC avec la caméra et la centrale inertielle

(b) Récepteur GPS

FIG. 1.26: la plate-forme RAXENV : aspect matériel

Par comparaison à d'autres travaux, **RAXENV** s'intéresse à la mise au point d'un système englobant une localisation précise basée multi-capteurs, une visualisation réaliste et significative via des terminaux mobiles ainsi que des techniques d'interaction en adéquation avec les besoins des utilisateurs issus des sciences et techniques de l'environnement. **RAXENV** propose une solution générique pour les différents aspects traités (localisation, visualisation et interaction) qui s'adapte à différents scénarios. Le système est conçu pour une certaine catégorie d'utilisateurs, tels que des géologues ou des agents d'assainissement dont l'espace de travail est très contraignant. Afin de faire face à ces contraintes, la plate-forme développée est composée d'une tablette-PC (cf. fig.1.26) qui constitue l'unité de traitement et le terminal de visualisation. L'utilisateur interagit avec les données virtuelles (modèles 3D, données SIG, MNT, coupes géologiques) via des *phidget* [Greenberg et Fitchett, 2001]. Le calcul du point de vue combine un GPS, une centrale inertielle et une caméra. Les travaux décrits dans cette thèse rentrent dans le cadre de ce projet exploratoire soutenu par l'ANR (Agence National de la Recherche). Nous y reviendrons ultérieurement.

1.4 Synthèse

Pour mieux visualiser l'évolution des applications de RA en extérieur, le tableau qui suit (cf. tab.1.1) présente une synthèse des applications décrites dans la section précédente. Les applications apparaissent par ordre chronologique.

Dans ce tableau, nous nous intéressons plus précisément au matériel utilisé c'est à dire le types de capteurs (3^{me} colonne), les dispositifs de visualisation et d'interaction (4^{me} colonne). A cela s'ajoute le type de méthodes qui ont été employées dans chaque systèmes ou bien ce qui fait la particularité de ces systèmes. Au fil des années, la réalité augmentée a trouvé son application dans divers domaines allant du génie civil, aux sciences et techniques de l'environnement et passant par le tourisme et la valorisation du patrimoine. Les applications n'ont cessées de se développer et de s'adapter aux besoins des utilisateurs et de leur métiers.

Nom	Année	Capteurs	Dispositifs de visualisation	Techniques et/ou remarques
MARS	1996	GPS et CI	écran tactile et casque HMD	- explorer la synergie entre la RA et l'informatique mobile - architecture matérielle comme modèle de base
Tinmith	1998	GPS et CI	casque HMD	- architecture logicielle basée sur des objets hiérarchisés calqués sur les systèmes de fichier UNIX - utilisé dans différentes applications
BARS	2000	GPS et CI	casque HMD	- basée sur le système MARS - interaction souris et commandes vocales
Archeoguide	2000	caméra	casque/laptop	- visualisation de ruines reconstruites en 3D. - localisation grossière basée GPS et compas - localisation fine basée appariement avec des images de références.
Augruscope	2002	GPS et accéléromètres	un écran monté sur un trépied	- interface portable de RM pour explorer un château.
Studierstube	2003	DGPS et CI et caméra	casque HMD	- application collaborative
Geist	2004	caméra	casque HMD	- interaction avec des avatars.
ARVISCOPE	2005	GPS et CI et caméra	casque	- plate-forme matérielle pour la localisation (UM-AR-GPS-Rover) - interaction en utilisant un Touch Pad
Vidente	2006	GPS et CI et caméra	UMPC	- interface de visualisation et d'interaction plus ergonomique - recalage de faible précision basé GPS et capteur inertiel.
IpCity	2006	GPS et camera	UMPC	- introduire la RM pour une meilleure communication entre les différents acteurs d'un projet urbain
TimeWarp	2006	GPS et webcam et CI	UMPC ou téléphone mobile	- jeu pour connaitre l'histoire d'une ville
RAXENV	2007	GPS et CI et caméra	Tablette PC	- visualisation sur terminal mobile. - interaction avec des phidgets. - localisation basé multi-capteur. - système générique adaptable à plusieurs scénario

TAB. 1.1: Synthèse des plates-formes de réalité augmentée en milieu extérieur

En effet, les premières applications avaient pour principale objectif de démontrer la faisabilité d'un système de RA dans des environnements extérieurs. Ces travaux, tel que **MARS** et **Tinmith**, ont proposé des architectures matérielles et/ou logicielles en adéquations avec le principe de la RA et surtout avec les conditions de travail en extérieur. Nous trouvons dans le projet **MARS** un prototypage matériel suivi par plusieurs autres applications tel que le projet **BARS**. Quant au projet **Tinmith**, les développeurs se sont intéressés d'une part à proposer une architecture matérielle proche de la plate-forme **MARS** ainsi qu'une architecture logicielle s'inspirant du système d'exploitation Unix.

Du point de vue matériel, les premiers systèmes ressemblaient à de gros sac à dos comprenant des unités de traitement (des laptops) auxquelles sont connectées divers type de capteurs, essentiellement un récepteur GPS et des capteurs inertiels. Ces unités de traitement peuvent communiquer avec des serveurs de bases de données distants comme c'est le cas du projet **ARCHEOGUIDE**. Les dispositifs de visualisation étaient soit des casques see-through (pour une majorité) ou bien de simples terminaux. Cependant, les développements connues au niveau technologique et méthodologique a fait évoluer ces applications ainsi que leur besoin. En effet, ces dernières années, l'informatique mobile a gagné en puissance de calcul, en autonomie et en ergonomie (dispositifs plus petit). Ceci a permet de remplacer l'unité de traitement et le casque de visualisation par une seule unité de traitement qui est utilisé aussi pour la visualisation. Ceci est le cas pour les tablette-Pc (RAXENV) et les UMPC (Vidente, IpCity). Certaines applications envisagent même d'utiliser des PDAs ou bien même des téléphones mobiles (TimeWarp). De plus avec les téléphones cellulaires nouvelle génération dotés de capteur de navigation (accéléromètre, GPS, etc), de nouvelles applications sont proposées aux utilisateurs grands publics. Ces applications qui surfent sur le concept de réalité augmentée, proposent de visualiser par exemples les stations de métro les plus proches de l'endroit où vous vous trouvez (cf. fig.1.27).

FIG. 1.27: Exemple de l'application Métro de Paris proposé par Apple pour les I-Phone

A cela, s'ajoute les avancées enregistrées en termes de capteurs qui sont de plus en plus précis, miniatures et surtout à moindre coût. En effet, le gain en précision permet d'imaginer des applications basées sur une localisation précise offrant un potentiel de construire un système de RA où les augmentations ne se restreignent plus à des simples annotations ou affichage d'avatars. Ceci est aussi du à la puissance de calcul des unités de traitement utilisées. Ainsi, la RA mobile en extérieur offre à l'utilisateur un outil d'assistance qui lui permet d'accéder à des bases de données telles que les Systèmes d'Informations Géographiques (SIG) et de représenter la connaissance sous forme visuelle. Par exemple, le géologue peut visualiser, sur terrain, ses coupes géologiques superposées à la vue réelle du volcan. De même, le géomètre pourra visualiser *in-situ* les réseaux d'assainissement modélisés en 3D. De plus, la RA mobile met en avant de nouvelles méthodes d'interaction afin de faciliter la manipulation des données par l'utilisateur final et ainsi enrichir les bases de

connaissances (SIG, modèles CAO, etc.) en confrontant les données à la réalité du terrain et en les mettant à jour en temps réel si nécessaire.

Le déploiement des systèmes de RA en extérieur doit faire face à de fortes contraintes imposées par le milieu extérieur, notamment les conditions climatiques, la technologie employée qui privilégie des solutions nomades à faible puissance de calcul et l'implantation de réseaux de télécommunication pour accéder à des données distantes et centralisées. La conception de systèmes mobiles de RA en extérieur englobe différentes problématiques qui seront détaillées dans la section suivante.

1.5 RA en extérieur : Problématiques

L'objectif principal des applications de réalité augmentée est l'incrustation cohérente d'objets virtuels dans la restitution de la scène réelle fournie à l'opérateur. Différents points doivent être gérés pour aboutir à cette cohérence qui se focalise essentiellement sur le recalage des objets virtuels avec les composantes de la scène réelle.

L'exportation de la RA vers des environnements extérieurs ainsi que les besoins grandissants des applications mises en œuvre a fait évoluer les problématiques. En effet, la mise en œuvre d'un système de réalité augmentée efficace nécessite l'implémentation de trois aspects complémentaires : la localisation, la visualisation et l'interaction (cf. fig.1.28).

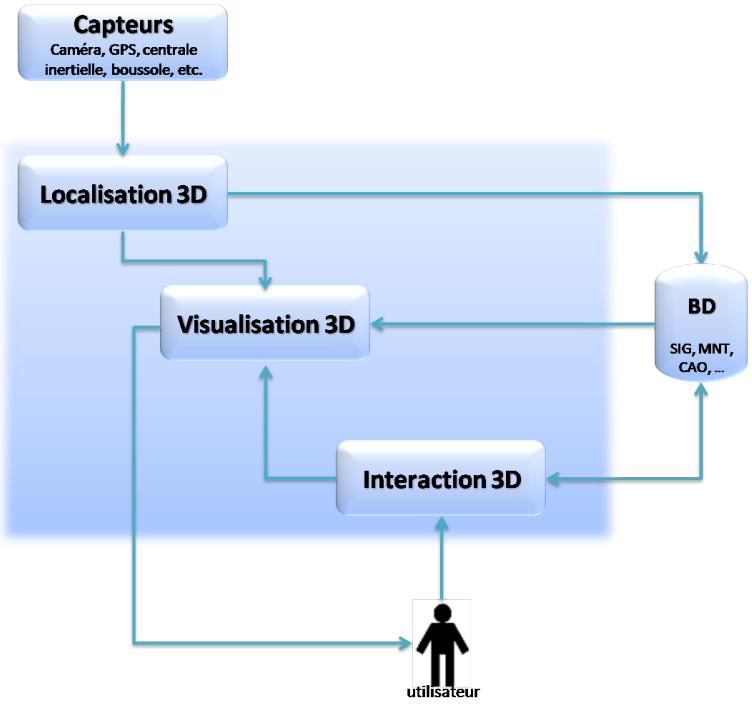


FIG. 1.28: Système de Réalité Augmentée mobile : architecture

1.5.1 Localisation

La RA requiert un bon alignement du réel-virtuel. L'alignement ou recalage consiste à insérer le virtuel en accord avec le repère associé à la scène réelle. Ceci exige une connais-

sance permanente du point de vue de la caméra, c'est-à-dire de sa position et de son orientation dans l'environnement. Cette connaissance permet de créer une caméra virtuelle ayant les mêmes caractéristiques que la caméra réelle.

Outre le besoin de recaler correctement le virtuel et le réel, l'estimation de la position et de l'orientation (la localisation) permet d'identifier, à partir des bases de données géo-référencées, la donnée (modèle 3D, image, etc.) à visualiser.

Or, la mobilité de l'opérateur, dans un environnement non restreint et non préparé, rend difficile le processus de localisation. Les systèmes de RA en extérieur combinent généralement une localisation absolue avec une localisation relative pour situer en permanence l'opérateur. La localisation absolue consiste à déterminer la position de l'opérateur dans l'environnement de travail, grâce à des balises de navigations ou un système GPS. La localisation relative utilise des capteurs de mouvement (accéléromètre, gyroscope, etc.) pour estimer le déplacement relatif de l'opérateur par rapport à une position de référence. La problématique de localisation réside dans la manière de combiner ces deux types de localisation.

Le processus de localisation doit faire face aux conditions régnant dans l'environnement, essentiellement les variations de luminosité, les changements d'échelle, les occultations et les mouvements brusques. A cela s'ajoutent les contraintes imposées par un environnement extérieur à grande échelle non contrôlé telle que la complexité et l'indisponibilité de modèles 3D précis et complets. A cela s'ajoute les perturbations des mesures fournies par les différents capteurs utilisés et qui sont difficilement compensables.

1.5.2 Visualisation

À partir de la position et de l'orientation du point de vue, les données virtuelles sont recalées sur la vue réelle. Cependant, la mobilité en milieu extérieur contraint l'utilisateur à visualiser les données sur des terminaux, dits mobiles, tels que des tablettes-PC, des PDAs ou des téléphones cellulaires (cf. fig.1.10).

Le défi est d'effectuer le rendu graphique des éléments virtuels de manière à ce qu'ils apparaissent comme appartenant au monde réel. Pour aboutir à un résultat réaliste, ce rendu doit prendre en compte d'une part la cohérence des profondeurs et gérer les occultations (i.e. l'objet le plus proche doit masquer un objet plus éloigné) et d'autre part, la cohérence photométrique entre les objets réels et virtuels.

Le problème est que les dispositifs de visualisation mobiles offrent des capacités limitées de traitement et de stockage, d'où l'intérêt d'adapter les techniques existantes de visualisation ou de mettre au point de nouvelles méthodes qui opèrent en temps réel et qui permettent de manipuler de grandes masses de données.

1.5.3 Interaction

Les applications de RA doivent offrir la possibilité de manipuler et d'interagir avec les données 3D visualisées. Cette tâche est réalisée au moyen d'interfaces Homme-machine (IHM). Dans les terminaux classiques (style PC), les outils d'interaction se basent sur des notions telles que les fenêtres, les icônes, les menus, etc (système WIMP).

Les contraintes des terminaux mobiles étant différentes de celles des PCs de bureau, les méthodes d'interaction homme/machine ont dû évoluer. Concernant l'interaction 3D, une mutation similaire des interfaces est nécessaire. En effet, les méthodes d'interaction qui ont été développées dans un contexte de bureau ne sont pas forcément adaptées au contexte de mobilité. Interagir avec des environnements 3D virtuels est un processus complexe. Il existe plusieurs dispositifs pour l'interaction 3D sur terminaux mobiles, à savoir : Les écrans tactiles, les touches de téléphones, les caméras, et les capteurs de positions et orientations. Un état de l'art sur les approches de visualisation et d'interactions 3D pour des applications de réalité augmentée a été présenté dans [Zendjebil et al., 2009]. Je conseille aux lecteurs de s'y référer.

1.6 Objectifs de la thèse

Le processus de localisation représente le cœur d'un système de RA. En effet, à partir de la connaissance de la position et de l'orientation du point de vue de l'utilisateur, les données sont alignées correctement pour être visualisées sur la vue de la scène réelle. Pour cela, tout système de réalité augmentée, qu'il soit dédié aux applications en extérieur ou non, doit être capable de se localiser continuellement dans son environnement.

L'objectif de cette thèse se focalise sur la mise au point d'une approche de localisation dans des environnements extérieurs dédiée à la réalité augmentée.

Notre travail s'oriente vers une approche combinant plusieurs capteurs de nature hétérogène afin de permettre une localisation permanente et en temps réel sous toutes les conditions de travail.

La mise en œuvre de cette combinaison soulève plusieurs problématiques à savoir :

- **Le choix de la stratégie de combinaison** : En effet, la combinaison des différents capteurs utilisés doit être faite de façon à tirer parti de l'ensemble des données disponibles à l'instant t. Cependant, cette combinaison ne doit pas altérer ni la précision ni la robustesse des mesures obtenues.
- **La calibration des systèmes multi-capteurs** : Les données brutes des capteurs sont exprimées dans des systèmes de repère propre à chaque capteur. Afin de fournir des mesures uniques, il faut unifier ces référentiels. Pour cela il faut mettre en œuvre des procédures de calibration qui permettent d'identifier les transformations rigides entre les repères des différents capteurs qui permettent d'exprimer des données fournies dans un repère d'un capteur dans le repère d'un autre capteur.

De plus, les enjeux d'un tel système de localisation sont essentiellement :

- **La précision** : L'information de position et d'orientation doit être assez précise pour avoir un bon recalage. Or, les données fournies par les différents capteurs utilisés ont des degrés de précision différents. De ce fait, il faut quantifier ces erreurs afin d'améliorer la précision globale du système.
- **La robustesse** : Le système doit être robuste face aux conditions de travail. Il doit être capable de s'adapter aux différentes situations auxquelles il est confronté. Comme par exemple, la défaillance d'un des capteurs et son incapacité à fournir une mesure cohérente.

Nous nous sommes attelés à proposer une solution générique qui peut être fonctionnelle dans différents types d'environnements et sous différentes conditions.

1.7 Conclusion

Dans le présent chapitre, nous venons d'effectuer un tour d'horizon autour de la réalité augmentée, ses applications ainsi que ces problématiques. En effet, après avoir présenté le paradigme de la réalité augmentée, nous avons décrit les composants d'un système de RA en général. Le chapitre a comporté une étude sur les systèmes de RA développés en environnements extérieurs. En effet, l'application de la réalité augmentée en extérieur devient de plus en plus répondue et trouve sa place dans divers domaines que ce soit en tant que système d'assistance dans certain type de métiers ou bien en tant que système ludo-éducatif. Cet intérêt ne cesse de grandir essentiellement du aux développements des technologies telle que la téléphonie qui commence à démocratiser le concept de la RA en proposant des applications pour le grand public.

A partir de l'étude réalisée sur les différents systèmes présentés, ceci nous a permis de cerner les différentes problématiques entourant leur conception et mises en œuvre. Comme nous l'avons vu, ces problématiques axent essentiellement sur trois aspects complémentaires qui sont : la localisation, la visualisation et l'interaction.

Comme nous l'avons cité auparavant, cette thèse s'intéresse au problème de localisation qui représente un processus important dans tout système de RA. Ce processus est d'autant plus difficile dans des environnements extérieurs. Plusieurs travaux, qui se sont penchés sur ce processus, se sont orientés vers des systèmes de localisation combinant différents type de capteurs. Cependant dans un premier temps, nous allons nous intéresser aux méthodes de localisation basées vision. En effet, ces méthodes restent les méthodes les plus utilisées essentiellement dans les applications de vision indirecte. Dans le chapitre qui va suivre, nous allons présenter dans un premier temps quelques approches utilisées essentiellement en RA. Dans un second temps, nous allons nous intéresser aux approches que nous avons mises au point dans le cadre de nos travaux.

Chapitre 2

Localisation basée vision

Se localiser dans l'environnement constitue un processus important pour de nombreux systèmes qu'ils soient dédiés à la réalité augmentée ou à la robotique. Comme nous l'avons mentionné précédemment, dans les applications de réalité augmentée, la connaissance de la position et de l'orientation du point de vue de l'utilisateur (i.e. de la caméra) est nécessaire, car elle permet d'assurer la cohérence spatio-temporelle de la scène augmentée. En effet, l'estimation des paramètres de localisation permet de modéliser une caméra virtuelle grâce à laquelle un rendu du monde virtuel est réalisé avec les mêmes caractéristiques que la caméra réelle, permettant d'aligner correctement les mondes réel et virtuel et ainsi de créer la vue augmentée.

La majorité de ces systèmes, notamment ceux qui utilisent la vision indirecte (c'est-à-dire les dispositifs de visualisation de type casques vidéo *see-trough* ou bien des écrans), utilise des approches basées vision. Cette classe de méthode permet d'estimer la position et l'orientation du système dans l'environnement à partir des informations extraites des images retournées par la caméra. L'estimation de la position et de l'orientation de la caméra est appelée, en vision par ordinateur, **estimation de pose**. Elle revient à recouvrer les caractéristiques extrinsèques de la caméra en se basant sur le modèle de projection appelé modèle sténopé (cf. annexe A).

L'estimation de pose peut se faire de plusieurs manières. Ceci dépend du type d'informations utilisées ainsi que de la connaissance dont nous disposons de la scène. Dans la première partie de ce chapitre, nous présenterons les méthodes les plus communément utilisées dans la communauté RA. Nous allons nous intéresser essentiellement aux méthodes adaptées aux environnements extérieurs. En seconde partie du chapitre, nous présenterons en détail les approches que nous avons mises au point. La dernière partie concernera les expérimentations conduites en conditions réelles. Elles ont pour but de démontrer les avantages des méthodes de vision et surtout d'identifier leurs limites.

2.1 Taxonomie des méthodes d'estimation de pose

L'estimation de pose est depuis longtemps un problème qui fait l'objet de l'attention de la communauté de la vision par ordinateur. En ce qui nous concerne, nous allons nous restreindre aux méthodes utilisées dans les applications de réalité augmentée. Ces méthodes peuvent être regroupées en deux classes selon le degré de connaissance dont nous dispo-

sons de la scène (connaissance totale, partielle ou nulle de l'environnement). La majorité des méthodes exploitées dans la communauté utilisent des approches qui ont une connaissance *a priori* de l'environnement. Le fait de disposer d'une base de connaissance sur l'environnement permet de l'exploiter par les algorithmes d'estimation de pose. Cependant, nous trouvons depuis peu de nouvelles méthodes d'estimation de pose pour les applications de RA qui ne supposent aucune connaissance de l'environnement ou du moins une connaissance partielle pouvant être enrichie en ligne. Nous allons présenter le principe des méthodes les plus connues.

2.1.1 Approches avec connaissance *a priori*

Cette classe de méthodes d'estimation de pose utilise un modèle mettant en relation la scène réelle avec l'image courante de celle-ci. Cette relation se base sur le modèle de projection qui décrit la transformation d'un point 3D défini dans le repère associé à la scène en un point 2D défini dans le repère de l'image. De ce fait, il suffit de retrouver l'ensemble des données 2D extraites des images qui correspondent aux données 3D visibles du modèle. Pour établir cette relation, nous devons disposer d'une connaissance totale ou partielle de l'environnement. Elle se schématise soit par la connaissance des coordonnées 3D associées à certaines données pertinentes faciles à extraire des images ou bien par des structures simplifiées basées sur la scène réelle.

Les approches existantes peuvent être classées en deux grandes catégories selon la nature des informations extraites. La première catégorie d'approches se base sur l'instrumentation des scènes avec des marqueurs artificielles. Quant à la seconde, elle exploite des informations naturelles qui existent dans la scène. Nous survolerons la première approche car, comme nous le verrons, elle n'est pas adéquate aux types d'environnements qui nous intéressent.

2.1.1.1 Méthodes basées marqueurs

De nombreux systèmes de RA utilisent des marqueurs artificiels placés dans la scène réelle afin de faciliter l'estimation de pose. Ces marqueurs sont particulièrement simples à détecter et peuvent contenir un code. Ainsi, il est facile de les distinguer et plusieurs cibles peuvent êtres simultanément suivies dans une même scène. Partant du principe que la position des marqueurs dans le repère monde est connue *a priori*, l'estimation de pose se décompose en plusieurs phases :

1. **Détection et identification des marqueurs** : cela consiste à extraire de l'image les marqueurs visibles par la caméra et de les identifier ;
2. **Mise en correspondance 2D/3D** : à chaque marqueur identifié dans l'image est associé sa position 3D ;
3. **Calcul de la pose de la caméra** : estimation de la pose à partir des appariements 2D/3D.

Il existe deux types de marqueurs. Nous retrouvons des marqueurs de forme circulaire telle que les marqueurs CCC (pour Concentric Contrasting Circle) [Hoff et al., 1996] qui sont formés de cercles noirs sur un fond blanc ou vice-versa. Ces marqueurs se distinguent par le fait qu'ils sont invariants aux distorsions perspectives. De plus, leur centre de gravité permet de fournir une position 2D stable. Ce type de marqueurs est utilisé dans les travaux de [L.Naimark et E.Foxlin, 2002] et [Claus et Fitzgibbon, 2004]. Le second type de marqueurs a été introduit dans [Koller et al., 1997]. Ces derniers ont une forme carrée et sont de

couleur noire et blanche. Ces marqueurs de forme rectangulaire ont été très utilisés dans les applications de RA. Nous les trouvons par exemple dans les travaux de [Kato et al., 2000] et [Rekimoto, 1998]. En effet, ce type de marqueurs est devenu populaire, notamment après le succès qu'a connu la bibliothèque ARToolKit [Kato et Billinghurst, 1999] (cf. fig.2.1). Les marqueurs utilisés dans cette bibliothèque ont des bords noirs sur un fond blanc. La partie interne du marqueur contient un code permettant de l'identifier. L'équipe RATC s'est intéressée aux méthodes utilisant ce type de marqueurs. Nous les retrouvons dans les travaux de *F. Ababsa* [Ababsa et Mallem, 2004, Ababsa et Mallem, 2006], de *M. Maidi* [Maidi, 2007] et de *J-Y. Didier* [Didier et al., 2006].

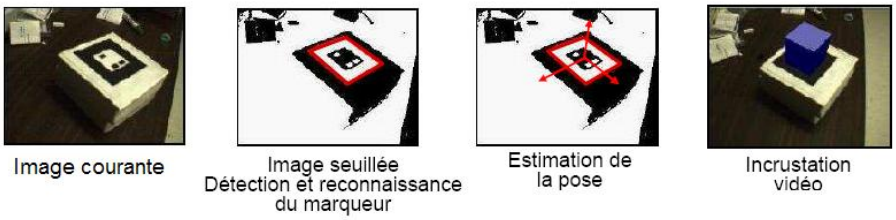


FIG. 2.1: Exemple d'utilisation d'**ARToolKit** [Kato et Billinghurst, 1999]

Les méthodes basées marqueurs sont assez précises. Néanmoins, leur inconvénient majeur est qu'il faut disposer les marqueurs dans l'environnement de travail de façon à ce que la caméra puisse toujours en détecter au moins un, ce qui est inadapté pour les environnements à grande échelle essentiellement en extérieur. Toutefois, quelques applications de RA mobile en extérieur utilisent ces cibles codées. C'est le cas dans [Piekarski et Thomas, 2002] où les auteurs utilisent, pour le module de vision, des marqueurs de la bibliothèque **ARToolKit** [Kato et Billinghurst, 1999] d'une surface d'un mètre carré placés sur les bâtiments (cf. fig.2.2). Ceci permet d'estimer la pose de la caméra lorsque l'utilisateur est proche des constructions. Cependant, pour les applications en extérieur, les méthodes utilisant les données naturelles sont les plus plébiscitées.

FIG. 2.2: Exemple d'utilisation de marqueurs en extérieur [Piekarski et Thomas, 2002]

2.1.1.2 Méthodes sans marqueurs ou "markerless"

En effet, les méthodes sans marqueurs ou *markerless* représentent une alternative à l'utilisation des marqueurs artificiels en exploitant les caractéristiques naturelles existantes dans la scène réelle telle que des coins, des contours, des segments de droites, etc. Généralement, ces approches utilisent des modèles 3D qui constituent une connaissance *a priori* de l'environnement où évolue l'opérateur. Les données 2D extraites de l'image de la scène sont mises en correspondance avec les données 3D (de même type) extraites du modèle (cf. fig.2.3).

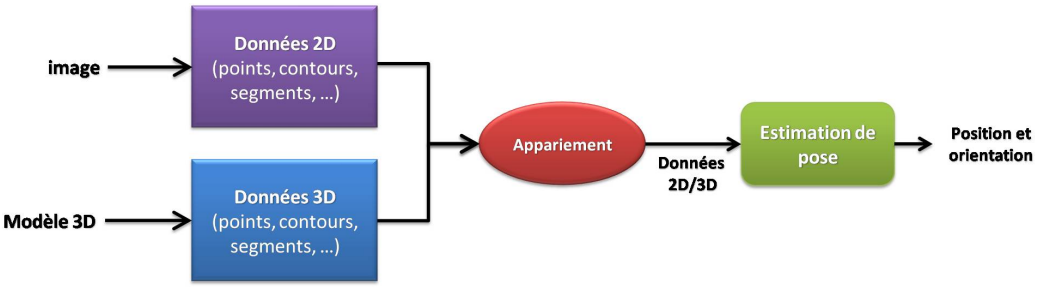


FIG. 2.3: Chaîne de traitement associée au calcul de pose

Les méthodes utilisant comme information les contours ou segments de droite sont les approches les plus utilisées. Elles ont l'avantage d'être très efficaces face aux changements de luminosité étant donné que les contours constituent des invariants globaux. Nous pouvons distinguer deux catégories d'approches. La première catégorie ne nécessite pas l'extraction explicite des contours. Le principe de l'appariement consiste à chercher le maximum du gradient le long de la normale aux points des contours projetés. Une fois cette phase terminée, la pose est estimée en utilisant un processus de minimisation basé sur un modèle. Plus précisément, les paramètres de pose sont estimés en minimisant la distance entre le contour projeté et le contour de l'image. Brièvement, parmi ces approches nous citons [Armstrong et Zisserman, 1995], [Comport et al., 2003], [Drummond et Cipolla, 1999] et [Marchand et Bouthemy, 2001]. Si dans la première catégorie il n'est pas nécessaire d'extraire explicitement les contours, dans la seconde catégorie, les contours ou segments doivent être extraits afin de les apparier avec les contours ou segments projetés du modèle 3D. Parmi les approches existantes dans la littérature, nous citons [Gennery, 1992] [Lowe, 1992].

FIG. 2.4: Etapes de l'approche utilisée dans [Reitmayr et Drummond, 2006]

La première catégorie est généralement la plus utilisée pour sa rapidité et la seconde pour sa robustesse. Parmi les nombreux travaux de ce type, nous retrouvons le système proposé par [Reitmayr et Drummond, 2006] qui se base sur la méthode décrite dans les travaux de [Klein et Drummond, 2003]. La méthode estime les parties visibles des contours projetés selon un point de vue prédit, puis le mouvement de la caméra qui permet d'aligner les contours projetés avec les contours images. Le mouvement représente le déplacement effectué dans la direction de la normale au contour. Ce mouvement permet de mettre à jour la pose prédite. Dans [Comport et al., 2006], les auteurs proposent un algorithme formulé en termes d'asservissement visuel virtuel pour un suivi local de contours. La loi de commande en boucle fermée minimise l'erreur entre la position courante et la position souhaitée de la caméra. L'utilisation d'un algorithme de type M-estimateurs permet de gérer les données aberrantes. Cette approche est robuste face aux occultations partielles et aux changements d'illumination. Elle converge rapidement pour de petits déplacements. Cette approche fournit des estimations satisfaisantes. En effet, l'estimation de la pose initiale

et du déplacement fournissent des augmentations (cf. fig.2.5), stables mais sujettes à de faibles effets de glissement (*jitter*). Une initialisation manuelle est requise pour la première image. Afin d'automatiser cette procédure, les auteurs proposent d'utiliser des descripteurs géométriques locaux, tels que les descripteurs SIFT [Lowe, 2004], associés à des images de références. À partir de l'appariement de plusieurs ensembles de points, la transformation entre l'image courante et les images de référence est estimée.

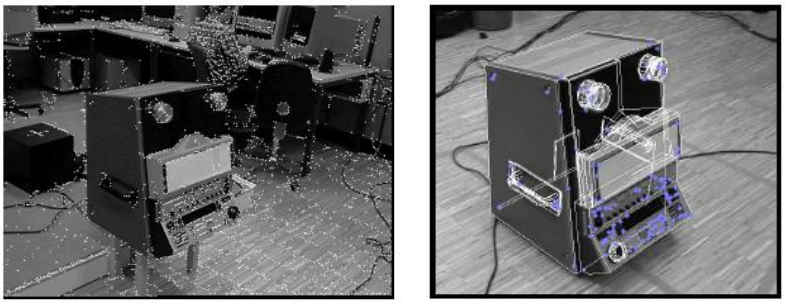
FIG. 2.5: Exemple d'augmentation d'un environnement extérieur [Comport et al., 2006]

Ces méthodes supposent que le premier contour rencontré dans la direction de la normale est le contour correspondant. Pour son bon fonctionnement, les approches qui utilisent cette hypothèse supposent que le mouvement entre deux images successives est faible. Pour pallier ceci, certaines méthodes prennent en compte plusieurs correspondants potentiels trouvés dans la direction de la normale. Par exemple, dans [Wuest et al., 2005], les auteurs proposent d'utiliser un critère à hypothèses multiples pour calculer les paramètres de la pose. En effet, au lieu de supposer que le point ayant un fort gradient dans la direction de la normal est le correspondant, ce qui n'est pas toujours vrai, l'approche proposée intègre dans la fonction de coût la prise en compte du point le plus proche du point contour projeté à chaque itération dans le processus de minimisation. Ceci permet de mettre à jour les correspondances à chaque itération et ainsi de rectifier les mauvais appariements.

L'inconvénient majeur de ces méthodes apparait lorsque l'environnement est très texturé. En effet, en présence de textures, des faux appariements peuvent être engendrés ce qui induit des erreurs dans l'estimation de pose. De ce fait, il existe une seconde classe de méthodes basées textures. La texture peut être utilisée sous forme de points d'intérêt ou sous forme d'imagettes (échantillons d'image). Selon le type d'information utilisée, les méthodes d'estimation de pose utilisent soit un suivi de points 2D ou bien l'alignement d'images ou d'imagettes.

Les approches de suivi 2D consistent à suivre un ensemble de données 2D d'une image à une autre. Cet ensemble de données peut correspondre à la projection des points 3D visibles du modèle. Il suffit donc d'extraire ces projections dans une image (appariement 2D/3D) et de les suivre. Il existe plusieurs méthodes de suivi 2D de points. Ces approches se basent, généralement, sur l'apparence du point d'une image à une autre en utilisant le principe de conservation de luminosité appelée contrainte Lambertienne (i.e. un point à la même intensité ou presque). Ceci peut être réalisé par une simple corrélation qui consiste à chercher les appariements d'un point défini dans une image i avec un ensemble de points extraits dans l'image $i+1$ en calculant un score de corrélation telle que le SSD (*Sum Squared Difference*) ou le ZNCC (*Zero mean Normalized Cross-Correlation*). Les couples d'appariements choisis correspondent aux couples ayant les plus grands scores de corrélation, i.e. la plus grande similitude. Cependant ces méthodes sont très consommatrices en temps de calcul et ne sont pas robustes face aux variations de luminosité. Leur intérêt, comme nous

l'avons dit auparavant, est que les points suivis peuvent représenter des points pertinents dans le modèle tels que les coins. D'autres types de méthodes peuvent être utilisés. Elles sont basées sur des descripteurs qui ont la particularité d'être invariants aux changements de luminosité, d'échelle et/ou de rotation. Le plus connu est le descripteur présenté dans [Lowe, 2004] appelé SIFT (*Scale-Invariant Feature Transform*) qui propose aussi un détecteur de points d'intérêts. Cependant, ces approches sont assez consommatrices en temps de calcul, essentiellement lors de l'extraction des données. Il existe des méthodes plus rapides qui approximent les points de SIFT telles que l'opérateur SURF [Bay et al., 2008]. Le problème qui se pose est que les points extraits par ces opérateurs ne peuvent pas correspondre à des points 3D du modèle. De ce fait, au lieu de les utiliser pour suivre directement les projections 2D des points 3D, ces méthodes sont exploitées d'une autre manière. En effet, les couples d'appariements peuvent être utilisés pour estimer le mouvement effectué par la caméra d'une image à une autre. Il suffit par la suite de mettre à jour la pose prédite qui est généralement la pose estimée à l'instant t. Par exemple, le mouvement de la caméra peut être calculé à partir de la matrice essentielle mais à un facteur d'échelle près. Cette méthode peut être intéressante lorsque le mouvement de translation est petit voir négligeable d'une image à une autre. Le mouvement peut alors être approximé par un mouvement rotationnel pur.



(a) Extraction des points d'intérêts dans l'image

(b) Prise en compte des points sur l'objet uniquement

FIG. 2.6: Méthode d'estimation de pose proposée dans [Lepetit et al., 2003]

Dans [Lepetit et al., 2003], les auteurs proposent une méthode basée points d'intérêts qui en utilise un modèle 3D partiel de la scène. Leur approche est subdivisée en deux phases : une phase d'initialisation automatique et une phase de suivi. L'initialisation automatique comprend une phase d'apprentissage hors ligne durant laquelle les points d'intérêts sont détectés dans des images calibrées de l'objet à suivre (cf. fig.2.6-a). Seuls les points appartenant à l'objet sont pris en considération (cf. fig.2.6-b), et des coordonnées 3D définies dans le repère associé à l'objet leur sont attribuées. Pour chaque point extrait, la méthode réalise un rendu selon différents points de vue. En ligne, les points d'intérêt sont détectés dans l'image courante et mis en correspondance par corrélation avec les points extraits lors de la phase d'apprentissage. L'utilisation d'une ACP (Analyse en composante principale) permet d'accélérer le calcul de la corrélation. La pose est estimée de manière robuste à partir des correspondances 2D/3D en utilisant RANSAC. Une fois la pose estimée, le suivi est réalisé de deux manières différentes. La première méthode utilise des images de références de l'objet acquises lors de l'étape hors ligne en mettant en correspondance l'image courante avec les images de référence rendues selon la pose estimée à l'instant $t-1$.

Cette approche gère les changements d'aspects et évite les dérives étant donné que la pose est estimée indépendamment d'une image à l'autre. Toutefois, elle est peu précise. La deuxième méthode consiste à suivre des points d'intérêts d'une image (image $t-1$) à une autre (image t) afin d'augmenter la précision. Les points considérés sont les projections des points 3D sur l'image de l'objet. La méthode estime la pose de la caméra ainsi que les positions 3D des points suivis en minimisant simultanément deux critères d'erreur de reprojection.

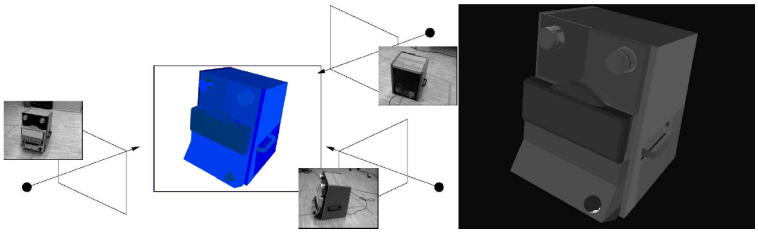


FIG. 2.7: Exemple du modèle utilisé et des images de références dans [Vacchetti et al., 2004]

Une autre manière de procéder consiste à utiliser le principe d'alignement d'image ou d'imagettes. L'alignement d'image peut être présenté comme étant le processus qui permet d'aligner une image ou une partition d'une image dans le repère d'une autre image de telle façon à faire correspondre les pixels des deux images. Cet alignement est obtenu en calculant une transformation qui permet de faire le passage d'une image à une autre. La procédure consiste à estimer une transformation qui minimise l'erreur entre l'image $t+1$ et l'image t transformée dans le repère de l'image courante. Cette erreur représente la différence de luminosité entre deux points images qui se correspondent. La transformation estimée est décrite en fonction des paramètres du mouvement de la caméra effectué entre les deux images. Ce mouvement permet de mettre à jour la pose estimée à l'instant t. L'approche la plus connue est l'approche décrite par Lucas-Kanade [Lucas et Kanade, 1981] qui propose d'utiliser une minimisation dite de premier ordre. Cette minimisation utilise une linéarisation de la fonction de coût (la somme des différences quadratiques entre les points de l'image $t+1$ et les points de l'image t transformé) en utilisant une décomposition de Taylor du premier ordre. Il existe d'autres méthodes qui utilisent une décomposition du second ordre telle que l'algorithme ESM [Benhimane et Malis, 2004, Ladikos et al., 2008] qui a été utilisé pour le suivi de surface planaire. Ces méthodes ne requièrent pas l'extraction de caractéristiques telles que des contours ou des points d'intérêts mais utilisent généralement la totalité de l'information existante dans la zone à suivre (l'imagette).

Nous retrouvons ce principe dans plusieurs travaux comme par exemple dans les approches proposées par [Bleser et al., 2006], [Hol et al., 2006] et [Zollner et al., 2008]. Par exemple, *G. Bleser* propose une approche dite descendante (*Top down approach*). En effet, à partir d'une pose prédite, un modèle texturé est projeté afin d'obtenir une image synthétique qui ressemble à la vue courante. Puis, des imagettes, dont les barycentres sont des points caractéristiques, sont alignées itérativement avec l'image courante en utilisant une variante du KLT (Kanade-Lucas Tracker) qui incorpore un modèle photométrique affine. Afin d'améliorer la convergence de l'approche, l'alignement se base sur une pyramide d'images. Ceci permet de réaliser l'alignement au niveau le plus grossier et de le propager jusqu'au niveau le plus fin des pyramides.

Parmi les méthodes basée textures, il existe des approches qui n'utilisent pas un modèle 3D. En effet, même si l'utilisation de modèles 3D permet d'améliorer la robustesse et les performances des méthodes de localisation, la construction de modèles précis pour des environnements à grande échelle reste délicate. Pour cela, dans [Stricker et Kettenbach, 2001], les auteurs proposent d'utiliser, dans le projet **ARCHEOGUIDE** [Gleue et Dähne, 2001], une base d'images panoramiques de référence (cf. fig.2.8) au lieu d'un modèle 3D. Lors de la phase de suivi, l'image courante est mise en correspondance avec les images de la base en utilisant une technique exploitant la transformée de Fourier. L'image de référence ayant le meilleur score est retenue. La transformation 2D entre cette image et l'image courante est ensuite estimée afin de déduire la pose de la caméra. L'approche est robuste face aux changements de luminosité. De plus, les différents traitements sont effectués avec un nombre fixe d'opération ce qui fait que la méthode s'exécute en temps réel.

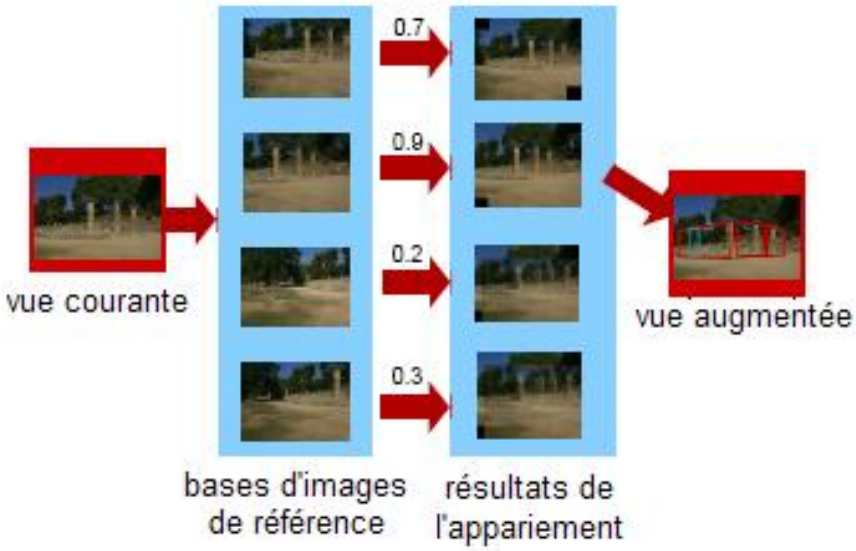


FIG. 2.8: Suivi avec d'images de référence [Stricker et Kettenbach, 2001]

Parmi les méthodes d'estimation de pose qui se détachent aussi de l'utilisation explicite d'un modèle 3D classique de type maillé ou surfacique, nous retrouvons une autre classe d'approche qui exploite plutôt la connaissance de la structure de la scène. Parmi ces approches, nous citons la méthode proposée par [Simon et Berger, 2002] qui proposent d'exploiter la présence de plans dans l'environnement aussi bien en intérieur qu'en extérieur. Ceci permet de simplifier le modèle de projection de la caméra. La méthode proposée choisit le repère monde de manière à suivre le plan défini par $z = 0$, ce qui permet de définir une homographie qui transforme un point du plan 3D dans l'image i. Brièvement, l'idée est d'estimer la meilleure homographie qui relie les points appartenants au même plan dans les images successives. Ceci permet de calculer le mouvement différentiel entre ces deux plans pour ensuite l'utiliser afin de déduire le mouvement de la caméra. En effet, lors de la phase de suivi, l'approche calcule l'homographie reliant l'image i à l'image $i+1$, en détectant les points d'intérêts telle que des coins de Harris par exemple dans l'image $i+1$, puis les apparie avec les points du plan obtenus dans l'image i. En pratique, l'approche a besoin de 4 points au minimum. Par la suite, l'homographie reliant le monde à l'image courante est

calculée. Ensuite, il suffit de déduire la rotation et la translation sachant que les paramètres intrinsèques sont connus. A l'étape initiale, les quatre coins formant le plan sont définis manuellement puis les points à suivre d'une image à une autre sont extraits à l'intérieur du plan. Cependant, la méthode n'a pas besoin de définir explicitement les coordonnées 3D des coins du plan. En effet, les coordonnées sont définies à un facteur d'échelle qui peut être calculé à partir des éléments de l'homographie calculée. De plus, l'approche permet de recommencer le suivi avec un nouveau plan détecté dans le flux d'images. Pour cela, il suffit juste de redéfinir un nouveau système de coordonnées monde en accord avec le nouveau plan. Pour rendre le calcul robuste, l'approche se base sur la méthode RANSAC [Fischler et Bolles, 1981]. L'utilisation de RANSAC rend le suivi robuste aux occultations partielles et aux données aberrantes. L'approche se distingue par sa simplicité de mise en œuvre, sa rapidité et sa précision. La méthode présente une erreur de l'ordre de 2.5 pixels qui représente le seuil de tolérance défini pour RANSAC. L'approche ne présente pas un effet de glissement (*jitter*). Toutefois, nous constatons la présence de dérive (*drift*) due à l'accumulation des erreurs d'estimation. L'approche peut être appliquée pour un suivi multi-plan en calculant la matrice de projection à partir de l'ensemble des appariements obtenus dans chaque plan.

(a)

(b)

FIG. 2.9: Suivi planaire proposé dans [Simon et Berger, 2002] : (a) Définition du plan à suivre, extraction des points et suivi (b) Recalage obtenu

Les méthodes basées contours ont l'avantage d'être efficaces robustes aux changements d'illumination, cependant elles sont inadaptées aux scènes très texturées. En revanche, les méthodes basées textures sont robustes aux occultations mais restent sensibles aux changements d'illumination. Ces deux limitations ont fait émerger une nouvelle catégorie d'approches dites "hybrides". Ces méthodes combinent différentes primitives visuelles, tels que les contours et la texture, pour réaliser le suivi. L'approche développée par [Vacchetti et al., 2004] combine des contours avec des points (cf. fig.2.10). En effet, pour chaque nouvelle image, des points d'intérêt sont extraits puis mis en correspondance avec des points de référence obtenus dans des images acquises lors d'une étape hors ligne. Cet appariement permet d'avoir des couples de correspondance 2D/3D. De plus, les contours 3D sont projetés selon la pose estimée à l'instant $t-1$ pour effectuer l'appariement avec les contours de l'image courante. La pose de l'image courante est obtenue en minimisant

simultanément l'erreur de reprojection des points d'intérêts, l'erreur de reprojection des points mis en correspondance avec l'image de référence et la distance entre les contours projetés et les contours image.

FIG. 2.10: Estimation de pose basée suivi hybride [Vacchetti et al., 2004]

L'approche proposée par [Pressigout et Marchand, 2005] recouvre la pose de la caméra en estimant une transformation 2D pour le suivi d'une structure planaire. Les paramètres de cette transformation sont obtenus à partir d'une minimisation itérative d'un critère qui combine les informations sur la texture et sur les contours de l'objet à suivre. Un M-estimateur est utilisé pour gérer les données aberrantes. Le processus de minimisation opère simultanément sur l'erreur de reprojection des points qui utilise le principe d'alignement et l'erreur de reprojection qui se base sur la distance entre les contours projetés et contours image. La méthode décrite a été généralisée aux suivi d'objets de forme quelconques [Pressigout et Marchand, 2006].

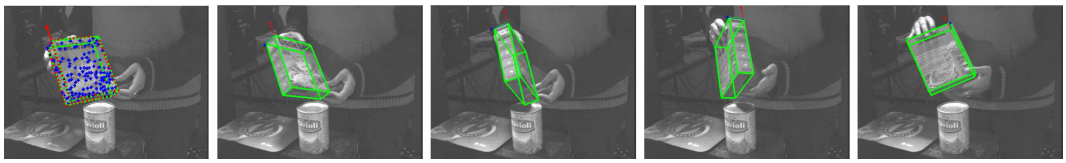


FIG. 2.11: Suivi obtenu avec l'approche hybride décrite dans [Pressigout et Marchand, 2006]

2.1.2 Approches sans connaissance *a priori*

La deuxième classe d'approche suppose que nous n'avons pas de connaissance sur l'environnement ou du moins nous pouvons disposer d'une connaissance partielle telle que les techniques appelées SFM (*Structure From Motion*) [Nistér, 2003] et SLAM (*Simultaneous Localization And Mapping*) [Mouragnon et al., 2006, Mei et Rives, 2007] utilisées en robotique mobile. Ces méthodes sont intéressantes car elles permettent de s'affranchir de l'utilisation des modèles 3D. En effet, ces approches permettent d'estimer, simultanément, la pose de la caméra et de reconstruire un modèle partiel de l'environnement. Ces méthodes sont de plus en plus prisées par les applications de RA car elles constituent une solution pour les applications évoluant essentiellement dans des environnements à grandes échelles. Ceci est du au fait qu'il est quasi impossible de disposer d'un modèle 3D précis et exhaustif. De telles approches permettent d'envisager d'utiliser des modèles 3D partiels de l'environnement et de les enrichir en ligne en reconstruisant les parties non-modélisées.

Par exemple, *G. Bleser* [Bleser, 2009] propose une méthode robuste qui ne nécessite pas une connaissance *a priori* complète de la scène. L'approche proposée combine une ap-

proche SFM, une approche SLAM et un suivi basé modèle. Tant que le modèle 3D partiel est visible, la pose est obtenue en re-projetant les lignes du modèle et en estimant les paramètres en alignant au mieux les contours projetés avec les contours extraits de l'image. Lors de la phase de suivi, les mouvements de la caméra ne sont pas restreints à l'observation de la contrepartie réelle du modèle. En effet, la structure de la scène est estimée automatiquement lors du suivi, ce qui permet de compléter le modèle. L'approche utilise un suivi de points d'intérêts basé sur une mise en correspondance d'imagettes. Pour gérer le changement d'échelle, les imagettes sont obtenues à partir d'une pyramide d'images. À partir de la triangulation, l'estimation de la structure 3D est raffinée récursivement en utilisant un filtre de Kalman étendu. Une mesure de qualité basée sur la contrainte de colinéarité est incorporée au système pour gérer l'influence des caractéristiques sur l'estimation de pose. En effet, si la pose estimée est correcte, la structure est raffinée. La méthode permet de gérer les occultations lors de la phase de suivi. Nous retrouvons aussi l'approche proposée par [Ababsa et al., 2008] qui combine deux types d'approches d'estimation de pose. En effet, les paramètres de la pose sont estimés en minimisant un critère qui combine une contrainte basée sur les points et une autre contrainte utilisant les segments de droites. A l'état initial, l'approche estime uniquement la pose à partir du modèle partiel existant de la scène. A partir de la seconde image, l'approche extrait de nouveaux points en utilisant l'opérateur de SIFT [Lowe, 2004]. Ces points sont appariés d'une image à une autre. L'ensemble des appariements obtenus est utilisé pour reconstruire leurs correspondants 3D à partir des poses estimées en utilisant le principe de la triangulation.

2.2 Discussion

Nous venons de voir le principe de quelques approches présentées dans la littérature et utilisées essentiellement dans des applications de RA même si certaines méthodes n'ont été testées qu'en intérieur. Cependant, elles peuvent présenter un potentiel pour des applications destinées à des environnements en extérieur. L'emploi d'une approche ou d'une autre dépend essentiellement du degré de connaissance que nous avons de l'environnement de travail. En effet, une connaissance partielle ou complète conduit nécessairement à utiliser des méthodes basées modèle qui sont précises et rapides étant donné que le processus se concentre sur la pose contrairement aux approches sans connaissance où le but est d'estimer simultanément la pose et la structure de la scène. Cependant, il est intéressant de combiner ces deux concepts, afin d'avoir d'un côté une estimation de pose précise et d'un autre côté reconstruire les parties de l'environnement non prises en compte dans le modèle. Ceci permet de les réutiliser dans l'estimation de pose lorsque l'utilisateur se trouve dans les zones correspondantes. Les méthodes utilisant les données naturelles extraites de la scène sont les plus adéquates et les plus privilégiées pour des environnements extérieurs. Le choix des primitives utilisées dépend du type de données prédominant dans la scène. Par exemple, si nous sommes en présence d'une scène texturées, nous privilégierons les méthodes basées textures. Si au contraire, nous observons dans la scène la présence de contours saillants ou de segments, il sera plus intéressant d'utiliser des méthodes basées contours. Les approches sans marqueurs se basant sur une connaissance *a priori* sont résumées dans le tableau ci-dessous (cf. tab.2.1). Ce dernier comprend un comparatif entre ces approches en s'intéressant aux avantages et inconvénients de chaque concept ainsi que les performances avancées par les différents auteurs (cf. section 2.1.1.2 page 35). Comme critères de performances, nous nous basons sur :

1. Le temps d'exécution ;
2. L'erreur moyenne en translation et en rotation ;

3. L'erreur de reprojection.

Approches	Avantages, Inconvénient et remarques	travaux	Performances
Contours/segments	⊕ Invariants globaux	Comport et al.	• Une erreur moyenne de (-0.1mm,0.9mm,-1.4mm) en translation avec σ=(1.1mm,0.8mm,1.3mm).
	⊕ Robuste aux occultations partielles des contours.		• Une erreur moyenne de (0.33°,-0.03°,-0.06°) en rotation avec un σ=(0.17°,0.12°,0.13°).
	⊕ Robuste aux variations de luminosité.		• Présente un temps d'exécution allant de 20ms à 66ms (pour des environnements en extérieur).
	⊖ Problème des contours visibles ou pas.	Reitmayr et al.	• Entre 0.5m et 3.5m de distance à une ligne de référence
	⊖ Le nombre des faux appariements croit avec les scènes très texturées.		• Présente un écart type de (0.09m,0.16m,0,14m) en position
	⊖ Effet de glissement pour les méthodes qui calculent le mouvement de la caméra pour mettre à jour la pose (Accumulation des erreurs)		• Pas d'indication sur l'orientation ni sur le recalage
	⊖ Les méthodes à une seule hypothèse supposent que le mouvement entre deux images successives est petit.	Klein et al.	• L'erreur présentée est entre 0.3 rad/s à 1.0 rad/s.
			• L'erreur de reprojection est entre 3 à 11 pixels.
			• Il présente un temps d'exécution pour le suivi visuel de 40ms
Points d'intérêts	⊕⊖ Utilise l'information de luminosité	Stricker et al.	• En rotation, une erreur de 0.9° a été obtenue sur des données de simulation
	⊖ Sensible aux variations de luminosité.		• En translation, la méthode donne 2 pixels de décalage sur les données de simulation.
	⊕ Reste assez robuste aux semi-ocultations (tout dépend de comment les points sont utilisés).		• 100ms en temps d'exécution sachant que les transformations de Fourrier sont précalculées.
	⊕⊖ Précision dépend de l'appariement 2D/3D ou de la pose initiale.	Bleser et al.	• Une erreur de 12.83mm en Translation.

	⊖ Les méthodes qui mettent à jour la pose à partir du mouvement obtenu avec les appariements sont sujettes à des dérives. ⊖ Certaines méthodes requièrent une base d'imagettes ou d'images de références. ⊕ Les méthodes utilisant des descripteurs sont robustes face aux variations de luminosité ⊖ Mais elles sont très consommatrices en temps de calcul		• entre 0.06° et 0.09° en Rotation. • Un temps de calcul de $28.84ms$
Plans	⊕ Assez Robuste aux occultations partielles (estimation même si une partie du plan est occultée). ⊕ Simple à mettre en œuvre. ⊖ Problème d'indentification des plans. ⊖ Sensible aux variations de luminosité (Appariement avec corrélation) mais peut être résolu par l'utilisation de descripteurs sous réserve du temps de calcul. ⊖ Problème de dérives dues à l'accumulation des erreurs de calcul d'homographie.	Simon et al.	• 2.5 pixels en erreur de reprojection • Pas d'indication sur les performances en translation et en rotation.
Textures+Contours	⊕ Robuste et précise pour les scènes texturées et non texturées ⊕ Englobe les avantages de chaque concept ⊖ Risque de glissement si l'approche se base sur un calcul de mouvement pour mettre à jour la pose ⊖ Influence des données aberrantes dans l'estimation.	Pressigout et al.	• Suivant le résultat d'asservissement, une erreur de $(-10.5mm, -2.4mm, -0.2mm)$ en translation • et une erreur de $(-0.2, 0.2mm, 1.8mm)$ en rotation. • 76ms en temps d'exécution.

TAB. 2.1: Comparatif des méthodes sans marqueurs avec connaissance *a priori*

Les performances que nous présentons pour chaque méthode ont été généralement obtenues soit à partir d'expérimentations réalisées à petite échelle ou à partir de données de simulation. En général, nous observons des performances qui se rejoignent. En effet, l'erreur en translation présentée est souvent de l'ordre du millimètre. Quant à la rotation, les méthodes obtiennent des erreurs qui sont inférieures à 1°. En termes d'erreur de recalage, qui reste la performance principale pour un système de réalité augmentée, certaines méthodes présentent une erreur allant jusqu'à 10 pixels ce qui reste raisonnable.

Cependant, ces résultats ne peuvent refléter le comportement de ces approches dans des environnements larges. Des méthodes citées auparavant, nous retrouvons la méthode utilisée par [Reitmayr et Drummond, 2006] qui se base sur l'approche décrite dans les travaux de [Klein et Drummond, 2003]. Cependant, nous ne disposons pas de performances établies à cette échelle. Les erreurs fournies concernent la position dans l'environnement. Cette erreur est calculée à partir d'une distance à une ligne de référence. Pour ce qui est de l'approche proposée dans les travaux de [Stricker et Kettenbach, 2001], les résultats présentés ont été obtenus en effectuant différentes rotations sur une image de référence ainsi que des translations. Concernant l'approche de [Simon et al., 2000], la seule information de performance dont nous disposons et celle de l'erreur de reprojection qui représente le seuil choisi pour l'algorithme de RANSAC lors du calcul de l'homographie.

Comme le fait ressortir notre tableau de comparaison, il n'existe pas de méthode parfaite ou optimale. Chaque méthode est adéquate à son cas d'utilisation, à l'environnement dans lequel elle est déployée et essentiellement au type d'informations disponibles sur la scène. Nous supposons que le modèle 3D de la scène est disponible. Ceci nous amènera logiquement à nous orienter vers des méthodes avec connaissance *a priori* sur l'environnement. Ensuite, il faut choisir le type de primitives à utiliser pour le calcul de pose. Ceci dépend essentiellement du type d'information pertinente dans la scène. En effet, nos environnements de tests, comme nous les verrons ultérieurement, sont assez texturés en plus, certains points d'intérêts pertinents peuvent être extraits facilement.

Le fait de disposer de modèles assez précis, et le fait d'évoluer dans des environnements qui présentent un nombre important de points d'intérêts (comme des coins), nous optons pour une approche utilisant des points. De plus, ceci nous permet d'explorer une méthode de type *PnP* (Perspectives *n* Points) qui calcule les paramètres de la pose en utilisant le modèle de projection de la caméra soit de manière analytique ou par optimisation. Ce type de méthode fournit de bonnes estimations à condition d'avoir des bons appariements 2D/3D en entrées ainsi qu'un nombre suffisant de couples d'appariements pour le calcul. L'utilisation d'une telle approche permet de réduire l'effet de dérive engendré par certaines méthodes qui se basent sur le calcul de mouvement entre deux images successives pour mettre à jour la pose obtenue à l'instant précédent.

La méthode doit répondre à certains critères à savoir :
- Précision de l'estimation des paramètres de pose ;
- Ne pas engendrer de dérive ;
- Assez robuste face à des mouvements larges, à certaines variations de luminosité ;
- Temps d'exécution réduit.

La méthode que nous mettons en œuvre s'inspire des approches basées marqueurs souvent utilisées par la communauté de la RA. Ce qui est décrit pas la suite représente une adéquation d'une méthode développée au sein du laboratoire mais en utilisant des données naturelles. En effet, nous retrouvons certains concepts de cette méthode basée marqueurs

décrite dans les travaux de [Ababsa et Mallem, 2004] [Didier et al., 2008].

2.3 Méthode basée point d'intérêts : vue globale

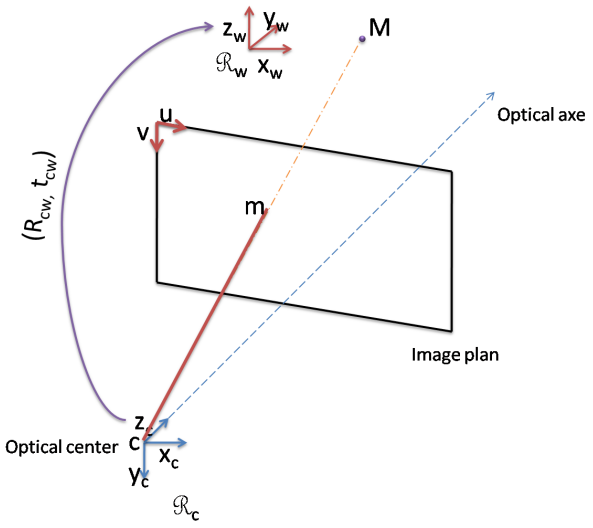


FIG. 2.12: Modèle de caméra sténopé

Les méthodes basées vision utilisent le flux vidéo pour estimer la position et l'orientation de la caméra dans le repère de travail. La pose de la caméra représente la relation qui transforme le système de coordonnées associé au monde $\mathscr{R}_W$ au repère associé à la caméra $\mathscr{R}_C$. Celle-ci permet de former une image à partir du modèle de projection perspective appelé sténopé (cf. annexe.A) présentée dans la figure 2.12. Soit $M_i = (X_i, Y_i, Z_i)^T$, $i = 1..n$, $n \geq 3$ un ensemble de points définis dans le repère $\mathscr{R}_W$, dont leurs coordonnées dans le repère caméra est données par $M_i^c = (X_i^c, Y_i^c, Z_i^c)^T$ sont données par la relation :

$$M_i^C = R_{CW} M_i + t_{CW} \tag{2.1}$$

Où $R_{CW} = (r_1^T, r_2^T, r_3^T)$ et $t_{CW} = (t_x, t_y, t_z)^T$ représentent respectivement la matrice de rotation et le vecteur de translation qui définissent la relation entre le repère associé au monde, noté $\mathscr{R}_W$, et le repère associé à la caméra $\mathscr{R}_C$.

Si $m_i = (u_i, v_i)$ est la projection image du point M_i sur un plan normalisé, la relation entre m_i et M_i est donnée par :

$$u_i = \frac{r_1^T M_i + t_x}{r_3^T M_i + t_z} \qquad v_i = \frac{r_2^T M_i + t_y}{r_3^T M_i + t_z} \tag{2.2}$$

Ce qui donne :

$$m_i = \frac{1}{r_3^T M_i + t_z}(RM_i + t) \tag{2.3}$$

Cette équation est connue en tant qu'équation de colinéarité. Au final, l'estimation de la pose est vue comme une minimisation d'erreur de reprojection entre les points 2D m_i extraits des images et la projection des points 3D M_i qui sont leurs correspondants dans le

monde réel. Ceci forme un ensemble d'appariements 2D/3D. L'erreur de reprojection se traduit mathématiquement de la manière suivante :

$$E(R_{CW}, t_{CW}) = \sum_i \| m_i - \frac{R_{CW} M_i + t_{CW}}{r_3^T M_i + t_z} \|^2 \tag{2.4}$$

Il existe plusieurs algorithmes pour minimiser le critère de l'équation 2.4. Nous citons la méthode de Gauss-Newton ou bien la méthode Levenberg-Maquardt. En se basant sur l'étude présentée dans [Didier et al., 2008], nous optons pour l'algorithme de l'itération orthogonale (IO) [Lu et al., 2000] en raison de sa précision et de sa rapidité de convergence. Un bref descriptif de l'algortihme est donné en annexe A.

A partir du principe que nous venons d'énoncer, pour estimer la pose de la caméra en utilisant les points, nous devons retrouver les appariements 2D/3D. Cependant, si nous retrouvons des méthodes pour apparier des points 2D entre deux images, nous ne disposons pas d'approches qui réalisent un appariement entre les points 2D extraits d'une image et des points 3D obtenus du modèle comme c'est le cas pour les méthodes basées marqueurs. En effet, ces méthodes identifient de manière automatique les couples d'appariements. Ceci est obtenu en reconnaissant les marqueurs visibles de la scène et en associant à chaque coin ou point des marqueurs 2D son correspondant dans le monde réel. Cependant, ceci n'est pas tout à fait pareil en ce qui concerne les méthodes sans marqueurs.

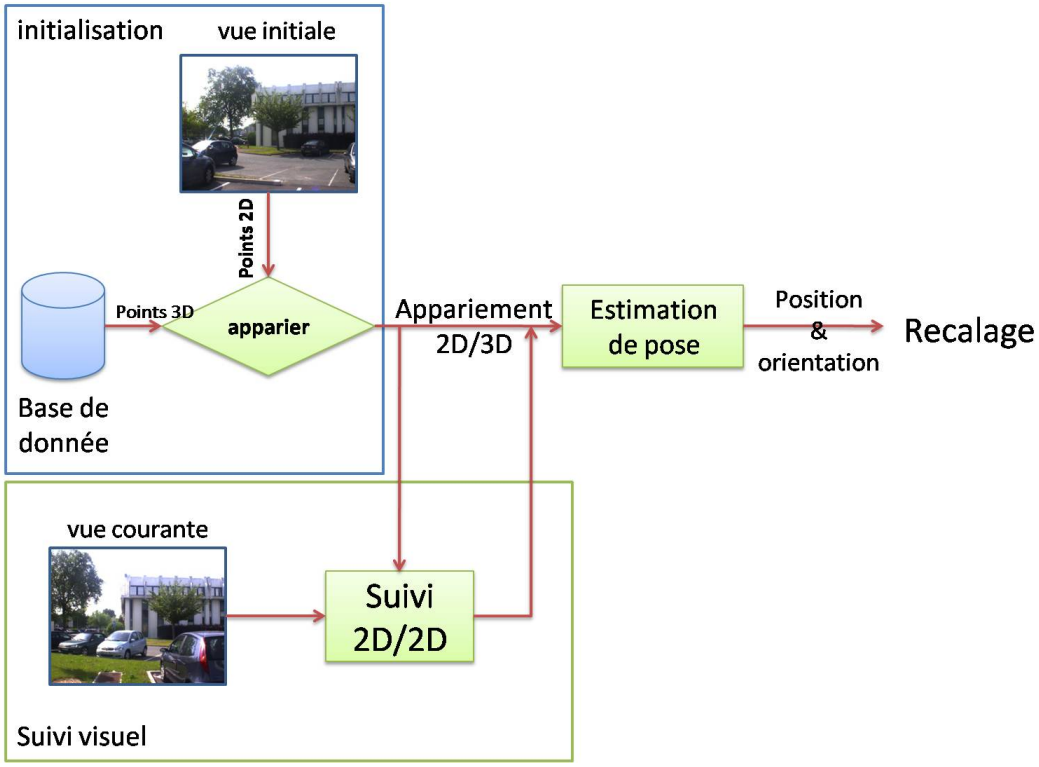


FIG. 2.13: Principe général de fonctionnement en utilisant une approche basée points.

Dans un premier temps, nous allons nous intéresser au point une méthode d'appariement 2D/3D en utilisant des points. L'idée est de faire correspondre des points 3D du modèle jugés pertinent et qui peuvent être identifiés facilement dans les images. Cependant, cette phase d'appariement ne se fera que lors d'une étape d'initialisation. En effet, cette

phase peut être couteuse en temps de calcul. Toutefois, cet appariement doit être maintenu dans le flux vidéo pour ainsi pouvoir estimer la pose. Cette phase d'initialisation est suivie d'une phase de suivi 2D. Le suivi visuel nous permettra de retrouver les positions des points 2D, identifié initialement comme la projection des points 3D du modèle, dans chaque nouvelle image du flux vidéo. En retrouvant la position de ces points dans chaque image, cela nous permettra de manière indirecte d'obtenir les appariements 2D/3D. Ayant l'appariement 2D/3D à l'instant t, et en suivant les correspondant des points 3D dans l'image $t+1$, nous pourrons retrouver par transitivité l'appariement 2D/3D dans l'image courante. La figure 2.13 illustre le principe de l'approche que nous venons de décrire.

Dans ce qui suivra, nous allons présenter notre méthode d'appariement 2D/3D puis nous nous intéresserons au suivi visuel.

2.4 Initialisation semi-automatique : Appariement 2D/3D

Comme nous l'avons vu, pour calculer la pose, il nous faut mettre en correspondance des données 3D avec leurs projections 2D. Cette tâche est cruciale car chaque erreur introduite influe sur le calcul de la pose. Cependant, la procédure d'appariement diffère suivant le type de données utilisées. Si nous prenons par exemple les contours (ou segments), la phase d'appariement consiste à les projeter dans le repère image avec un point de vue prédéfini. Il suffit alors de chercher dans l'image le correspondant de chaque point extrait du contour projeté qui correspond généralement au maximum de la norme du gradient. Cette recherche est faite dans la direction de la normale ce qui revient à faire une recherche 1D. Cependant, lorsque nous utilisons des points, la procédure est différente. Il ne suffit pas juste de projeter les points et de chercher les correspondants. En effet, les approches d'appariement basées points ont besoin d'informations supplémentaires associées à chaque point 3D.

Ces approches d'appariements dépendent du type d'informations additionnelles utilisées. Nous retrouvons certaines approches qui utilisent des imagettes. Ces imagettes sont définies autour des projections des points 3D dans des images de référence. L'appariement peut être obtenu en alignant ces imagettes avec l'image courante. Une fois ceci fait, le correspondant du point 3D dans l'image courante est celui qui correspond au barycentre de l'imagette. Cette approche a été utilisée par exemple dans les travaux de G. Bleser [Bleser, 2009]. Aussi, des approches de type corrélation peuvent aussi être utilisées. Cependant que ce soit l'alignement ou bien la corrélation, elles dépendent fortement de la luminosité. De ce fait, s'il existe une grande variation de luminosité entre les imagettes de référence et l'image courante, l'appariement ne donne pas de bon résultat. Pour compenser ceci, d'autres approches utilisent des descripteurs qui sont invariants à la luminosité telle que les SIFT [Lowe, 2004]. Cependant ces approches sont lourdes en temps de calcul. Nous trouvons aussi des approches qui utilisent la reconnaissance d'objet dont on connait la position 3D dans l'environnement. Ceci est le cas pour les approches basées marqueurs ou dans le système proposé par [Zollner et al., 2008]. Dans ces approches, le système doit cependant disposer d'une base de données assez complète des objets existants dans la scène. Ceci est difficile à obtenir dans des environnements à grandes échelles.

Ce qui nous intéresse est d'effectuer l'appariement lors d'une phase d'initialisation. Les approches d'initialisation sont variées et peuvent être classifiées selon le degré d'intervention de l'utilisateur, entre méthodes manuelles où l'utilisateur effectue l'appariement

par lui même, méthodes semi-automatiques où l'utilisateur intervient le moins possible uniquement pour guider l'initialisation et méthodes automatiques où l'appariement est effectué sans intervention de l'utilisateur. En ce qui nous concerne, nous proposons une procédure d'initialisation qui nécessite l'intervention de l'utilisateur. Mais cette intervention sera restreinte juste pour guider la phase d'appariement. De ce fait, la procédure semi-automatique se décompose en deux phases : une phase d'alignement manuelle et une phase d'appariement.

Lors de la phase manuelle, l'utilisateur devra aligner manuellement un rendu du modèle filaire représentant l'environnement (cf. fig.2.14-a) obtenu à partir d'une pose prédéfinie. Une fois cet alignement effectué (cf. fig.2.14-b), l'utilisateur le valide pour rendre la main au système afin d'effectuer la phase d'appariement. Après validation, le système apparie les points caractéristiques extrait du modèle avec les points de l'image courante. Le fait d'aligner le rendu du modèle filaire permet de localiser approximativement les projections des points 3D dans la vue courante. La phase d'appariement consiste à raffiner cette localisation.

(a) (b)

FIG. 2.14: (a) Rendu du modèle filaire (b) Alignement manuel du rendu avec la vue courante

Pour effectuer cet appariement, nous associons des informations à chaque point 3D. Dans notre cas, nous optons pour des descripteurs au lieu d'utiliser des imagettes afin d'avoir un appariement robuste aux variations de luminosité. Nous utilisons les descripteurs SURF [Bay et al., 2008] qui décrivent la répartition des intensités au sein d'une échelle dépendant du voisinage des points. L'appariement s'effectue de la manière suivante. Une zone de recherche est définie autour de chaque point 3D projeté afin de chercher son correspondant. Pour cela, l'approche extrait des points d'intérêts dans cette zone (cf. fig.2.15). Cependant pour cette approche d'initialisation, nous n'utilisons pas le détecteur proposé par l'approche SURF car les points pertinents pour ce détecteur ne correspondent pas aux points pertinents du modèle 3D. Etant donné que les points extraits du modèle correspondent à des coins, nous faisons appel au détecteur de Harris [Harris, 1993] qui permet d'extraire les coins pour lesquels nous calculons leur descripteur SURF. Une fois ceci fait, il ne reste qu'apparier ces coins extraits avec les points 3D projetés. L'appariement consiste tout simplement à calculer la distance entre les descripteurs. Les paires d'appariements correspondent aux couples ayant la plus petite distance. Cependant, l'appariement peut engendrer des données aberrantes (*outliers*) qui peuvent être éliminées en utilisant l'algorithme

RANSAC [Fischler et Bolles, 1981] (cf. annexe.A).

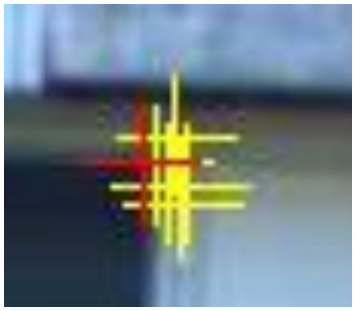


FIG. 2.15: ensemble de points extraits (en jaune) autour de la projection d'un point 3D (en rouge)

Pour valider l'appariement effectué, nous comparons la pose calculée à partir des appariements 2D/3D avec la pose prédéfinie utilisée pour le rendu du modèle filaire. Pour cela, nous supposons que la différence entre la pose utilisée pour le rendu et la pose calculé à partir des appariements varie peu. Soit la matrice des paramètres de la pose prédéfinie P_{pre} et P la matrice de la pose calculée. Si les deux matrices P_{pre} et P sont identiques, nous avons $P_{pre} = P \Leftrightarrow P_{pre}P^{-1} = I$. Avec la supposition que le mouvement est faible, P_{pre} et P vérifient que :

$$P_{pre}P^{-1} \approx I \tag{2.5}$$

En calculant la trace de $P_{pre}P^{-1}$, si le mouvement est faible, la trace du produit est non seulement proche de $trace(I) = 4$ mais également toujours inférieure ou égale à cette valeur. Nous pouvons alors fixer un seuil minimal auquel la trace doit être supérieure ou égale. D'après l'étude faite dans [Gagnières, 2006], le seuil peut être définit autour de 3.95 ce qui correspond à une variation d'angle de 12°.

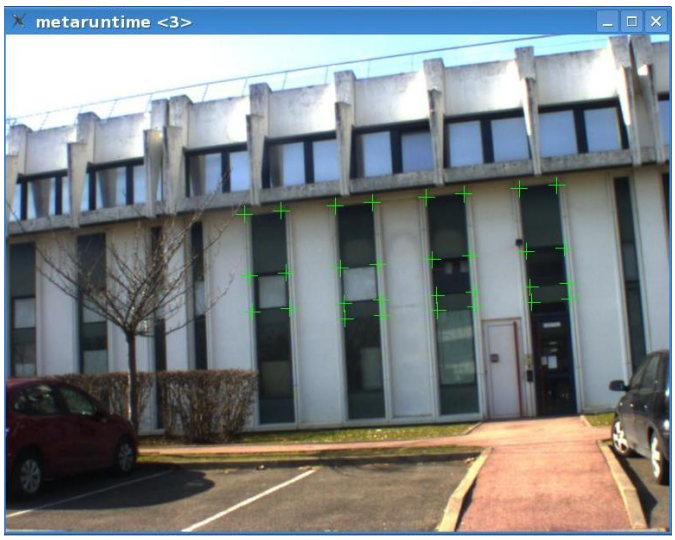


FIG. 2.16: Résultat de l'appariement : correspondant 2D des points 3D visibles du modèle

Même si les descripteurs SURF sont associé à un ensemble de points non pertinents par rapport au détecteur SURF, mais il restent suffisants pour cette phase d'initialisation où l'alignement manuel permet de restreindre la zone de recherche (cf. fig.2.16). Nous allons

voir maintenant comment l'appariement 2D/3D obtenu lors de cette phase est maintenu au fil du flux vidéo.

2.5 Suivi basé points

Une fois les appariements 2D/3D obtenu, la seconde phase est de les maintenir tout au long du flux vidéo. Pour cela, nous utiliserons une approche de suivi visuel des points 2D qui représentent les projections des points 3D retrouvés dans l'image initiale. Dans cette section, nous allons décrire la méthode de suivi que nous avons utilisé dans notre approche.

Au départ, nous avons pensé tout simplement à une corrélation entre les projections des points 3D et des points 2D extraits de l'image courante. Cependant, ce type d'approche est très consommatrice en temps de calcul et est très sensible aux variations de luminosité, aux mouvements larges ce qui engendre beaucoup de faux appariements. Nous pourrions pallier ceci en utilisant des méthodes invariantes mais elles restent toutefois consommatrices en temps de calcul. Parmi les méthodes les plus utilisées, nous retrouvons la méthode de KLT [Lucas et Kanade, 1981] (Kanade-Lucas-Tomasi Tracker) qui repose sur le principe du flot optique. Cette méthode à l'avantage de fonctionner en temps réel. Voici un bref descriptif de cette approche.

L'approche suppose que deux images prises entre deux instants proches sont généralement fortement liés. Ceci est du au fait que les images représentent la même scène prises sous deux points de vue voisins. Par le biais d'hypothèses simplificatrices, ceci peut s'approximer par le fait qu'une zone de l'image courante n'est le résultat que d'un mouvement image d'une zone de l'image précédente. En supposant que le mouvement inter-images est petit, celui-ci peut être approximé par une translation. Si on définit $d = (d_u, d_v)$ le déplacement d'un pixel p de coordonnées (u, v) dans le repère d'image I_t et d'intensité $I_t(u, v)$, dans l'image courante I_{t+1}, nous obtenons la relation suivante :

$$I_t(u, v) = I_{t+1}(u + d_u, v + d_v) \tag{2.6}$$

d représente également la vitesse de l'image au point p. A partir de ceci, un pixel p défini dans une image I_t peut être suivi dans l'image I_{t+1} en retrouvant son déplacement. Cependant, la valeur de l'intensité du pixel peut varier d'une image à une autre ou être confondue avec un pixel adjacent qui a une intensité proche. Pour pallier cela, au lieu de suivre un pixel tout seul, la méthode se base sur le suivi d'un ensemble de pixels définis à partir d'une zone autour du pixel candidat. Cette zone définie par une imagette se déplace suivant le mouvement d. De là, pour estimer ce mouvement, il suffit de trouver le meilleur déplacement qui minimise la différence entre l'imagette initiale définie dans l'image I_t et celle obtenue dans l'image I_{t+1}. Cette différence est exprimée par l'erreur suivante :

$$\varepsilon(d) = \varepsilon(dx, dy) = \sum_{x=u-w}^{u+w} \sum_{y=v-h}^{v+h} (I_t(x, y) - I_{t+1}(x + d_u, y + d_v)) \tag{2.7}$$

En général, le mouvement d'une image à une autre peut être modélisé par une transformation affine f telle que :

$$\varepsilon(d) = \sum_{x=u-w}^{u+w} \sum_{y=v-h}^{v+h} (I_t(x, y) - I_{t+1}(f(x, y))) \tag{2.8}$$

Les paramètres de cette transformation affine sont estimés en minimisant cette erreur. En retrouvant les paramètres de la transformation qui aligne les deux imagettes, ceci permet

de suivre le point d'une image à une autre.
Afin d'accélérer les calculs, la méthode se décline sous une version hiérarchique. Cette version de l'approche suit le point dans un niveau grossier d'une pyramide d'image. Puis, le résultat est propagé du niveau le plus grossier de la pyramide au niveau le plus fin jusqu'à retrouver la position du point suivi dans l'image initiale.

2.6 Expérimentations et résultats

Nous présentons dans cette partie les résultats que nous avons obtenus à partir des expérimentations menées sur cette approche de localisation basée vision. Ces expérimentations ont pour but de caractériser l'efficacité et la précision des différents modules à savoir l'initialisation et le calcul de pose. Pour cela nous avons défini un ensemble de critères de performance résumés comme suit :

1. Le temps d'exécution ;
2. L'erreur de reprojection : représente l'écart entre les points 2D identifiés de l'image et la projection de leurs correspondants 3D avec les paramètres de pose obtenus ;
3. L'erreur de généralisation : représente l'écart entre la projection des points 3D non pris en compte dans le calcul de pose et de leur correspondants 2D ;
4. L'erreur de localisation : consiste à estimer la précision de l'estimation de la localisation en les comparant à des données réelles ;

Pour information, pour nos tests nous avons utilisé une caméra industrielle USB uEye UI-2220RE avec une distance focale de 8*mm*. La caméra est calibrée en utilisant la méthode de Faugeras et Toscani [Faugeras et Toscani, 1987]. Les paramètres intrinsèques sont donnés dans la table 2.2 :

Taille de l'image en pixels		768x576	
Paramètres de projection		Paramètres de distortion	
Facteurs d'échelle		Coefficients de distortion radial	
α_u	-985.19821	κ_1	-0.34326
α_v	992.33987	κ_2	1.29673
Centre de projection optique		Coefficients de distortion Tangentielles	
u_0	403.08131	p_1	0.00058
v_0	279.27419 s	p_2	0.00033

TAB. 2.2: Paramètres de calibration de la uEye UI-2220R

2.6.1 Performances de l'initialisation semi-automatique

Etant donné que la phase d'initialisation est importante, nous avons quantifié la précision de cette procédure. En effet, la précision de l'estimation de la pose dépend étroitement de la précision de l'initialisation. Pour cela, nous réalisons plusieurs phases d'initialisation avec différents point de vue. Nous retrouvons dans la figure 2.17 quelques résultats obtenus. Ceux-ci représentent les points extraits avec l'opérateur de Harris [Harris, 1993] (en jaune) autour des points 3D projetés et alignés sur la vue réelle (en rouge) (cf. fig.2.17-a). Nous retrouvons aussi le résultats de l'appariement (en jaune) après correction de l'appariement avec RANSAC (point en rouge) (cf. fig.2.17-b). Nous pouvons constater que l'approche a fournie des résultats satisfaisant même en situation de variation de luminosité.

(a)

(b)

FIG. 2.17: Résultats d'appariement 2D/3D : (a) Projection des points 3D (en rouge) avec les coins extraits dans la zone de recherche (en jaune) (b) résultats de l'appariement avec le SURF (rouge) et résultats aprés élimination des données aberrantes (en jaune)

Nous allons nous intéresser maintenant à l'erreur obtenue par l'appariement. Pour cela, nous calculons la distance moyenne entre les points 2D obtenus à l'issue de la phase d'initialisation semi-automatique et les points 2D extraits des images. Ceci nous donne une erreur moyenne égale à 3.8216 pixels avec un écart type de l'ordre 1.0873 pixels. Ce résultat est jugé satisfaisant et permet d'avoir des appariements 2D/3D corrects. Ceci permet d'avoir une bonne estimation de la pose. En ce qui concerne le temps d'exécution, nous reprenons dans le tableau 2.3 le détail des temps réparties pour chaque phase de l'initialisation :

Etape	Temps en moyenne
Alignement manuel	non quantifiable
Extraction des points et appariement	50 ms
RANSAC	200 ms
Total	250 ms (sans l'alignement manuel)

TAB. 2.3: Temps de calcul moyen par phase pour l'initialisation

Certes les temps de calcul présentés sont élevés, mais ceci ne pose pas de problème étant donné que cette phase n'est réalisée qu'à l'initialisation et qu'elle n'est pas réitérée lors de la phase de suivi.

2.6.2 Performances de l'estimation de pose

Dans ce qui suit, nous allons nous focaliser sur les performances de l'approche d'estimation de pose présentée dans ce chapitre. Ces performances concernent celles du suivi visuel et du calcul des paramètres de pose. Nous avons établi différents protocoles expérimentaux correspondant à chacun des critères décrit auparavant. Voici un descriptif de

ces expérimentations ainsi que des résultats obtenus. A noter que ces résultats sont indépendants des performances de la phase d'initialisation. Dans ce qui suit, les résultats sont obtenus à partir d'une initialisation manuelle.

2.6.2.1 Erreur de reprojection

L'erreur de reprojection est un critère important pour les systèmes de réalité augmentée car elle reflète la qualité du recalage obtenu. Pour quantifier cette erreur, l'expérimentation conduite consiste à se déplacer dans un environnement et de calculer les paramètres de la pose. Pour chaque pose obtenue, nous calculons l'erreur de reprojection qui représente la distance entre les points 3D projetés et les points 2D extraits de l'image. Ces points 2D sont définis initialement de manière manuelle, puis ils sont suivis dans le flux image avec l'algorithme KLT. Cette erreur est donnée par l'équation :

$$\varepsilon = \sum_{i=1}^{n} \|m_i - Proj(M_i, R, t)\| \tag{2.9}$$

Où (m_i, M_i) sont le couple d'appariement 2D/3D et (R,t) la pose estimée. Sur un ensemble de plus de 1200 images, la figure 2.18 présente sur le tracé en haut les erreurs moyennes obtenues par image. Quant au tracé en bas, il représente l'erreur moyenne calculée en fonction du nombre de points utilisés pour estimer les paramètres de pose.

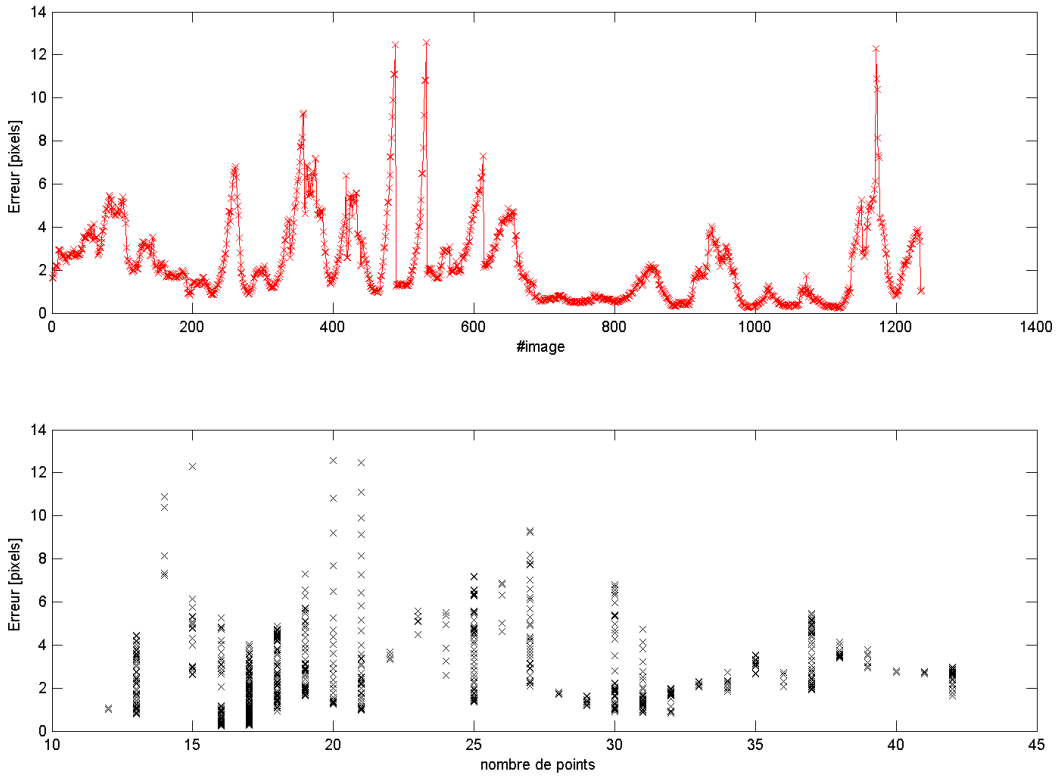


FIG. 2.18: Tracé des erreurs de reprojection

Visuellement, les erreurs sont en général inférieures à 6 pixels. En observanr le tracé des erreurs en fonction du nombre de points utilisés, nous constatons que les erreurs maximales sont obtenus avec un nombre inférieur à 20 points. Ceci est logique vu que plus

nous avons de points plus la pose estimée est précise. En plus, ceci dépend aussi de la répartition des points dans la scène. Nous retrouvons dans le tableau 2.4 quelques résultats numériques présentés par l'erreur moyenne, l'écart type et le 3^{eme} quartile qui correspond à 75% des données (erreurs). Les résultats que nous avons obtenus sont équivalents à celles présentés par d'autres travaux. Cette erreur reflète approximativement le seuil défini dans l'algorithme RANSAC pour éliminer les données aberrantes.

Moyenne	Ecart Type	3^{eme} quartile
2.3108	1.8579	3.1183

TAB. 2.4: Erreurs de reprojection

Nous avons généralisé l'erreur de reprojection en calculant l'erreur de reprojection sur un ensemble de points qui n'est pas utilisé pour le calcul des paramètres de pose. Nous reprenons la même expérimentation que précédemment où nous nous déplaçons dans un environnement et estimons les paramètres de pose. Cependant, les poses sont calculées à partir d'un sous ensemble des points suivis. Une fois la pose estimée, nous calculons l'erreur de reprojection sur le second ensemble de points. Les résultats obtenus sont donnés dans le tableau 2.5.

Erreur Moyenne	Ecart Type	3^{eme} quartile
6.9630	11.3776	7.7295

TAB. 2.5: Erreurs généralisées

En moyenne l'erreur obtenue est assez satisfaisante. En observant le 3^{eme} quartile (75% des erreurs), les erreurs calculées sont autour de l'erreur moyenne avec une différence de 1 pixel pour l'erreur maximale (l'erreur minimale est égale à 0.0621 pixel).

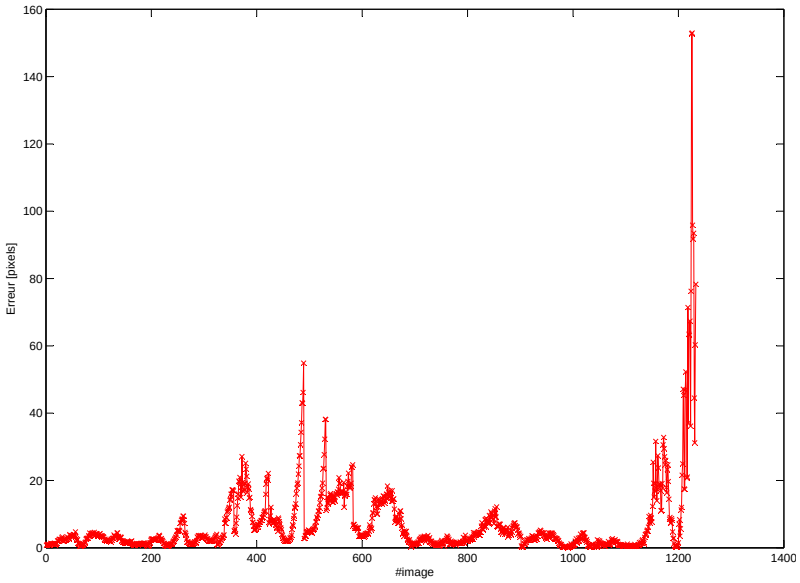


FIG. 2.19: Tracé des erreurs de généralisation

En analysant les erreurs obtenus sur la séquence, nous constatons que vers la fin cette erreur de généralisation croit (cf. fig.2.19). Pour indication, elle varie entre 30 et 150. Ceci est du à la précision de la pose qui dépend du nombre de points utilisés pour l'estimation ainsi que de leur répartition dans l'image. Si les points sont dans un voisinage proche des points utilisés dans le calcul, l'erreur de reprojection sera réduite.

2.6.2.2 Erreur de localisation : Position

Jusque là, nous nous sommes intéressés aux performances de notre méthode d'un point de vue recalage. Dans les expérimentations qui vont suivre, nous allons quantifier les performances de notre méthode d'un point de vue localisation. Nous allons commencer par la position. Pour cela, nous avons établie deux protocoles différents. Le premier protocole, que nous appelons test de "**la ligne droite**", consiste à évoluer le long d'une droite et de mesurer la position en utilisant la vision. La ligne droite est utilisée comme donnée de référence, elle est échantillonnée en plusieurs points dont les positions par rapport au repère monde sont connues. Celles-ci sont mesurés avec un télémètre laser ayant une précision de +/-15cm. En chaque point de la droite, nous calculons la position avec la vision. Dans la figure 2.20, nous représentons la trajectoire obtenue avec la vision (en ligne bleu) comparée avec les positions de référence (en ligne rouge).

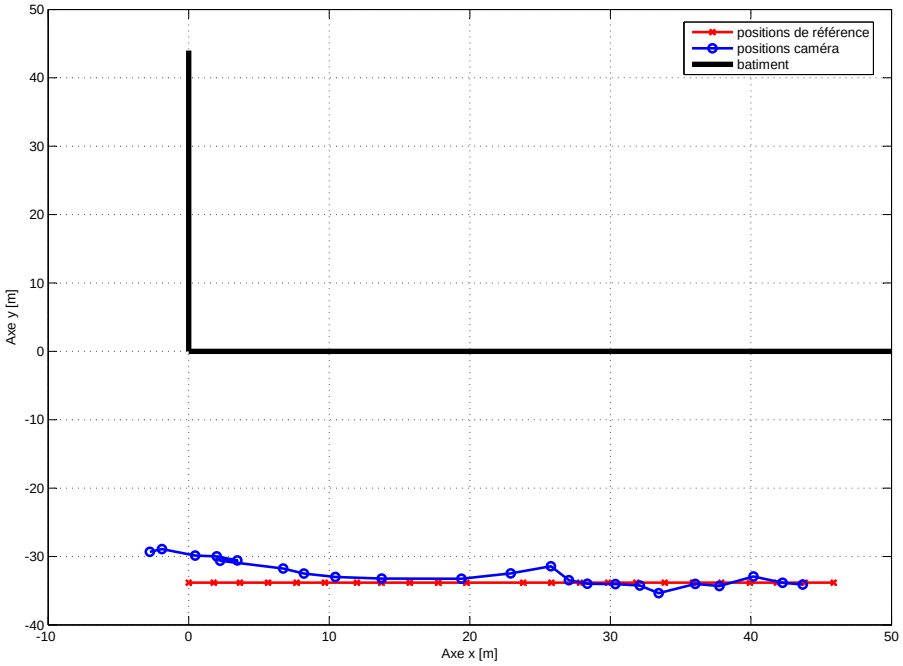


FIG. 2.20: Positions caméra (ligne bleu) vs. positions de référence (ligne rouge) le long d'une droite

Pour consolider les résultats obtenus sur la ligne droite, nous faisons une autre expérimentation, mais cette fois-ci les positions sont choisies de manière aléatoire dans l'environnement. Après avoir calculé les positions réelles, nous calculons les positions avec la vision et nous les comparons aux données de références. La figure 2.21 présente les différentes positions obtenues avec l'approche basée vision (en bleu) comparées aux positions de référence (en rouge).

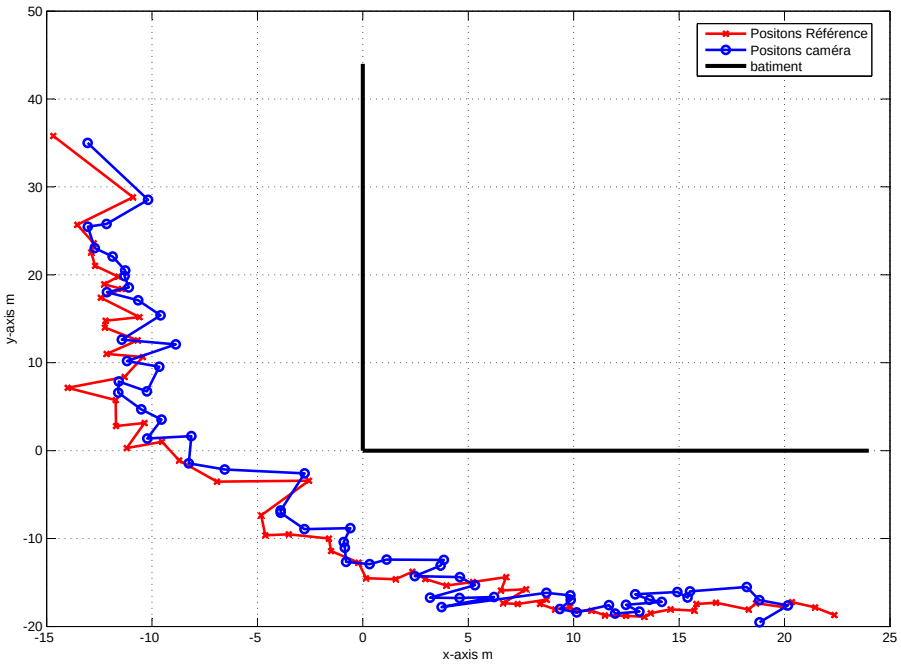


FIG. 2.21: Positions caméra (en bleu) vs. positions de référence (en rouge) le long d'une droite

Nous reprenons dans le tableau qui suit (cf. tab.2.6) les erreurs moyennes et les écarts types obtenus sur chaque axe pour chaque protocole.

		Erreur Moyenne	Ecart type
Ligne droite	Axe X	2.770	1.867
	Axe Y	1.867	1.554
Poisition aléatoire	Axe X	1.619	0.98
	Axe Y	1.288	0.611

TAB. 2.6: Localisation basée vision : Erreurs moyennes de position et écart types

Ces résultats sont jugés satisfaisants au vue de la taille de l'environnement où nous évoluons. Par exemple, en ce qui concerne le protocole de "**la ligne droite**", nous sommes à une distance de 33.814m de l'origine du repère monde sur l'axe Y. De plus, notre droite s'étale sur une distance de 48.115m. Nous ne pouvons pas obtenir des résultats de l'ordre du centimètre comme c'est le cas des méthodes testées en milieu à petit échelle. Mais si nous comparons avec ce que présente [Reitmayr et Drummond, 2006] par exemple, où il présente une erreur entre 0.5m et 3.5m par rapport à une droite, nos résultats sont assez encourageants.

2.6.2.3 Erreur de localisation : Orientation

Nous allons maintenant nous intéresser aux performances de la méthode point de vue orientation. Pour cela, nous générons trois séquences à partir d'image réelles prises de notre environnement. Ces séquences sont générées en réalisant des mouvements rotationnels sur chaque axe du repère de la caméra et appliquée sur l'image. Ceci nous permet d'avoir des

orientations de références à comparer avec celles estimées avec l'approche de vision décrite dans ce chapitre. A partir des orientations calculées et les orientations de références, nous calculons les erreurs sur chaque axe. Nous visualisons ces erreurs dans les tracés présentés dans les figures fig.2.22 pour les rotations autour de X, fig.2.23 pour les rotations autour de Y et fig.2.24 pour les rotations autour de Z.

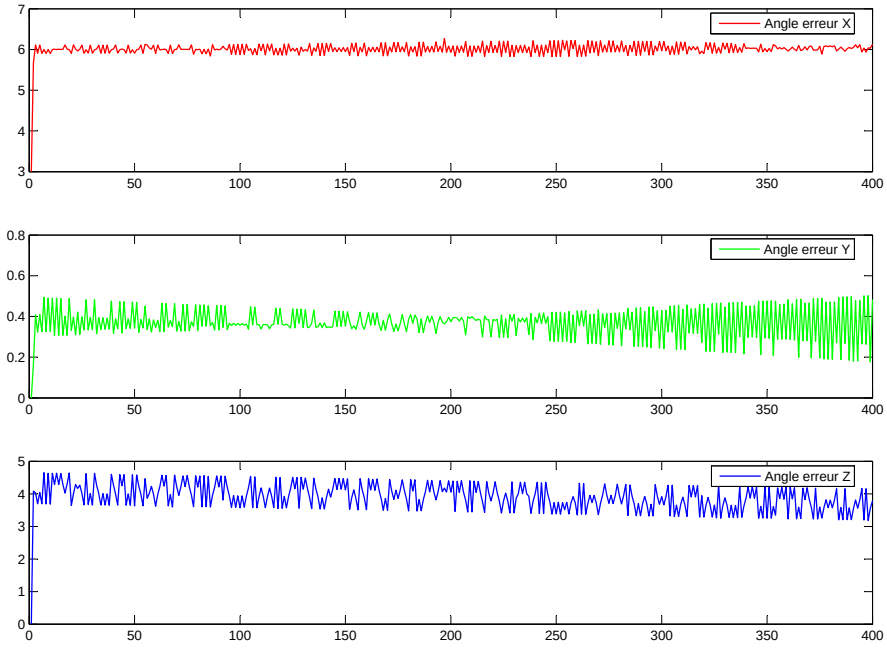


FIG. 2.22: Erreurs d'orientation avec des rotations autour de l'axe X

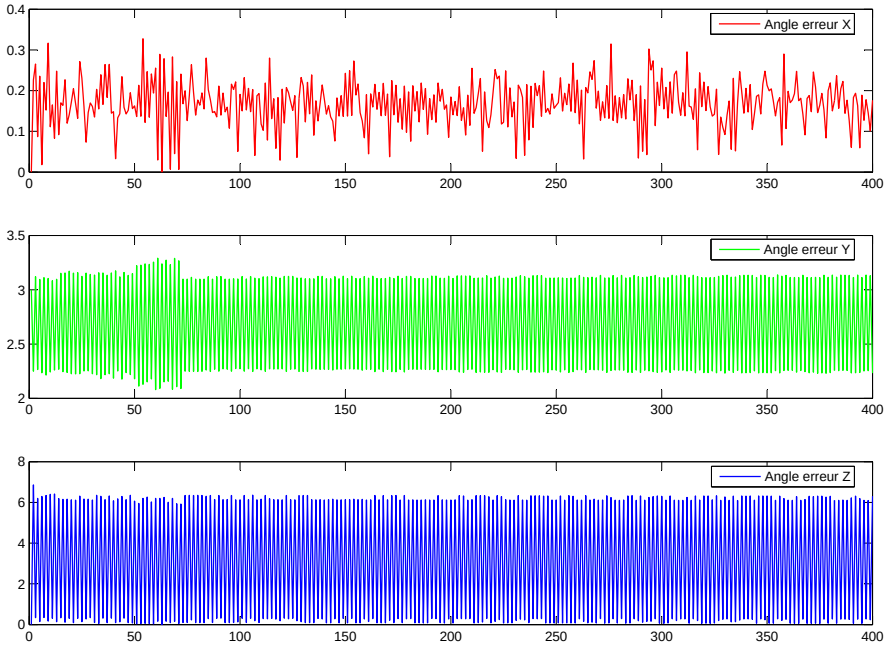


FIG. 2.23: Erreurs d'orientation avec des rotations autour de l'axe Y

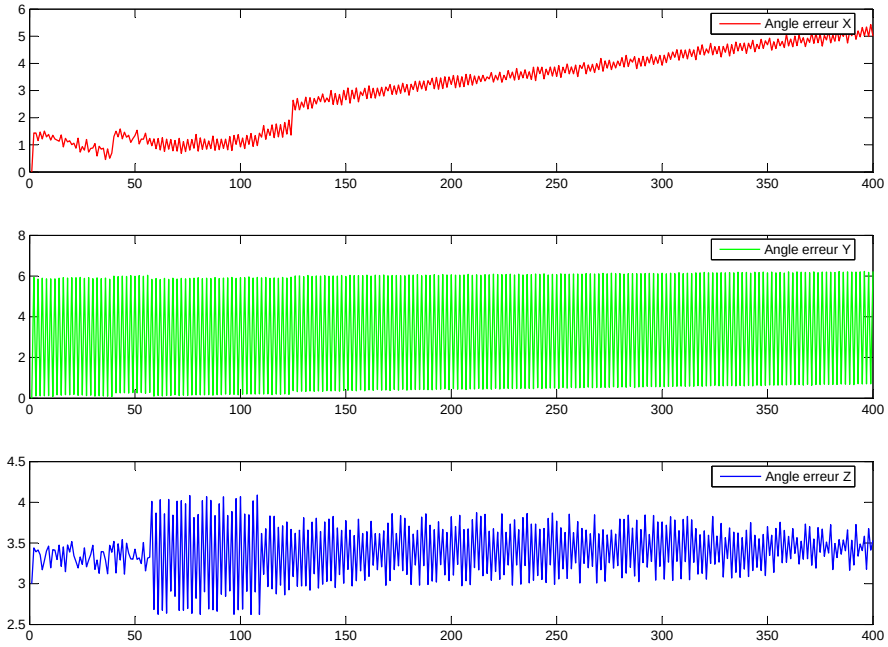


FIG. 2.24: Erreurs d'orientation avec des rotations autour de l'axe Z

Les erreurs obtenues sont données dans le tableau qui suit (cf. tab.2.7). Celles-ci sont représentées par l'erreur moyenne et l'écart type obtenues sur l'axe X (colonne 1), l'axe Y (colonne 2), l'axe Z (colonne 3) et l'angle représentant le mouvement entre les rotations (colonne 4).

		Axe X	Axe Y	Axe Z	θ
Rotation X	moyenne	6.0067	0.3677	3.9100	7.1886
	Ecart type	0.1853	0.0765	0.4271	0.3213
Rotation Y	moyenne	0.1680	2.6841	3.1890	4.8725
	Ecart type	0.0591	0.4433	3.0259	1.7400
Rotation Z	moyenne	3.0159	3.2274	3.3972	6.1634
	Ecart type	1.4285	2.8241	0.3468	1.7968

TAB. 2.7: Localisation basée vision : erreurs moyennes et écarts types des orientations

Par rapport aux résultats avancés par d'autres travaux, nos erreurs sont assez élevées. Cette différence est du à l'échelle de l'environnement. En effet, nous ne retrouvons pas de performances pour de méthodes évoluons dans de larges environnements. En ce qui nous concerne, nous pouvons conclure que ce que nous obtenons est satisfaisant. En effet, ceci nous permet d'obtenir des résultats de recalage corrects. De plus, par rapport à l'échelle de l'environnement, ces erreurs restent petites.

2.6.2.4 Temps d'exécution

Maintenant, nous allons nous intéresser aux temps de calcul pour l'estimation de la pose. Comme nous l'avons décrit auparavant, le calcul de pose passe par trois étapes essentielles à savoir :

1. le suivi visuel pour récupérer les appariements 2D/3D ;

2. élimination des appariements aberrants en utilisant l'algorithme RANSAC ;
3. calcul des paramètres de la pose à partir des couples obtenus avec l'algorithme RANSAC.

De ce fait, le temps de calcul de la pose par image est réparti sur ces trois phases. Le tableau 2.8 présente les temps d'exécution obtenus dans chaque phase. Il est exprimé en miliseconde. Il faut prendre en compte que ces temps dépendent du nombre de points utilisés. Ceux présentés dans ce tableau sont obtenus pour un ensemble de 42 points.

Etape	Temps en moyenne
Suivi avec KLT	10 ms
RANSAC	10 à 30 ms
Calcul de pose	$\leq$ 10 ms
Total	de $\leq$ 30 ms à $\leq$ 50 ms

TAB. 2.8: Temps de calcul par phase pour le calcul de pose

Entre les trois phases, l'algorithme RANSAC reste le plus consommateur en temps de calcul étant donné son aspect itératif. Malgré le fait que le calcul de pose soit effectué par minimisation itératif, l'itération orthogonale arrive à converge rapidement avec un temps généralement inférieur à 10 ms.

Le calcul de pose s'exécute pour une cadence comprise entre 30 images par secondes et 20 images par secondes. Cependant en pratique, nous l'avons évolué pour une cadence de 10 images par secondes. Ceci est du à une certaine contrainte matérielle (La vitesse du port USB). Mais ceci n'altère en rien la qualité du suivi, ni du système de réalité augmentée.

2.6.2.5 Résultats de recalage

Après avoir quantifié les performances de l'approche de vision que nous avons mise au point, nous allons nous intéresser aux résultats visuels obtenus c'est à dire le recalage réel/virtuel obtenu à partir des poses estimées.

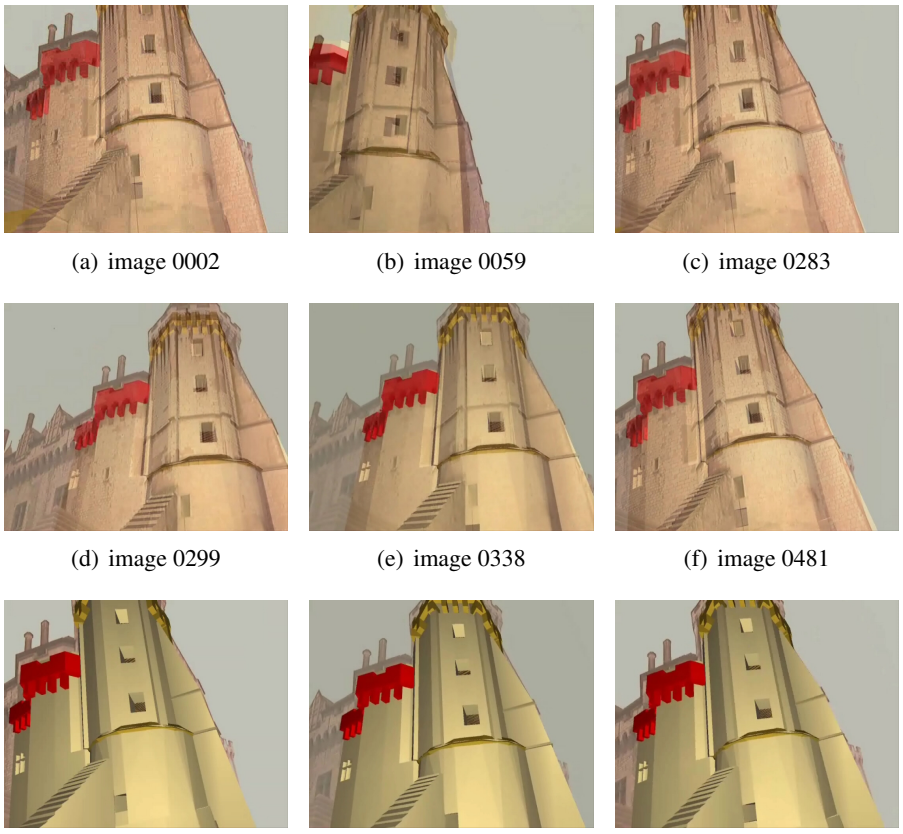
Dans un premier temps, nous réalisons le recalage d'un modèle filaire représentant une façade d'un bâtiment. Différents point de vue sont pris du bâtiment. La figure 2.25 présente le résultat obtenu où nous visualisons le modèle filaire (en ligne rouge) projeté sur la façade réelle du bâtiment.

Comme second résultat, nous présentons le recalage d'un modèle surfacique représentant un château, sur une séquence d'images acquises autour de la tour de ce château. A partir des poses calculées avec la méthode décrite dans ce chapitre, le modèle surfacique de ce château est projeté sur la vue réelle. La caméra évolue autour d'une des tours du château. Le modèle virtuel représentant cette tour se recale bien avec la tour réelle comme nous pouvons l'observer sur les images présentées dans la figure 2.26.

2.7 Conclusion

Dans ce chapitre, nous nous sommes intéressés aux approches de localisation utilisant la vision. Ces approches restent privilégier dans de nombreuses applications et essentiellement en vision par ordinateur. Nous avons présenté une taxonomie qui se base sur le degré de connaissance de l'environnement (méthodes avec connaissance *a priori* et sans

FIG. 2.25: Résultats de recalage d'un modèle filaire (rouge) sur la façade d'un bâtiment.



(a) image 0002 (b) image 0059 (c) image 0283

(d) image 0299 (e) image 0338 (f) image 0481

(g) image 0527 (h) image 0634 (i) image 0779

FIG. 2.26: Résultat de recalage d'un modèle 3D sur le château de Saumur

connaissance). Dans chaque classe, nous nous sommes intéressés aux principes des méthodes proposées qui diffèrent selon la nature des données utilisées. Par la suite, nous avons effectué une étude comparative entre les principales méthodes. Chaque approche présente des avantages et des inconvénients. Son efficacité dépend généralement du cadre d'utilisation. Toutefois, cette étude nous a permis de cerner notre cas d'étude et nous a guidés dans nos choix.

Nous avons opté pour notre système de localisation pour une approche basée modèle étant donné que nous disposons d'un modèle partiel et précis de l'environnement. Les environnements où nous évoluons contiennent plusieurs types de données tels que des plans, des contours et des points. Notre choix s'est porté sur les points car les environnements sont texturés. Nous aurions pu utiliser les contours ou les segments, mais fasse à ce type de scène le nombre de faux appariements augmente ce qui influe sur la précision de l'estimation. L'autre choix qui s'offrait à nous était d'utiliser les plans. Mais dans ce cas là, nous nous serions détachés de l'utilisation du modèle 3D. A partir de là, notre choix s'est porté sur l'utilisation des points 3D.

L'approche que nous proposons s'inspire du principe d'approches proposé au sein de notre équipe mais qui s'inscrivent dans la classe des méthodes basées marqueurs. Le principe général est d'identifier l'ensemble des appariements 2D/3D pour estimer les paramètres de pose. La clé de l'approche réside dans cette phase d'appariements. Pour cela, nous avons subdivisé cette étape en deux. La première étape d'initialisation qui permet d'obtenir un premier ensemble d'appariement. Cette initialisation, qui requière l'intervention de l'utilisateur, consiste à chercher les appariements des points 3D extraits en utilisant des descripteurs. La seconde étape a pour rôle de maintenir l'appariement obtenu. Pour cela, nous optons pour une approche de suivi visuel qui permet d'identifier dans chaque image les projections 2D des points 3D retrouvés à l'instant précédent.

Différentes expérimentations ont été conduites pour quantifier les performances de l'approche que nous avons mise au point. Notre approche donne de bon résultat notamment pour le recalage réel/virtuel qui est le critère principal pour les systèmes de réalité augmentée. Ces résultats restent similaires à ceux avancés par d'autres chercheurs. Cependant, l'approche présente quelques limites et échoue dans certains cas. Ceci est du essentiellement à la qualité de l'appariement 2D/3D. En effet, les occultations ou bien les mouvements brusques peuvent survenir à tout instant et engendrer ainsi des mauvais appariements. La question qu'on s'est alors posé est comment détecter ces défaillances ? Et surtout, comment peut-on régénérer cet ensemble d'appariement 2D/3D après l'échec du suivi ?

Dans ce qui va suivre, nous allons apporter des réponses à nos interrogations. Pour cela, nous nous orientons vers une approche combinant plusieurs types de capteurs afin de compenser les défaillances des méthodes basées vision. Dans le chapitre qui suit, nous allons présenter les systèmes multi-capteurs proposées dans la réalité augmentée ainsi que la solution que nous proposons.

Chapitre 3

Systèmes de localisation multi-capteurs

Le processus de localisation est crucial pour les applications de réalité augmentée. Comme nous l'avons vu précédemment, les méthodes basées vision sont privilégiées, essentiellement en vision indirecte, pour leur précision. Cependant, en milieu extérieur, ces approches restent sensibles aux conditions de travail (changements de luminosité, occultations et mouvements brusques). Certes, ces conditions existent également en intérieur, mais leur influence reste moindre car elles sont généralement contrôlables. C'est pour cette raison que les applications de réalité augmentée en extérieur convergent vers l'utilisation de systèmes de localisation dit multi-capteurs. En effet, la combinaison de différents types de capteurs a pour objectif de pallier les inconvénients de l'utilisation d'un seul type de capteur. Ceci permet de gagner en robustesse et en précision.

Dans le présent chapitre, nous nous intéressons aux systèmes qui combinent plusieurs types de capteurs. Tout d'abord, nous dressons un bref état de l'art sur les systèmes multi-capteurs dédiés essentiellement aux applications de RA mobile. Nous présentons une taxonomie utilisant comme critère la stratégie de combinaison des différents capteurs. Par la suite, nous établissons une étude comparative sur la base de cette étude bibliographique. Ceci nous permet de justifier nos choix et ainsi mettre en avant les problématiques traitées par un tel système de localisation. La seconde partie du chapitre esquisse les grandes lignes du système de localisation que nous proposons. De plus, nous dresserons une liste des problématiques traitées tout au long de sa mise en œuvre.

3.1 Taxonomie des systèmes multi-capteurs

L'idée de combiner plusieurs types de capteurs n'est pas nouvelle. Les premiers systèmes multi-capteurs pour la localisation firent leur apparition avec les applications robotiques. En effet, [Viéville et al., 1993] proposaient de faire coopérer la vision avec un capteur inertiel pour corriger automatiquement la trajectoire d'un robot mobile autonome. Cette idée s'inspire du comportement humain. L'être humain s'oriente dans son environnement en s'aidant de l'organe vestibulaire, situé au niveau de l'oreille interne, et de ses yeux. Par comparaison, la centrale inertielle a la fonction de l'organe vestibulaire et la caméra remplace l'œil. Parallèlement, dans [Azuma, 1993], suivant les contraintes de reca-

lage imposées par les applications RA, *R. Azuma* propose d'utiliser les capteurs hybrides afin d'améliorer la précision du recalage. Il donne l'exemple des capteurs inertiels, qui ont une portée infinie, mais qui perdent graduellement de la précision (phénomène de dérive). Grâce à des mesures fournies par plusieurs types de capteurs pendant une courte période de temps, la dérive peut-être corrigée et l'efficacité globale du système amélioré.

Les premiers systèmes de réalité augmentée en extérieur tel que la plate-forme MARS [Hollerer et al., 1999] et la plate-forme BARS [Julier et al., 2000] utilisaient un GPS ou un DGPS pour estimer la position absolue de l'utilisateur ainsi qu'un capteur inertiel couplé avec un compas électronique pour estimer l'orientation. Étant donné que ce type de capteur est moins précis, les travaux récents se sont plutôt orientés vers le couplage des méthodes basées vision avec d'autres types de capteurs essentiellement de type inertiel.

L'hybridation permet de compenser les faiblesses des différents capteurs lorsque ces derniers sont utilisés séparément. Les capteurs les plus récurrents sont le GPS et la centrale inertielle. Pour rappel, le GPS est un système de positionnement par satellites d'une précision de l'ordre de 10 mètres pour les GPS grand public et du mètre pour les GPS professionnels. Toutefois, le GPS est généralement moins précis en environnement urbain à cause des réflexions multiples des ondes sur les façades des immeubles ou de l'occultation des satellites, ce qui le rend peu exploitable dans ces milieux. Par ailleurs, les centrales inertielles sont composées d'accéléromètres et de gyroscopes, leur problème principal est celui de la dérive (cf. section 1.2.2.2). Plusieurs travaux se sont intéressés à cette problématique de fusion telle que les travaux de [Azuma et al., 1999], [Foxlin et Naimark, 2003], [Alves et al., 2004], [Caarls et al., 2008] et [Chai et al., 2002]. Dans la littérature, nous pouvons distinguer deux stratégies de combinaison des capteurs de nature hétérogène : la fusion de données ou la suppléance.

3.1.1 La fusion de données

Plusieurs travaux existants dans la littérature convergent vers le principe de fusion de données, effectuée essentiellement entre les données extraits de la vision et celles fournies par la centrale inertielle, en utilisant un filtre de Kalman comme par exemple dans les travaux de [You et al., 1999], [Ribo et al., 2002], [Foxlin et Naimark, 2003], [Hol et al., 2006], [Reitmayr et Drummond, 2006] et [Bleser et Stricker, 2008a] [Ababsa, 2009] ou un filtre particulaire [Ababsa et Mallem, 2007] et [Bleser et Stricker, 2008b]. Cette stratégie consiste à fusionner toutes les données fournies par tous les capteurs utilisés pour se localiser. Ces systèmes suivent un modèle de prédiction/correction. Les données fournies par d'autres capteurs, comme les gyroscopes et magnétomètres, sont utilisés pour prédire le mouvement 3D de la caméra qui est ensuite ajusté et affiné en utilisant des techniques de vision artificielle.

Souvent utilisé, le filtre de Kalman [Kalman, 1960] est un filtre récursif qui estime l'état d'un système dynamique non-linéaire à partir d'une série de mesures bruitées. L'estimateur récursif signifie que seulement l'estimation de l'état à l'instant précédent et les mesures à l'instant courant sont nécessaires pour estimer l'état actuel. Donc, il n'est pas nécessaire d'avoir un historique des observations et/ou des estimations. Dans ce qui suit, nous allons étudier quelques approches intéressantes proposées dans la littérature et essentiellement celle de la communauté RA.

3.1.1.1 You et al.

Dans [You et al., 1999], les auteurs ont démontré la faisabilité d'un capteur hybride combinant un système de vision avec un compas et trois gyroscopes pour estimer l'orientation du point de vue dans un environnement extérieur (cf. fig.3.1). Cette fusion a pour objectif d'exploiter la nature complémentaire des capteurs pour compenser le manque de robustesse de la vision ainsi que l'effet de dérive du à l'intégration des vitesses angulaires pour mesurer les orientations. En effet, les données inertielles fournies par les gyroscopes augmentent la robustesse et réduisent le temps de calcul alors que le système de vision fournit une estimation de l'orientation de la caméra qui permet par la suite de corriger les dérives du système gyroscopiques.

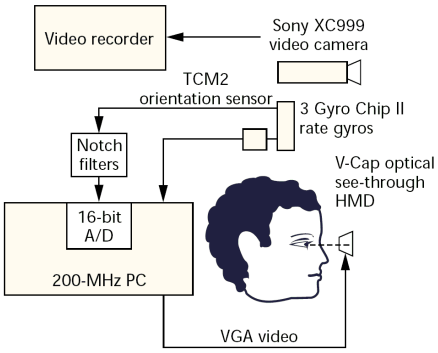


FIG. 3.1: Descriptif du système multi-capteurs présenté dans [You et al., 1999]

Afin de déterminer les orientations du point de vue de l'utilisateur, la fusion est faite selon un modèle de prédiction/correction. La prédiction du mouvement angulaire est estimée en filtrant et fusionnant les données d'un compas avec celles des gyroscopes par extrapolation des données fournies par ces capteurs (cf. fig.3.2). L'utilisation du compas a pour but de stabiliser les orientations calculées à partir des gyroscopes.

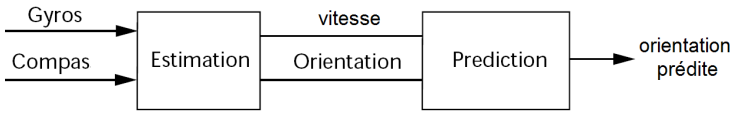


FIG. 3.2: Le modèle de prédiction de l'orientation à partir du compas et des gyroscopes [You et al., 1999]

Ces orientations sont utilisées pour approximer le mouvement 2D des points images. Concrètement, ceci permet de prédire la position des points suivis dans l'image courante en se basant sur la relation entre les orientations et le mouvement image. Par la suite, le suivi visuel corrige et raffine l'approximation du mouvement 2D par une recherche locale des points dans l'image. Ceci est obtenu en utilisant un calcul de flot optique basé sur le mouvement des normales. Le calcul du flot optique est obtenu par minimisation de moindre carrées afin de trouver la meilleure estimation de mouvement pour chaque région centrée sur un point 2D. La dernière phase consiste à convertir le mouvement résiduel 2D en une orientation 3D qui permet de corriger la dérive des gyroscopes. Ceci consiste à retrouver la dérive qui minimise le mouvement résiduel. Le mouvement résiduel n'est autre que la

différence entre la vitesse obtenue à partir des données gyroscopiques et la vitesse calculée à partir des points images.

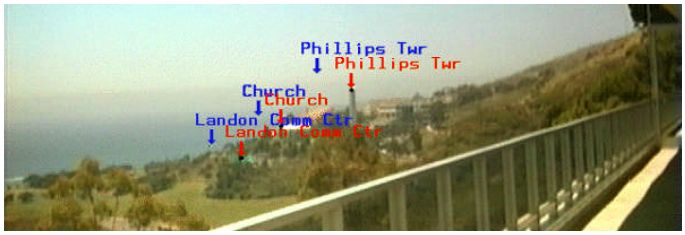
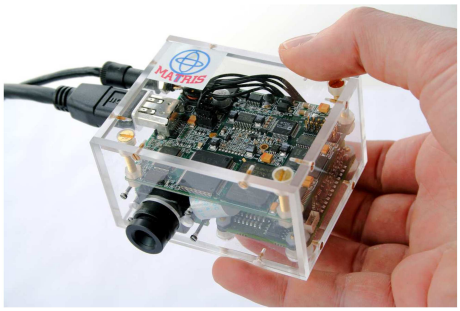


FIG. 3.3: Exemple d'annotations dans un environnement extérieur [You et al., 1999] : avec les données inertielles sans correction (en bleu) et avec correction en utilisant la vision (en rouge)

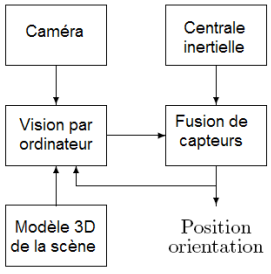
Ce système a été testé aussi bien en intérieur qu'en extérieur. Dans leurs expérimentations, les auteurs sélectionnent l'ensemble de points à suivre. Ces points sont re-projetés en utilisant les orientations fournies par le système. Comme critère de précision, les auteurs utilisent l'erreur moyenne entre la position des points projetés et la position réelle dans l'image. En comparant entre les orientations fournies uniquement par le capteur inertielle (i.e. gyroscope et compass) et celles utilisant la correction basée vision, *You et al.* constatent une amélioration de l'ordre de 4.27 pixels ce qui est équivalent à 0.4°. En supposant que les points à suivre sont très distant, la translation peut être négligée et le mouvement de la caméra est approximé par une rotation pure. Ceci n'est plus vrai lorsque les points deviennent très proches d'où le besoin de données additionnelles pour estimer le mouvement de translation. Ce modèle a été expérimenté hors ligne en raison du temps de traitement élevé (en 1999). A titre indicatif, le système fonctionnait à une cadence de 2 à 4 images par seconde sous SGI O2 et à 10 images par seconde.

3.1.1.2 Hol et al.

Dans le cadre du projet européen "**Matris**" [Hol et al., 2006], les auteurs proposent de combiner les données d'une centrale inertielle avec la vision en utilisant un filtre de kalman étendu (non-linéaire).



(a) Capteur hybride



(b) Principe de la fusion

FIG. 3.4: Le modèle de fusion selon [Hol et al., 2006]

Pour fusionner les données de la vision et de la centrale inertielle, les auteurs proposent un modèle de processus et un modèle d'observation. Le modèle de processus est

déduit en intégrant les équations décrivant la cinématique de la caméra. Celle-ci est décrite par un ensemble d'équations différentielles continues. Le modèle 3D utilisé est composé d'un ensemble de points 3D auxquels sont associés des imagettes. La figure 3.5 illustre un exemple d'imagettes composant un modèle 3D d'une scène. Lors de la phase de suivi, les imagettes sont transformées par une homographie estimée à partir de la dernière pose prédite de la caméra. Les imagettes obtenues sont mises en correspondance avec l'image courante en utilisant l'algorithme de Kanade-Lucas-Tomasi (KLT) pour l'alignement [Lucas et Kanade, 1981]. La liste d'appariements 2D/3D est utilisée dans le modèle d'observation pour corriger la pose estimée.

FIG. 3.5: Exemple d'une scène avec les imagettes constituant le modèle 3D

Pour tester les performances de leurs systèmes, les auteurs ont utilisé des données inertielles avec des correspondances obtenues en projetant une scène artificielle sur un plan image dont la position et l'orientation sont considérées comme données de références. Ces correspondances virtuelles sont injectées avec un bruit réel dans le filtre avec des données fournies par la centrale inertielles. Trois types de mouvements ont été expérimentés : un mouvement lisse et lent (en utilisant un trépied à roulettes), un mouvement lisse mais relativement rapide (caméra portée par un utilisateur qui marche) et un mouvement rapide avec de grandes accélérations (caméra portée par un utilisateur qui court). Les résultats sont jugés satisfaisants avec une erreur quadratique moyenne de l'ordre de $0.005m$ en position et 0.1° en orientation pour le scénario rapide. En ce qui concerne les deux autres scénarios, ils obtiennent une erreur quadratique moyenne inférieur à $0.005m$ en position et 0.05° en orientation. Selon les auteurs, ces performances permettent de caractériser le filtre dans la mesure où on a de très bons appariements 2D/3D. Ceci n'est pas toujours le cas. Comme amélioration, les auteurs proposent d'intégrer une approche de type SLAM [Bailey et Durrant-Whyte., 2006] (Simultaneous Localisation And Mapping) pour l'apprentissage et l'enrichissement en ligne du modèle 3D.

3.1.1.3 Bleser et al.

G. Bleser présente dans ses travaux de thèse [Bleser, 2009] un système de localisation combinant une caméra avec une centrale inertielle. En effet, la pose obtenue en fusionnant des correspondances 2D/3D avec les données fournies par la centrale inertielle en utilisant un filtre de kalman étendu. La figure 3.6 illustre le flot de données dans le système proposé par *Bleser*. A un instant t, le capteur hybride fournit une image avec un ensemble de données inertielles acquises sur un intervalle. Dans un premier temps, le système effectue un test de divergence du filtre en se basant soit sur l'initialisation soit sur le processus de

la centrale inertielle. Trois critères différents sont utilisés *. Si l'un de ces critères est défaillant, alors le filtre est jugé divergeant. Dans le cas où le filtre diverge, le système procède à une initialisation qui consiste à définir les correspondances de points 2D/3D. *G. Bleser* propose 3 différents modes selon les données présentes :

- méthode manuelle en alignant le capteur hybride avec une pose fixée dans le cas ou il existe des perturbations ;
- méthode semi-automatique en utilisant l'orientation de la centrale inertielle si il n'y a pas de perturbations magnétiques et en utilisant une position définie manuellement ;
- méthode automatique en utilisant des images de référence dont les poses sont connues et un ensemble d'appariements 2D/3D est identifié. Lors de l'initialisation, il suffit d'apparier l'image courante avec l'image de référence qui présente le plus de similitude (selon l'histogramme de couleurs) pour ainsi déduire la pose initiale et les couples d'appariements.

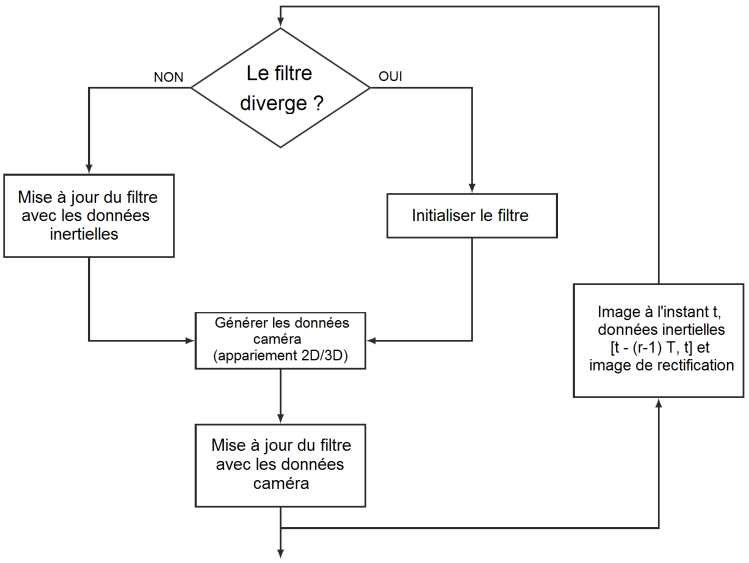


FIG. 3.6: Flot de données dans l'approche de localisation proposée dans [Bleser, 2009]

Dans le cas où le filtre ne diverge pas, ce dernier prédit une pose à partir de la pose précédente et des données fournies par la centrale inertielle et génère des appariements 2D/3D en utilisant l'approche haut-bas décrite dans la section 2.1.1.2 (page 35). Une fois les appariements 2D/3D obtenus, ceux-ci sont injectés dans le filtre de kalman avec les vitesses angulaires et les accélérations linéaires. Le modèle, dit modèle d'accélération, utilise toutes les informations fournies par l'accéléromètre, à savoir l'attitude de la caméra et les accélérations. Le modèle de mouvement suppose des accélérations et des vitesses angulaires constantes. L'utilisation des accélérations a pour effet de changer, durant la phase de mise à jour du filtre, les mesures en rapport avec la position, essentiellement l'accélération, ce qui se propage à la vitesse et la position dans les mises à jour ultérieures.

Afin d'évaluer les performances de leur système, les auteurs ont mené différentes expérimentations selon la taille de l'environnement : à petite échelle (sur un bureau) avec

*Le système utilise comme critère : l'écart entre la prédiction et les données réelles (l'innovation du filtre de Kalman étendu) calculée pour des appariements 2D/3D dont la distribution doit suivre une loi normale centrée ; la valeur de la norme de Frobenius de la matrice de covariance est dans un intervalle définie et la norme des quaternions est égale à 1.

des mouvements contrôlés (en utilisant un bras de robot), à moyenne échelle (salle) et à grande échelle (un foyer). Lors de ces deux derniers scénarios, le mouvement était libre vu que le dispositif était porté par l'utilisateur. Pour chaque expérience, le cas de mouvements rapides et de mouvements lents ont été considérés. Bien que les tests aient été réalisés en intérieur, ce travail méritait d'être cité car le modèle de fusion et de suivi sans marqueurs nous semblaient intéressant. Pour évaluer les performances du modèle de fusion proposé, *Bleser* l'a comparé à deux modèles de référence utilisés à savoir le modèle gyroscopique et le modèle de gravité. Le modèle gyroscopique combine les correspondances 2D/3D avec les vitesses angulaires. Cela suppose que les vitesses linéaires et angulaires sont constantes. Le modèle dit de gravité utilise l'accéléromètre comme un inclinomètre. Il se base sur un modèle de mouvement qui suppose que les vitesses linéaire et angulaires sont constantes. Le modèle gyroscopique est inclus dans le modèle d'accélération proposé.

En utilisant comme critère de performance la distance euclidienne entre la position prédite des points caractéristiques et leurs positions alignées, le modèle d'accélération présente une erreur moyenne de l'ordre de 0.93 pixels avec un écart type égal à 0.62 pixels au lieu de 3.83 pixels ($\sigma = 4.03$ pixels) pour le modèle gyroscopique et 3.82 pixels ($\sigma = 4.02$ pixels) pour le modèle de gravité. Ces résultats sont essentiellement obtenus pour le premier scénario à petite échelle avec des mouvements rapides. Concernant le scénario à grande échelle (le plus intéressant dans notre étude), les auteurs présentent une erreur moyenne qui ne dépasse pas 1.42 pixels avec un écart type de l'ordre de 1.46 pixels. Concernant les deux autres modèles témoins (i.e. gyroscopique et de gravité), les auteurs n'ont pas obtenu de résultats sur les séquences tests à grandes échelles. Par rapport au suivi basé uniquement sur la vision, les auteurs observent qu'ils ont moins d'oscillations (*jitter*). De plus, les temps de calcul sont moindres lorsque la prédiction est précise, l'alignement des imagettes synthétique est obtenu plus rapidement (convergence rapide). Les auteurs jugent que les résultats obtenus dans un environnement à petite échelle et à grande échelle, sous différentes conditions de luminosité et de mouvements de caméra, sont robustes. Cependant, lors d'occultations de données de vision, le système reste fonctionnel durant un court moment seulement.

3.1.1.4 Ababsa et al.

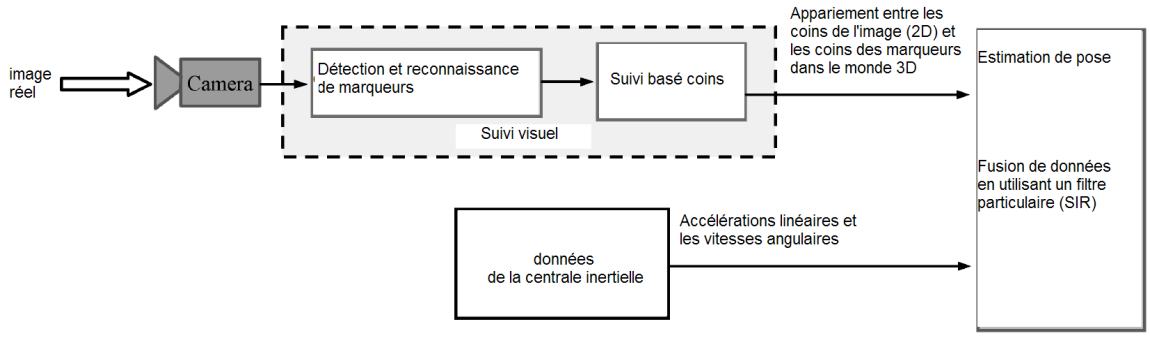


FIG. 3.7: Le système Hybride proposé dans [Ababsa et Mallem, 2007] : flot de données

Contrairement aux approches de fusion classiques qui se basent sur un filtre de Kalman, [Ababsa et Mallem, 2007] proposent d'utiliser un filtre particulaire. Ce dernier, connu aussi sous le nom de Méthode séquentielle de Monte-Carlo, est une technique d'estimation de

modèles basée sur la simulation. Les filtres particulaires sont généralement utilisés pour estimer des modèles Bayésiens. S'ils sont conçus correctement, les filtres particulaires peuvent être plus rapides que les Méthodes de Monte-Carlo par Chaînes de Markov. Ils constituent souvent une alternative aux filtres de Kalman étendus avec l'avantage qu'avec suffisamment d'échantillons, ils approchent l'estimée Bayésienne optimale. Le filtre particulaire tire son avantage du fait que contrairement au filtre de kalman, il permet d'employer d'autres hypothèses que le bruit blanc gaussien sur les mesures. Ils peuvent donc être rendus plus précis que les filtres de Kalman.

L'approche proposée par *Ababsa et al.* combine un suivi basé marqueurs avec des données gyroscopiques et des accélérations fournies par une centrale inertielle (cf. fig.3.7). L'algorithme de vision est subdivisé en deux phases : une phase de reconnaissance des marqueurs visibles suivi d'une phase d'appariement des données 2D/3D. L'algorithme de fusion est un filtre particulaire avec ré-échantillonnage par importance SIR (Sampling Importance Resampling). Le filtre estime la pose de la caméra à partir des appariements 2D/3D et des données inertielles suivant un modèle décrivant la cinématique de la caméra. Le modèle de mouvement utilisé suppose que l'accélération de la caméra est constante. Le mouvement de la caméra est décrit par les orientations (angle d'Euler), la vitesse angulaire, la position, la vitesse et l'accélération de la caméra par rapport au référentiel associé au monde.

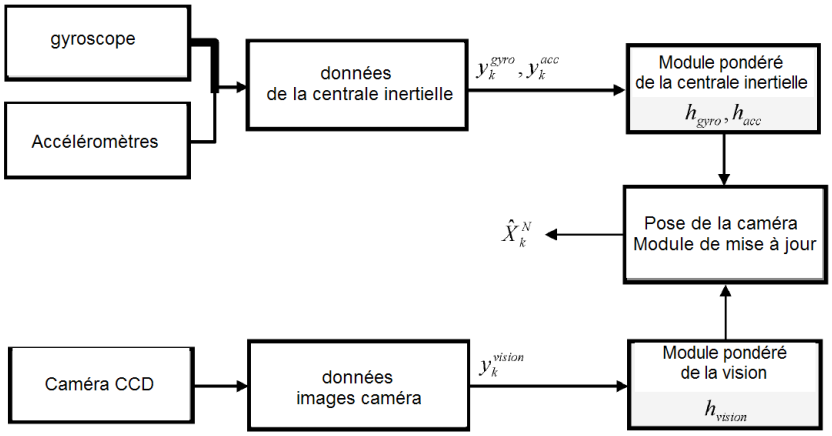


FIG. 3.8: Un filtre complémentaire pour la fusion de capteurs [Ababsa et Mallem, 2007]

Les deux capteurs n'ayant pas la même fréquence d'échantillonnage, les auteurs utilisent un filtre complémentaire (cf. fig.3.8). Ce dernier a la particularité d'avoir deux canaux pondérés : un canal pour les mesures de vision et un autre canal pour les données de la centrale inertielle. Ceci permet de gérer l'indisponibilité des données à tout moment. En effet, si par exemple les données de vision ne sont pas disponibles (à cause des occultations des cibles par exemple), l'estimation de pose est obtenue uniquement à partir des données de la centrale inertielle et vice versa.

Les résultats de simulation présentés sont satisfaisants. En effet, avec 100 particules, les auteurs obtiennent un résultat correct avec une erreur en position de l'ordre de $0.16mm$ et de $0.00019rad \simeq 0.01°$ en orientation. Par comparaison avec le filtre de Kalman étendu, le filtre particulaire donne de meilleurs résultats. Cependant, il présente un temps d'exécution

plus important. *Ababsa et al.* font état d'un coût calculatoire de l'ordre de 33*ms* (29*ms* pour la phase d'identification des marqueurs et de 4*ms* pour l'estimation de pose).

3.1.1.5 Reitmayr et Drummond

Le système proposé par [Reitmayr et Drummond, 2006] combine également la vision avec des données inertielles (cf. fig.3.9). Le système utilise un suivi basé contours afin d'obtenir une localisation précise. Des mesures gyroscopiques sont utilisées pour gérer les mouvements brusques. En ce qui concerne les mesures de gravité et le champ magnétique, celles-ci sont utilisées pour éviter l'effet de dérive.

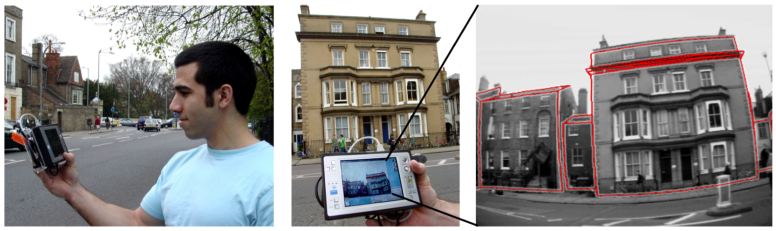


FIG. 3.9: Suivi de contours basé Inertiel/Vision [Reitmayr et Drummond, 2006]

Le suivi de contours se base sur la méthode décrite dans [Klein et Drummond, 2003] (cf. section 2.1.1.2 page 35). La fusion des données est obtenue avec un filtre de Kalman étendu (EKF) basé sur un modèle de vitesse constante. Le vecteur d'état contient les paramètres de pose et de vitesse. Dans ce cas, les données inertielles sont utilisées pour estimer le mouvement rotationnel.

Pour détecter la défaillance du suivi, les auteurs proposent d'utiliser un test de qualité qui se base sur un calcul de rapport logarithmique entre la probabilité d'un suivi correct et la probabilité d'un suivi défaillant. Dans la première déclinaison, lorsque le suivi échoue, le système essaye de réinitialiser par appariement entre l'image courante et des images de références dont les poses sont connues *a priori* et obtenues en ligne. Si l'appariement réussit, la pose est mise à jour à partir du mouvement obtenu entre l'image courante et l'image de référence. Dans le cas contraire, le filtre de vitesse est réinitialisé et la pose n'est pas fusionnée afin d'éviter d'introduire des erreurs d'estimation. Cependant, les auteurs constatent que vers la fin des séquences, il est difficile d'avoir une bonne réinitialisation et donc que le suivi diverge. De plus, le système ne comprend pas de phase d'initialisation. Cette dernière est faite de manière manuelle. L'utilisateur aligne manuellement la vue à partir d'un point de vue prédéfini. En ce qui concerne la robustesse, les occultations totales sont tolérées à condition que la vue dégagée après l'occultation ressemble à une des vues de références.

Le système présenté a été testé sur deux sites différents dont les modèles 3D sont constitués de surfaces planaires texturées. En utilisant les cartes de vecteurs[†] (*vector map*) comme données de référence, l'erreur sur la position obtenue présente un écart type de $(\sigma_x, \sigma_y, \sigma_z) = (0.0979m, 0.1577m, 0.1463m)$[‡]. De plus, selon les expérimentations conduites, le système n'est pas robuste face aux mouvements brusques et/ou larges.

[†] carte vectorielle est une collection de données SIG basée vecteur à différents niveaux de détails. Wikipédia

[‡] x est l'est, y l'élévation et z est le nord

Dans la procédure de récupération décrite dans [Reitmayr et Drummond, 2006], il est indispensable de sauvegarder en ligne les images avec les poses associées. De plus, il faut apparier l'image courante avec ces images de références ce qui peut être fastidieux. Le mécanisme de récupération ne peut remplacer une procédure d'initialisation car il est quasi impossible d'avoir des images de références bien distribuées dans l'environnement. De plus, la précision et la robustesse de la localisation dépend fortement de la phase d'initialisation. De ce fait, les auteurs proposent dans [Reitmayr et Drummond, 2007] d'utiliser la position 2D fournie par un GPS (longitude et latitude) pour initialiser le suivi. La méthode propose d'utiliser une zone de recherche sous forme d'ellipse définie autour de la position GPS. La phase d'initialisation se décline en plusieurs étapes qui sont :

1. Définir une zone de recherche sous forme d'ellipse dont le centre est la position 2D du GPS.
2. Réduire la zone de recherche en éliminant les parties d'intersection avec les bâtiments (cf. fig.3.10).
3. Découper cette zone en grille.
4. A chaque position de cette grille, réaliser un rendu image du modèle 3D afin d'extraire les contours visibles du modèle 3D et de les apparier avec les contours de l'image courante.
5. Une fois la pose estimée à partir de cet appariement, calculer le score du test de qualité.
6. Retenir la position avec le plus grand score.

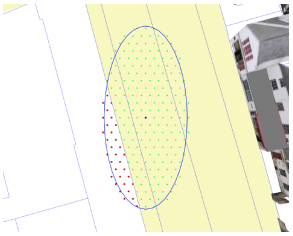


FIG. 3.10: Définition de la zone de recherche dans [Reitmayr et Drummond, 2007] : les points de la grille en vert sont les positions validées pour la recherche, les points rouges sont les points de l'intersection.

La procédure d'initialisation est aussi utilisée lorsque le suivi devient défaillant. Dans ce cas, les auteurs proposent de corriger la position fournie par le GPS avec une erreur prédite en utilisant un processus Gaussien [Williams, 1997]. Ce processus se base sur un apprentissage en ligne de l'erreur qui est définie comme l'écart entre la position estimée par la vision et la position GPS. La correction à partir de l'erreur prédite permet de se rapprocher de la position réelle et ainsi de réduire le temps de réinitialisation. En effet, le temps requis pour la réinitialisation dépend de la distance entre la position GPS et la position réelle de la caméra.

En terme de performances, en considérant que la position fournie par la vision est la donnée de référence, les auteurs obtiennent une erreur moyenne de $0.148m$ avec un taux de 0.11% de faux appariements en utilisant la recherche aux voisinages. Ces résultats sont obtenus lors de la phase d'initialisation à partir de différentes positions fixes dans l'environnement. Pour évaluer les performances de la prédiction d'erreur, les positions GPS corrigée

par l'erreur prédite sont injectées dans un filtre de kalman à vitesse constante afin de prédire une position de la caméra. Ces positions sont comparées aux positions obtenues avec l'approche de vision. Les auteurs constatent que l'erreur prédite est souvent inférieure à l'erreur réelle. Avec le temps, cette erreur converge car le processus gaussien converge vers une moyenne nulle. En étudiant les résultats présentés, nous remarquons que cette différence varie entre $1m$ et $10m$. Dans leurs différentes expérimentations, les auteurs constatent qu'ils ont un écart type de l'ordre de $\sigma = (1.9m, 4.3m)$ sur les directions est-ouest et nord-sud.

L'approche proposée est intéressante. Cependant, dans le processus de fusion, la translation est négligée. Ceci ne pose pas de problème quand le mouvement entre deux images successives est très petit car le mouvement peut être considéré comme une rotation pure. Il est clair que cette situation n'est pas toujours vérifiée, surtout dans le cas des mouvements rapides. D'autre part, la procédure d'initialisation reste lourde en temps de calcul surtout en cas d'échec.

3.1.1.6 Schall et al.

FIG. 3.11: Système de RA proposé dans [Schall et al., 2009]

Parmi les récents travaux, nous trouvons le système proposé par [Schall et al., 2009] visant à estimer une pose globale dédiée à des applications en environnements extérieurs. D'une part, les auteurs proposent de mettre au point un système d'un point de vue matériel et ergonomique plus convivial pour les utilisateurs finaux. D'autre part, le système doit fournir une estimation de pose globale précise et robuste pour une grande qualité de recalage réel/virtuel. Le système est illustré dans la figure suivante (cf. fig.3.11).

Le système combine un DGPS ou un GPS employant la technique de positionnement dite cinématique temps réel (Real Time Kinematic ou RTK) [§] avec un capteur barométrique pour estimer la position globale. Le capteur barométrique est utilisé pour l'altitude. Pour l'orientation, le système proposé utilise une centrale inertielle afin d'utiliser les vitesses angulaires fournies par les gyroscopes, les accélérations linéaires estimées par les accéléromètres et les champs magnétiques fournis par les magnétomètres. Afin de compenser les perturbations de la centrale inertielle essentiellement des mesures magnétiques déformées par les champs électromagnétique, le système comprend une approche de suivi panoramique qui n'estime que l'orientation.

[§]La Cinématique temps réel (Real Time Kinematic) est une technique de positionnement par satellite basée sur l'utilisation de mesures de phase des ondes porteuses des signaux émis par le système GPS, GLONASS ou Galileo. Une station de référence fournit des corrections en temps réel permettant d'atteindre une précision de l'ordre du centimètre. Dans le cas particulier du GPS, le système est alors appelé Carrier-Phase Enhancement ou CPGPS. Wikipédia

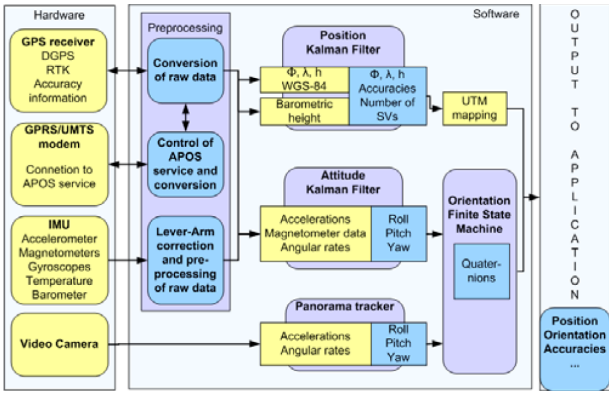


FIG. 3.12: Architecture du système proposé dans [Schall et al., 2009]

Le fonctionnement du système est décrit par l'architecture proposée dans la figure 3.12. Pour estimer la position, le système se base sur un filtre de Kalman appelé filtre de Kalman **Position** qui fusionne les données fournie par un DGPS (ou un GPS et un RTK) avec les hauteurs obtenues avec un capteur barométrique. Selon les auteurs, ce dernier est utilisé parce que les hauteurs fournies par le capteur barométrique sont plus stables que les hauteurs fournies par le GPS/DGPS. A l'état initial, le capteur barométrique est initialisé avec la hauteur du GPS utilisée comme référence. Le filtre de Kalman se base sur un modèle de vitesse constante ainsi qu'un modèle de mouvement uniforme pour le modèle dynamique. De plus, le filtre réalise la fusion entre les deux capteurs en tenant compte de leurs précisions (c'est à dire respectives une pondération des données utilisées). Pour ce qui est des orientations, le filtre de Kalman **Attitude** combine les données fournies par les gyroscopes, les accéléromètres et les magnétomètres. Ce filtre se base sur 3 modèles différents : un modèle pour des mesures fournies par les gyroscopes afin d'estimer les vitesses de rotation du roulis, du tangage et du lacet, un modèle pour les mesures fournies par les magnétomètres pour estimer le lacet magnétique qui diffère du vrai lacet étant donné que le magnétomètre correspond à une boussole indiquant le nord magnétique, et un modèle pour les mesures retournées par les accéléromètres qui permettent d'estimer le roulis et le tangage.

Afin de compenser les dérives de la centrale inertielle, de détecter et de corriger les déviations des magnétomètres, les auteurs proposent une méthode basée vision qui n'utilise pas de modèle 3D. Les points caractéristiques sont déterminés dynamiquement et au fur et à mesure. L'approche utilisée est similaire aux méthodes "*environment mapping*", *reflection environment* ou *skybox* connues en infographie. L'idée est de cartographier l'environnement dans un cylindre. L'approche suppose que le mouvement est un mouvement rotationnel pur (la translation étant négligée). Pour chaque image, la nouvelle pose est estimée afin d'ajouter de nouvelles entrées dans une carte dense construite de l'environnement. L'orientation est mise à jour en utilisant les appariements obtenus entre les points extraits de la carte et de l'image courante. L'appariement est réalisé de manière itérative à l'aide de l'algorithme de Gauss-Newton. Puis, l'image est projetée pour ajouter de nouvelles entrées. Selon les auteurs, leur approche diffère des approches SLAM, où le système estime simultanément la pose et la cartographie de l'environnement (reconstruction 3D de l'environnement), par le fait que les entrées sont ajoutées dans la carte et ne sont pas mise à jour par la suite contrairement aux approches SLAM où la reconstruction est raffinée au fur et à mesure. Si l'écart entre l'orientation retournée par le filtre de Kalman **Attitude** et l'orientation estimée par la vision varie, l'orientation obtenue par la vision est jugée plus fiable et est utilisée comme

orientation finale. Une fois que l'écart a été réduit et devient proche de zéro, c'est l'orientation calculée par le filtre de Kalman **Attitude** qui est considérée. Dans le cas où le suivi visuel échoue, les données de la centrale inertielle sont considérées valides jusqu'à ce que le suivi visuel se réinitialise.

La précision obtenue par les différents filtres n'est pas présentée explicitement. On notera une précision en moyenne de l'ordre de $(x = 0.8m, y = 0.765m)$ avec un écart type égal à $(\sigma_x = 1.848, \sigma_y = 1.535)$ pour le DGPS. D'après les graphiques présentés, l'écart entre les orientations filtrées et les orientations fournies par la centrale inertielle est de l'ordre de 5 degrés. Pour le recalage réel/virtuel, les auteurs présentent une capture d'écran où on observe le recalage d'un réseau de tuyaux ainsi que de câbles électriques sur une portion d'une rue. Le système proposé fait suite aux travaux menés dans le cadre du projet Vidente présenté dans le chapitre 1.

3.1.2 La suppléance des données

Les approches décrites précédemment tirent leur intérêt du fait qu'à tout moment les mesures sont estimées en fusionnant toutes les données fournies par les différents capteurs utilisés suivant un modèle qui décrit la cinématique de mouvement de la caméra. Certains travaux proposent d'utiliser des filtres de type complémentaire afin de pallier les différences de temps d'échantillonnage ainsi que les indisponibilités de données à certain moments.

Parallèlement aux approches de fusion, d'autres travaux présentés dans la communauté de RA proposent une approche dite de suppléance. Le principe consiste à utiliser un capteur principal pour fournir une information de localisation assez précise et robuste. Lorsque ce capteur ne peut pas fournir des données cohérente, il est suppléé par un ensemble d'autres capteurs. Ce concept apparait avec les travaux de [Borenstein et Feng, 1996] qui proposent de faire collaborer des gyroscopes et de l'odométrie pour les robots mobiles. Ceci dans le but de pallier le fait que les méthodes de fusion utilisent des modèles de mouvement qui ne permettent pas d'anticiper certains mouvements telle que les mouvements larges et brusques.

En ce qui concerne les approches de localisation, la vision a démontré à travers plusieurs travaux qu'elle est capable de fournir une estimation satisfaisante. Cependant, le problème se pose lorsque ce capteur est dans l'incapacité de fournir une estimation cohérente comme par exemple dans le cas d'occultations (partielles ou totales) ou de mouvements brusques (présents dans les applications de type hand-held). Dans ces cas, la vision a besoin d'être suppléée par d'autres capteurs. Donc nous nous retrouvons avec un système comprenant deux sous-systèmes : un sous-système principal et un sous-système de suppléance. Ces deux sous-systèmes fonctionnent de manière complémentaire et par alternance. Le système principal fournit en permanence les mesures. Lorsque ce dernier devient défaillant (i.e. il ne peut plus fournir une mesure correcte) le système de suppléance prend le relais jusqu'à ce que le système principal soit de nouveau fonctionnel. Voici un descriptif de quelques approches qui utilisent ce concept.

3.1.2.1 Aron et al.

Dans les travaux présentés dans [Aron et al., 2007], les auteurs proposent d'utiliser le capteur inertiel pour remplacer la caméra lorsque la méthode basée vision engendre des erreurs. Les justifications énoncées sont :

– Les méthodes de fusion utilisent généralement un filtre de Kalman. Cependant ce type de filtre suppose un mouvement régulier donc il ne prend pas en compte les mouvements brusques et inattendus ;
– Utilisée toute seule, la vision donne des résultats qui sont satisfaisants ;
– Le capteur inertiel n'est pas utilisé systématiquement en raison de sa faible précision qui altère la qualité et la robustesse du suivi ;
– Utiliser ces deux capteurs de manière alternative permet d'éviter à avoir à les synchroniser constamment.

Le système proposé utilise une approche de suivi de plan décrite dans les travaux de [Simon et Berger, 2002] (cf. section.1.2.2.2 page 15). Cependant, selon les auteurs, le système proposé peut utiliser une approche de suivi visuel basé sur n'importe quel type de primitives. De plus, l'approche doit offrir un critère pour évaluer la précision de l'estimation tel que le nombre de primitives de suivi. En effet, dans l'approche proposée, le nombre de bon appariements retournés par RANSAC [Fischler et Bolles, 1981] sert à déterminer la précision de l'estimation de l'homographie (plus nous avons de points, plus l'estimation est précise). Le système bascule de la vision vers le capteur inertiel suivant le nombre de bons appariements obtenus d'une image à une autre. Dans ce cas, le capteur inertiel est utilisé pour prédire la position dans l'image des primitives à suivre. Nous savons que lorsque la caméra réalise une rotation pure, l'image est transformée selon une homographie. Connaissant cette homographie, nous pouvons déterminer la position 2D des pixels dans l'image courante. Cette homographie peut être déduite à partir de la rotation, fournie par la centrale inertielle, exprimée dans le repère caméra. A partir des positions prédites, l'approche définit des zones de recherche sous forme d'ellipses obtenues par propagation des erreurs du capteur inertiel. Les erreurs obtenues dans la phase de calibration sont prises en compte dans une matrice de covariance afin de déduire une ellipse de confiance par approximation linéaire. L'utilisation des orientations relatives permet de corriger les dérives dues à l'accumulation des erreurs de la centrale inertielle.

(a) (b)

FIG. 3.13: (a) Appariement lors d'un mouvement brusque (b) Résultat en milieu extérieur [Aron et al., 2007]

La méthode présente de bons résultats. La précision du système dépend de la précision du capteur utilisé à l'instant t. En effet, lorsqu'à l'instant t, la vision est opérationnelle, la précision du système dépend de la précision du suivi visuel. Dans le cas contraire, la précision du système dépend de la précision des données fournies par la centrale inertielle. La précision de l'approche de vision dépend étroitement du nombre de points utilisés pour l'estimation de l'homographie. Pour ce qui est de la précision des données fournies par la centrale inertielle, les auteurs ont mené plusieurs expérimentations. Pour cela, la centrale

inertielle a été fixée sur une unité mobile ayant deux degrés de liberté et une précision de l'ordre de $0.0013°$. Suite aux différentes séries de mesures effectuées sur chaque axe de rotation, les auteurs remarquent que lorsque la centrale inertielle est fixée horizontalement (son axe z est vertical et pointe vers le haut), les erreurs sur les orientations sont moindres. Cette configuration permet d'obtenir un bruit gaussien sur chacun des axes de moyenne nulle et d'écarts types de l'ordre de $(0.499°, 0.155°, 0.155°)$. Dans le système proposé, le capteur inertiel n'est utilisé que pour estimer un mouvement de rotation. Toutefois, en présence de larges translations, l'ajout d'un capteur pour estimer la position reste primordiale.

3.1.2.2 Maidi et al.

Dans [Maidi et al., 2009], les auteurs proposent une approche de suivi de cibles en combinant différents types de capteurs et de techniques suivant les conditions existantes dans l'environnement. Cette combinaison a pour but de gérer les occultations (partielles ou totales). Pour cela, le système proposé suit un schéma de suppléance pour s'adapter à la situation selon le degré d'occultation des cibles suivies. En effet, les deux capteurs fonctionnent en parallèle. Si les cibles sont visibles et identifiées, la vision permet d'estimer la rotation de la caméra. Dans le cas où les cibles sont occultées, c'est la centrale inertielle qui détermine cette rotation. Ainsi, les deux capteurs se relayent pour estimer l'orientation de la caméra. En ce qui concerne la translation de la caméra, les deux capteurs l'estiment simultanément. De plus, la caméra est utilisée pour corriger les dérives de la centrale inertielle et les données de la caméra sont utilisées pour rectifier la translation de la centrale inertielle.

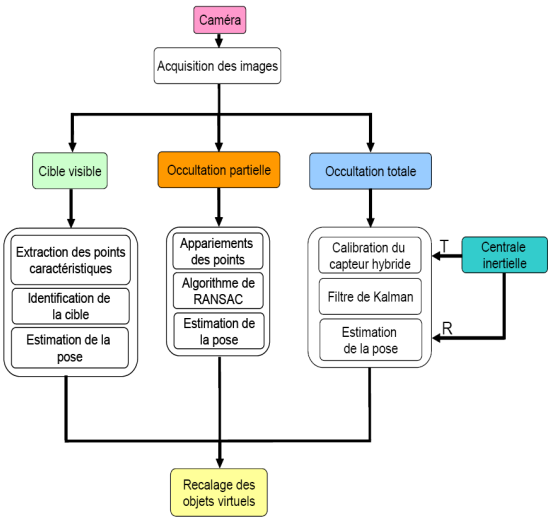


FIG. 3.14: Descriptif du système de localisation [Maidi et al., 2009]

Cette suppléance n'a pas pour but de stabiliser ou d'améliorer la précision du suivi. L'idée est de concevoir un système multimodal afin d'aiguiller le système vers l'approche de suivi adéquate. Ainsi, le système tel qu'il a été conçu comprend trois modules correspondant à trois modes de fonctionnement différents (cf. fig.3.14) : mode d'estimation de pose, mode d'estimation de l'orientation à partir de la centrale inertielle et mode d'estimation de la position à partir des accélérations fournies par la centrale inertielle. Si les cibles utilisées sont totalement visibles, le système estime la position et l'orientation de la caméra en identifiant les cibles et en les appariant avec les positions 3D connues des cibles. Si les cibles sont partiellement occultées, c'est-à-dire ne peuvent pas être identifiées, le système

se base alors sur un suivi basé points d'intérêts pour recouvrer la pose de la caméra en utilisant un calcul d'homographie. Enfin, en cas d'occultation totale (i.e. cible non visible ou lors de mouvement brusque), la centrale inertielle fournit la rotation ainsi que la position.

Les orientations sont fournies par la centrale inertielle à partir des accélérations angulaires. Un filtre de Kalman incorporé dans la centrale inertielle permet d'affiner l'estimation et ainsi d'obtenir des orientations précises. Les positions sont estimées à partir des accélérations en utilisant un filtre de kalman linéaire et un modèle cinématique de la caméra. Le filtre récupère les mesures d'accélérations de la centrale inertielle et réalise une double intégration pour estimer les positions. Lorsque les données du capteur sont disponibles, le filtre effectue une prédiction et une correction des états. La prédiction utilise l'état estimé à l'instant précédent pour prédire une estimation à l'état courant. Puis, lors de l'étape de correction, les observateurs à l'instant courant sont utilisés pour corriger l'état prédit.

Le calcul de la position à partir des accélérations produit des dérives à long terme ce qui met en échec le suivi. Le problème de dérive est causé par l'accumulation des erreurs dues au processus d'intégration. Ce problème peut être corrigé par la caméra mais sous la contrainte d'un court temps de défaillance. Ceci impose l'utilisation d'un autre type de capteur pour le calcul de la position. Certes le système ressemble à celui présenté par dans [Aron et al., 2007]. Cependant, dans les travaux de *Maidi et al.*, le capteur inertiel est utilisé pour la position et l'orientation contrairement à l'approche d'*Aron et al.* En effet, ce dernier calcule uniquement une homographie à partir des orientations. De plus, la procédure de la calibration proposée par *Maidi et al.* diffère de la procédure de *Aron et al.* (cf. chapitre 4). D'après les expérimentations menées, le système proposé par *Maidi et al.* fournit de bons résultats. En effet, testé dans différentes situations, le système a prouvé qu'il pouvait s'adapter et ainsi fournir une pose correcte (cf. fig.3.15). Cependant, comme nous l'avons dit précédemment, à cause de la dérive de la position calculée à partir de la centrale inertielle, le système ne tolère que des occultations qui durent au maximum 600 millisecondes, ce qui est insuffisant.

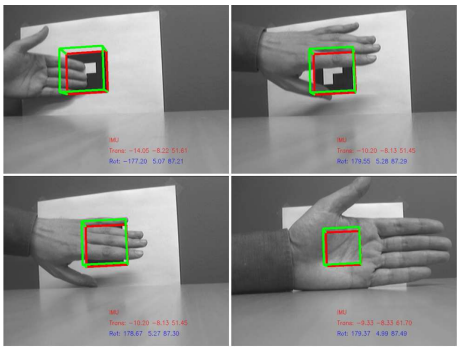


FIG. 3.15: Résultas en situation de semi-occultation et occultation totale [Maidi et al., 2009]

3.2 Synthèse et étude comparative

Dans cette section, nous faisons une synthèse des approches que nous avons vues précédemment. Dans le tableau 3.1, nous retrouvons toutes les approches basées fusion de données. Pour chaque approche, nous nous intéressons aux types de capteur utilisés, le mi-

lieu où a été déployé le système, le principe de l'approche et ce qui fait la particularité de l'approche proposée.

Approche	Capteurs	Milieu	Principes et particularités
You et al.	• Caméra+Gyromètres	Extérieur	• Prédire la position des points avec les gyromètres ; • Recherche locale autour des positions prédites ; • Corriger la dérive des gyromètres avec la vision ; • Suppose une rotation pure pour les points éloignés.
Hol et al.	• Caméra+IMU	Intérieur	• Fusion en utilisant un EKF ; • Modèle de processus basé sur la cinématique ; • Modèle d'observation basé sur les appariements 2D/3D.
Bleser et al.	• Caméra+IMU	Intérieur	• Fusion en utilisant un EKF ; • Modèle d'accélération (englobe aussi un modèle de gyromètres) : en entrée les accélérations, les vitesses angulaires et les appariements de points 2D/3D ; • Modèle de mouvement supposant des accélérations et des vitesses angulaires constantes ;
Ababsa et al.	• Caméra+IMU	Intérieur	• Approche basée marqueurs ; • Fusion en utilisant un filtre particulaire ; • Modèle basé cinématique avec un modèle de mouvement basé accélération constante ; • Filtre complémentaire avec 2 canaux pondérés.
Reitmayr et al.	• Caméra+IMU+GPS	Extérieur	• Fusion en utilisant un EKF (orientation) ; • Modèle de mouvement basé vitesse constante ; • Approche de vision basée contour ; • GPS utilisé uniquement pour l'initialisation ; • Erreur GPS prédite avec un processus Gaussien.
Schall et al.	• DGPS + IMU + Caméra+ Baromètre	Extérieur	• Deux EKF complémentaires : un pour la position et un pour l'orientation ; • Baromètre pour maintenir l'estimation de la hauteur stable ; • Approche de vision sans marqueurs et sans modèle ; • Approche de vision pour corriger les dérives du magnétomètre.

TAB. 3.1: Systèmes Multi-capteurs basés fusion de données : Synthèse

De même, le tableau 3.2 représente une synthèse de ces approches d'un point de vue capteurs, milieu d'utilisation, principes et particularité.

Approche	Capteurs	Milieu	Principes et particularités
Aron et al.	• Caméra + IMU	Intérieur	• Suivi visuel basé plan : calcul d'homographie ; • Prédiction de la position des points suivi à partir d'une homographie estimée avec les orientations de la centrale inertielle ; • Les deux capteurs se relayent.
Maidi et al.	• Caméra + IMU	Intérieur	• La vision et la centrale inertielle se relayent pour estimer la rotation ; • Trois modes de fonctionnements selon la visibilité des marqueurs ; • La translation estimée simultanément ; • La vision corrige la dérive de la position estimée avec la centrale inertielle mais durant un court temps

TAB. 3.2: Systèmes Multi-capteurs basés suppléance de données : Synthèse

Dans le cas général, les approches basées fusion de données utilisent un schéma de prédiction/correction avec un filtre. Le filtre de Kalman est le plus utilisé. Cependant, nous retrouvons quelques travaux utilisant le filtre particulaire [Ababsa et Mallem, 2007] [Bleser et Stricker, 2008b] mais son utilisation reste timide. Les méthodes proposées se rejoignent dans le principe d'utiliser un modèle de mouvement qui se base sur la cinématique de la caméra. Ceci permet de prédire le mouvement effectué par la caméra et ainsi pouvoir extraire les couples d'appariements 2D/3D. Ceux-ci sont injectés dans le processus d'estimation pour corriger le mouvement prédit. Cependant, les modèles de mouvement utilisés supposent que la vitesse est constante ce qui se traduit par des accélérations nulles. Nous trouvons aussi des modèles de mouvement qui supposent que les accélérations sont constantes ainsi que les vitesses angulaires comme dans le modèle proposé par [Bleser, 2009]. Supposer que les vitesses ou les accélérations sont constantes revient à supposer que le mouvement est régulier. Mais ceci n'est pas toujours vérifié dans la pratique. Si cela est vrai pour les applications de robotiques où les mouvements des robots restent assez réguliers, ce n'est pas le cas dans des applications de RA mobiles de type *handheld*. En effet, dans ce type d'application, le dispositif de localisation est généralement porté par l'utilisateur dans les mains. Cantonner le mouvement de l'utilisateur à un mouvement régulier ne permet pas de prendre en compte des mouvements brusques et inattendus.

En ce qui concerne les approches de suppléance, le système comprend une approche basée vision qui est utilisé principalement pour la localisation. En effet, les approches de vision permettent d'estimer la position et l'orientation en retrouvant la relation entre les données 3D et leurs projections dans l'image 2D. L'identification de cette relation ne se base pas sur un modèle de mouvement mais plutôt sur les propriétés géométriques de la caméra (i.e. le modèle de projection). De ce fait, tant que la mise en correspondance 2D/3D est retrouvée et précise, la pose peut toujours être recouvrée. Dans ce cas, la difficulté majeure réside dans la génération de ces appariements au fil du flux d'image. Cependant, les mouvements brusques peuvent altérés les correspondances. De plus, les données caractéristiques

peuvent être occultées ou subir une variation visuelle ce qui met en échec les approches basées vision. Le but est de maintenir l'estimation de la localisation dans toutes les conditions sans avoir à restreindre les mouvements de l'utilisateur. De ce fait, les systèmes basés suppléance ont recours à d'autres types de capteurs pour assister la vision et la remplacer le temps qu'elle redevienne opérationnel. Dans le système de [Aron et al., 2007], lorsque la vision est défaillante, c'est à dire les points suivis sont occultés, l'homographie calculée à partir de l'orientation de la centrale inertielle permet de prédire la position des points dans l'image. Cependant, la translation est négligée. Supposer que le mouvement entre deux images est une rotation pure n'est pas toujours suffisant. Dans [Maidi et al., 2009], les auteurs proposent d'estimer l'orientation à partir de la vision lorsque cette dernière est fonctionnelle et à partir de la centrale inertielle dans le cas contraire. Cependant, cette suppléance ne peut être maintenue sur un long moment d'ou le besoin d'un autre type de capteurs. Nous remarquons que les approches basées suppléance de données n'ont pas été testées dans des milieux extérieurs à grande échelle contrairement aux approches basées fusion de données [Reitmayr et Drummond, 2006] [Reitmayr et Drummond, 2007] [Schall et al., 2009] hormis quelques résultats présentés dans [Aron et al., 2007] mais obtenus à partir d'un point fixe. Il serait intéressant d'explorer la faisabilité d'un tel concept dans un environnement à grande échelle. Ceci permet de voir le comportement d'une approche basée suppléance dans un environnement extérieur en situation de mobilité et ainsi de mesurer son efficacité.

Nous réalisons un comparatif des performances des systèmes présentés ultérieurement (cf. tab.3.3). Ces performances sont exprimées par les erreurs moyennes (μ) ou les erreurs quadratiques (EQM) et le temps d'exécution. Par N.I, nous signifions que la donnée est non indiquée.

	Approches	Position ou translation	Orientation	temps (ms)
Intérieur	Hol et al.	μ =0.05 m	μ =0.1°	40 ms
	Bleser et al.	$\mu_x = 0.26cm$ $\mu_y = 0.26cm$ $\mu_z = 0.27cm$	$\mu_x = 0.57°$ $\mu_y = 0.45°$ $\mu_z = 0.33°$	40 ms
	Ababsa et al.	$\mu = 0.16mm$	$\mu = 0.01°$	33 ms
	Aron et al.	N.I	$\mu_x \in$[1.87°-19.62°] $\mu_y \in$ [1.93°-19.75°] $\mu_z \in$ [2.08°-19.49°]	N.I
	Maidi et al.	EQM_x=1.72e^{-4}m EQM_y=1.73e^{-4}m EQM_z=1.46e^{-4}m	EQM_x=0.3674° EQM_y= 0.127° EQM_z=0.0775°	40 ms
Extérieur	You et al.	N.I	4°	250ms à 100 ms
	Reitmayr et al.	μ_{gps}=0.148m	N.I	La localisation : 40.16ms Initialisation $\geq$ 10000ms
	Schall et al.	μ_x^{gps}=1.058m μ_y^{gps}=0.617m μ_x^{dgps} = 0.8m μ_y^{dgps}= 0.765 m	$\alpha^{IMU} \in$ [0.71°-1.56°] $\alpha^{CAM} \in$ [0.002°-1.95°]	N.I

TAB. 3.3: Comparatif des performances des approches de localisation basées multi-capteurs

Les systèmes présentés utilisent des critères de performance différents. De plus, les conditions d'expérimentations ne sont pas les mêmes. De ce fait, il est difficile d'établir une comparaison entre ces systèmes. La précision de l'estimation de la position et/ou l'orientation dépend fortement de la taille de l'environnement dans lequel évolue la caméra. Certains critères comme l'erreur de reprojection ou de recalage, qui représente un critère important pour un système de réalité augmentée, ne sont pas indiqués par tous les auteurs. Par exemple, dans [Bleser, 2009], les auteurs présentent une erreur moyenne de projection qui varie de 0.93 à 1.42 pixels ce qui est un bon résultat. Cependant celle-ci est obtenue à partir d'un environnement à petite échelle. Par ailleurs, les travaux menés dans des environnements extérieurs comme [Reitmayr et Drummond, 2006], ne donnent pas beaucoup d'indication. Pour ce qui est du système décrit dans les travaux de [Schall et al., 2009], les auteurs présentent une capture d'écran d'un recalage qui a l'air satisfaisant. Toutefois, la précision sur un long moment, n'a pas été indiquée hormis la précision de chaque capteur. Pour ce qui est des méthodes basées suppléance, nous nous ne pouvons pas nous prononcer concernant leur efficacité dans des environnements extérieur. Cependant, les résultats présentés restent similaires à ceux présentés par les méthodes basées fusion de données. Il faut savoir que les performances des systèmes utilisant la suppléance sont fortement liées aux performances des capteurs qui sont utilisés à l'instant courant.

Après avoir vu quelques approches développées dans la communauté, nous allons présenter le système que nous avons mis au point.

3.3 Proposition d'un système de localisation multi-capteurs

Le système de localisation que nous proposons est un système mobile qui devra être porté à la main. Ceci implique que les mouvements de l'utilisateur ne peuvent être supposés réguliers. Ceci nous oriente plutôt vers la seconde approche à savoir la suppléance de donnée. Comme nous l'avons dit auparavant et vu dans le chapitre 2, la vision, à elle seule, peut fournir une bonne estimation de la position et de l'orientation à partir du flux d'images. Le problème se pose lorsque ce capteur ne peut plus fournir une estimation cohérente (occultations des données, dérives de la pose, etc.). Dans ce cas, il est nécessaire de le remplacer par d'autres capteurs.

Suivant un schéma de suppléance, l'idée poursuivie est d'exploiter au mieux les données fournies par tous les capteurs pour ainsi proposer un système adaptatif. En effet, nous proposons un système autonome qui s'adapte à différentes situations rencontrées lors du travail. L'idée est que selon la précision des données disponibles, le système privilégie un des traitements compris dans le système. Ceci dans le but de fournir une bonne estimation de la pose à tout moment. Cela se traduit par le fait que lorsque la vision est opérationnel le système accorde sa confiance aux mesures fournies par les méthodes basée vision. De plus, le système est capable de détecter les défaillances de la vision pour basculer vers le système d'assistance. Certes l'idée de suppléance n'est pas récente vu qu'elle a été déjà proposée dans les travaux de [Aron et al., 2007] et [Maidi, 2007]. Cependant cette stratégie n'a été testée que dans des environnements à petits échelle. Le but que nous nous sommes aussi fixé est de voir le comportement d'une telle stratégie dans des environnements assez larges et de voir le potentiel de son exploitation en extérieur et en situation de mobilité. En effet, les méthodes basées fusion ont été très utilisées dans différents types d'environnement.

Nous voulons par nos travaux proposer une solution "logicielle" qui permet de basculer

entre deux solutions différentes. D'une part, nous avons des méthodes de vision qui ont déjà prouvé leur efficacité. D'autre part, nous proposons un système palliatif qui peut approximer la précision de recalage obtenue avec la vision. Du point de vue matériel, notre système de localisation comprend une caméra utilisée comme capteur principal. Nous rajoutons à cette caméra une centrale inertielle pour estimer les orientations. Même si nous pouvons déduire les positions à partir des accéléromètres, ceci ne peut être fait que durant un court laps de temps ce qui est très insuffisant. De ce fait, pour la position, nous proposons d'utiliser le GPS étant donné que ce capteur est adapté aux environnements extérieurs. La caméra et la centrale inertielle sont rigidement liées et le GPS sera porté par l'utilisateur. Notre système est composé de deux parties qui vont interagir en continu l'une avec l'autre à la différence du système proposé par *Aron et al.* où le capteur inertiel est utilisé seulement dans le cas où le suivi visuel est en échec. Les données fournies par certain capteurs peuvent être utilisés pour valider les estimations fournies par le reste des capteurs. De plus, les mesures fournies par une partie des capteurs peuvent être corrigées en utilisant des données fournies par les autres capteurs. Pour cela, notre système est subdivisé en deux sous-systèmes (cf. fig.3.16) : un sous-système de **vision** et un sous-système que nous appelons **Assistance à la localisation**. Le système multi-capteurs estime en continu la position et l'orientation du point de vue même si la vision échoue.

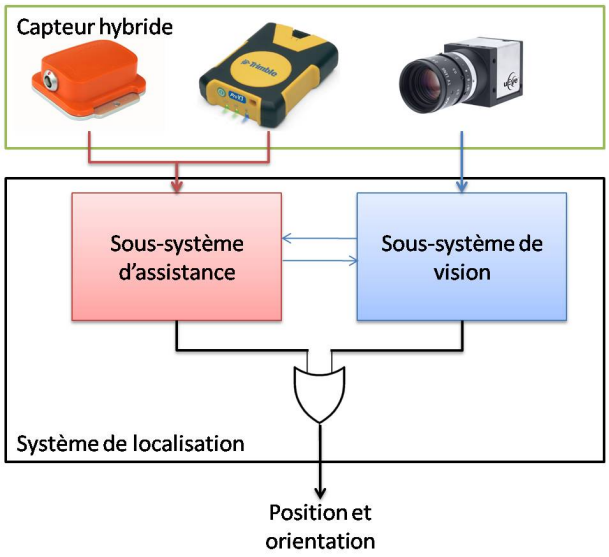


FIG. 3.16: Vue globale du système de localisation

Dans ce qui suit, nous allons décrire les différentes problématiques qui entourent la mise en œuvre d'un tel système.

3.4 Enjeux et problématiques

Différentes problématiques complémentaires entourent la mise en œuvre de notre approche. Ces problématiques constituent des verrous scientifiques et technologiques importants à lever. Voici un aperçu de ce que nous avons traité.

3.4.1 Calibration du capteur hybride

Chaque capteur fournit des mesures dans un système de coordonnées qui lui est propre. Notre besoin en localisation et recalage consiste à déterminer la position et l'orientation du point de vue, c'est-à-dire la caméra, dans l'environnement. Pour cela, nous devons déterminer la relation existante entre le repère associé à la caméra et le repère associé au monde réel selon le modèle de projection.

Il faut donc exprimer les mesures fournies par les autres capteurs, la centrale inertielle et le GPS, par rapport au repère de la caméra. Ceci se traduit par l'identification des transformations qui permettent le passage d'un repère à un autre. Cette procédure dite de calibration est importante car la précision du capteur hybride dépend de la précision de cette procédure.

3.4.2 Localisation et suivi basée vision

Etant donné que la vision est le capteur principal, il nous faut mettre au point une approche basée vision qui fournit une localisation précise et robuste face aux conditions de travail en extérieur. L'approche sera sans marqueur et utilise les données naturelles existantes dans l'environnement. En effet, il est impossible d'instrumenter l'environnement de travail avec des marqueurs artificiels. Cette approche se base sur une connaissance *a priori* de l'environnement constitué de modèles 3D.

Différent défis sont à relever. Le premier réside dans le choix des primitives à utiliser et qui doivent être en adéquation avec l'environnement extérieur. En vision, l'estimation de la pose est obtenue à partir de la relation existante entre des primitives 3D extraites du modèle 3D et leurs projections 2D extraites des images. Cette mise en correspondance de deux entités n'est pas aisée. La précision de l'estimation basée vision dépend énormément de cette phase d'appariement 2D/3D.

Cet appariement doit être maintenu pour chaque nouvelle image. Un suivi 2D de primitives est indispensable ce qui permet d'éviter de réaliser l'appariement 2D/3D directement mais plutôt de le déduire à partir du suivi 2D. Etant donné son importance, le suivi 2D doit être essentiellement robuste aux changements d'apparences des primitives suivies.

De plus, en raison de son caractère de suppléance, le système de localisation doit être capable de détecter les défaillances du sous-système de vision. Ceci permettra au système de localisation de basculer vers le sous-système d'assistance.

La solution proposée est présentée dans le chapitre 2. Cependant, nous présenterons quelques traitements complémentaires pour le fonctionnement dans notre système global.

3.4.3 Prédiction d'erreur et correction

Les capteurs composant le sous-système d'assistance à la localisation sont moins précis que la vision. En effet, la précision du GPS dépend du nombre de satellites qui couvrent la zone de travail et qui sont utilisés pour la triangulation. De plus, le signal peut être altéré (cf. annexe D). De plus, la centrale inertielle peut être perturbée par les objets composant l'environnement (influence du champ magnétique par exemple). Lorsque la vision est défaillante et que le système de localisation bascule vers le sous-système AL, nous avons besoin d'une estimation qui soit proche de la localisation réelle. Afin de faire converger cette estimation

vers la position et l'orientation réelle, nous avons besoin de connaitre l'erreur commise pour ainsi corriger et raffiner cette estimation. Cette erreur représente le décalage obtenu entre les deux sous-systèmes.

3.5 Conclusion

Le processus de localisation est un élément important d'un système de réalité augmentée. Il permet d'instrumenter de manière cohérente la vue réelle par le virtuel. Ce processus devient difficile à réaliser lorsqu'il s'agit de travailler dans un environnement extérieur où il est difficile d'avoir un contrôle sur les événements et les conditions qui y règnent. Les méthodes basées vision, longtemps privilégiées dans les applications de réalité augmentée essentiellement en intérieur, ne sont plus suffisantes toutes seules. Pour cela, les applications de réalité augmentée dédiées aux environnements extérieurs ont convergé vers des systèmes multi-capteurs afin de pallier les inconvénients de l'utilisation d'un seul capteur.

Comme nous l'avons vu dans la section 3.1 (page 65), plusieurs travaux s'orientent vers une approche de fusion de données. Ces approches se basent sur des filtres (essentiellement le filtre de Kalman) qui supposent que le modèle de mouvement est à vitesse constante. Ceci revient à supposer que le mouvement est régulier. Cependant ceci n'est pas toujours vérifié. De ce fait, ces modèles ne gèrent pas les mouvements brusques et inattendus. En effet, dans des applications mobiles où le système est généralement porté par l'utilisateur, les mouvements de ce dernier sont aléatoires. De plus, définir un modèle avec une vitesse variable n'est pas chose aisée alors qu'avec les approches de vision nous n'avons pas besoin de faire de supposition sur la forme du mouvement. La détermination de la pose s'appuie uniquement sur la relation entre le monde réel et l'image. Par ailleurs utiliser uniquement la vision peut être auto-suffisant en général. Seulement, dans le cas où la vision ne peut pas fournir cette estimation, elle a besoin d'une assistante qui prend sa place en attendant qu'elle devienne opérationnelle à nouveau.

Pour ces raisons, nous nous sommes orientés vers une stratégie de suppléance de données. En effet, notre application est une application mobile qui est totalement soumise aux mouvements de l'utilisateur. L'objectif est d'avoir à tout moment une estimation de la pose de la caméra pour effectuer un recalage cohérent du réel et du virtuel. Pour cela, notre système comprend une caméra qui est suppléée par le couplage d'un GPS et d'une centrale inertielle. La centrale inertielle est uniquement utilisée pour remplacer la caméra dans l'estimation de l'orientation. La position est, quand à elle, fournie par le GPS dans le but de pallier la dérive rapide lors de l'estimation de la position à partir des accélérations.

Nous nous sommes attelés à concevoir un système dont les composantes fonctionnent de manière complémentaire et interagissent entre elles. Le but principal est de disposer à tout moment d'une estimation de la pose la plus précise possible quelque soient les conditions.

Dans la réalisation de notre système, différentes problématiques exposées dans la section 3.4 (page 86) ont été prises en compte. Dans le chapitre 2, nous avons déja présenté une partie de notre sous-système de vision en exposant notre approche sans marqueurs. Dans les chapitres qui suivent, nous allons détailler les solutions que nous proposons pour le reste des problématiques identifiées précédemment (cf. section 3.4 age 86). Ainsi dans le chapitre suivant (chapitre 4), nous nous intéressons au problème de la calibration qui permet de définir la relation existante entre les différents capteurs utilisés.

Chapitre 4

Modélisation et Calibration de Capteur Hybride

Par définition, la calibration, aussi appelée étalonnage, consiste à valider et vérifier l'acuité et la précision de mesures spécifiques obtenues à différentes amplitudes et/ou précisions en les comparant à d'autres mesures de références. En pratique, la calibration consiste à déterminer les paramètres et caractéristiques propres à un capteur en se basant sur un modèle décrivant le processus d'acquisition des mesures. Par exemple, la procédure de calibration de caméra se base sur le modèle de formation de l'image. Cette procédure de calibration revient à déterminer la relation spatiale entre l'environnement réel et l'image obtenue de cet environnement.

Dans ce chapitre, nous nous intéressons à la modélisation et la calibration d'un capteur hybride combinant plusieurs types de capteurs. En supposant que chaque capteur pris séparément est calibré (pour la caméra voir annexe A, la centrale inertielle et le GPS sont calibrés par le constructeur), modéliser le capteur hybride consiste à modéliser les transformations reliant les différents repères locaux associés à chaque capteur. En effet, chacun fournit des mesures dans un référentiel qui lui est propre. La procédure de calibration revient à identifier ces transformations. Ceci permet d'exprimer les données fournies par un capteur dans son propre repère dans le repère d'un autre capteur et ainsi d'uniformiser les différentes mesures afin de les exprimer dans un repère commun.

La précision de la combinaison dépend de la précision de la calibration. Comme nous l'avons vu, notre système de localisation combine trois types de capteurs différents, une caméra, un GPS et une centrale inertielle. Le processus de localisation consiste à déterminer la position et l'orientation du point de vue, c'est-à-dire de la caméra, par rapport au repère associé à la scène. De ce fait, les données fournies par la centrale inertielle et par le GPS doivent être exprimées dans le repère associé à la caméra. Dans ce qui suit, nous présentons deux processus de calibration : un processus de calibration Inertiel/Caméra et un processus de calibration GPS/Caméra.

Le processus de calibration Inertiel/Caméra permet de définir une transformation rigide entre le repère associé à la centrale inertielle et le repère de la caméra. A partir de cette transformation, l'orientation de la caméra peut être déduite à partir de l'orientation don-

née par la centrale inertielle. Concernant la calibration GPS/Caméra, la procédure permet d'identifier la transformation qui permet d'estimer la position de la caméra dans l'environnement à partir de la position GPS.

Dans un premier temps, nous allons nous intéresser aux approches déjà proposées dans la littérature. Ceci nous permettra de réaliser un comparatif entre elle. Par la suite, nous présentons notre processus de calibration Inertiel/Caméra. La seconde partie de ce chapitre concerne le couplage GPS/Caméra. Dans cette partie, nous détaillons le processus de calibration que nous avons mis au point pour ce couplage. Le reste du chapitre comporte des expérimentations et évaluations effectuées pour quantifier les performances de nos procédures de calibration.

4.1 Calibration Inertiel/Caméra : état de l'art

Plusieurs travaux se sont penchés sur la question de combiner une caméra avec une centrale inertielle. Parmi les problématiques traitées, nous trouvons des approches qui permettent de calibrer ce type de combinaison. Les approches proposées sont connues sous le nom de méthodes *Hand-eye*.

La calibration *Hand-Eye* est définie comme le calcul simultané de deux relations spatiales inconnues dans un cercle de relations spatiales. La calibration *hand-eye* est apparue dans la communauté de robotique. Elle tient son nom du fait qu'on utilisait une caméra qui représente l'œil (*eye*) montée sur la pince d'un robot qui représente la main (*hand*). Le but est d'identifier la relation spatiale entre le repère de la caméra et le repère de la pince du robot. A cela s'ajoute la relation entre le repère du robot avec le repère monde.

En ce qui concerne les approches de calibration Inertiel/Caméra, la procédure est effectuée avec des données issues de la vision et des données fournie par la centrale inertielle. Voici un descriptif de quelques approches proposées dans la littérature.

4.1.1 You et al.

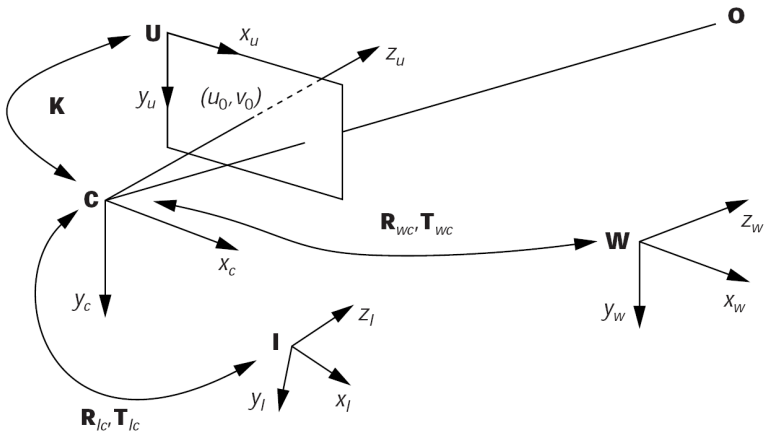


FIG. 4.1: Configuration des systèmes de coordonnées utilisés dans le capteur hybride proposé par [You et al., 1999]

Nous avons vu dans la section 3.1.1.1 (page 67) que, dans [You et al., 1999], les auteurs proposent de fusionner les données de la centrale inertielle avec un suivi visuel pour augmenter les performances et corriger la dérive de la centrale inertielle. Dans l'approche décrite, les auteurs proposent une approche de calibration basée mouvement pour calculer la transformation entre les systèmes de coordonnées de la caméra C et de la centrale inertielle I. La transformation rigide (R_{IC}, T_{IC}) entre les deux capteurs est définie par la relation :

$$\begin{pmatrix} x_C \\ y_C \\ z_C \end{pmatrix} = R_{IC} \begin{pmatrix} x_I \\ y_I \\ z_I \end{pmatrix} + T_{IC} \tag{4.1}$$

Si ω_C et ω_I représentent les vitesses angulaires des points de la scène définies respectivement dans le repère caméra C et le repère inertiel I, la relation entre les deux repères est donnée par :

$$\omega_C = R_{IC}\omega_I \tag{4.2}$$

Le mouvement angulaire ω_I est fourni par la centrale inertielle alors que les vitesses angulaires de la caméra ω_C doivent être estimées. Le mouvement de la caméra se décompose en une translation linéaire et un mouvement angulaire. A partir du modèle sténopé, on peut déduire le mouvement 2D d'un ensemble de caractéristiques images. De ce fait, connaissant les paramètres intrinsèques de la caméra, les données ω_I et le mouvement 2D de la caméra, la matrice R_{IC} est estimée à partir de l'équation suivante :

$$\dot{X}_u = \Lambda R_{IC}\omega_I \tag{4.3}$$

Avec $\dot{X}_u = (\dot{x}_u, \dot{y}_u)$ la vitesse d'un point (x_u, y_u) dans le plan image et Λ est la matrice extraite du modèle de mouvement de la caméra. Les auteurs présentent une erreur d'alignement de l'ordre de 4.27 pixels ce qui se traduit par une erreur de rotation d'approximativement de $0.4°$.

4.1.2 Alves et al.

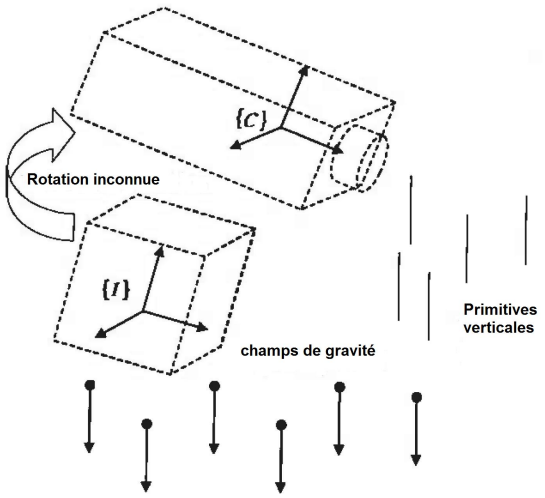


FIG. 4.2: Inertiel/Camera observant la direction verticale [Alves et al., 2004]

Nous retrouvons dans les travaux de [Alves et al., 2004] ainsi que [Lobo et Dias, 2007] une approche de calibration qui suppose que la caméra et la centrale inertielle observent

toutes les deux la direction verticale (cf. fig.4.2). En effet, lorsque la centrale inertielle est dans un état stationnaire, elle mesure l'accélération due à la gravité. De ce fait, la centrale inertielle fournit la direction verticale. Concernant la caméra, en utilisant un damier (une mire) de calibration placé verticalement ou une scène avec comme contours prédominants des contours verticaux, la direction verticale peut être détectée en utilisant des points de fuite (*vanishing point*). Ces points correspondent aux intersections des lignes parallèles dans l'image. A partir de plusieurs observations à différentes positions, l'orientation absolue caractérisant la relation entre les deux capteurs est estimée utilisant la méthode de Horn [Horn, 1987] qui propose une solution analytique en utilisant les quaternions. Les expérimentations effectuées donnent une erreur moyenne de l'ordre de 0.219° avec une variance $\sigma^2 = 1.5699^\circ$.

4.1.3 Lang et Pinz

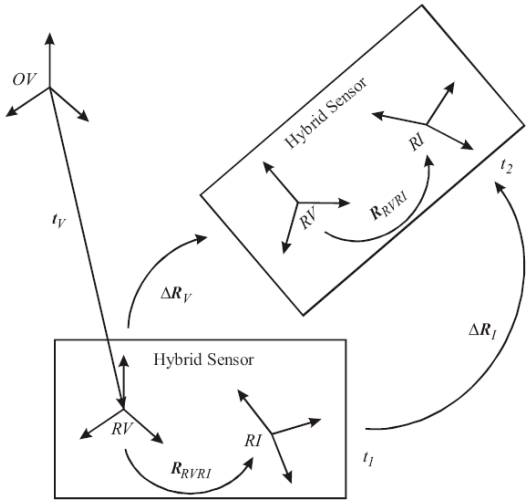


FIG. 4.3: Approche de calibration décrite par [Lang et Pinz, 2005]

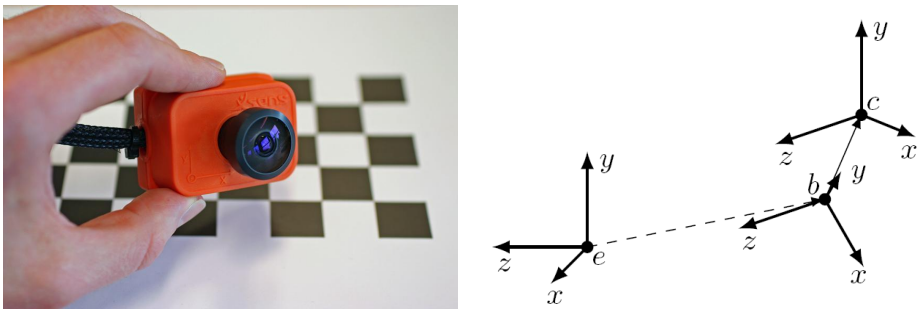
Lang et Pinz proposent dans [Lang et Pinz, 2005] d'estimer la rotation entre les deux capteurs combinés en se basant sur la différence de rotations. En d'autres termes, la méthode utilise le mouvement rotationnel relatif entre deux poses différentes. D'une part, nous avons la rotation ΔR_V exprimée dans le repère caméra *RV* effectuée entre deux instants différents (t_1 et t_2) (cf. fig.4.3). D'autre part, ΔR_I est la rotation relative réalisée par le repère de la centrale inertielle *RI* entre deux instants différents. R_{RVRI}, la rotation entre les deux capteurs, est estimée en minimisant l'erreur entre l'ensemble des mesures $(\Delta R_{V_i}, \Delta R_{I_i})$ en exploitant la relation suivante :

$$\Delta R_I = R_{RVRI}^{-1} \Delta R_V R_{RVRI} \tag{4.4}$$

Les expérimentations effectuées par les auteurs présentent une erreur moyenne de l'ordre de 0.39° en configuration **outside-in** et de 0.85° en ce qui concerne la configuration **inside-out**.

4.1.4 Hol et al.

L'approche proposée dans [Hol et al., 2008] consiste à estimer la translation et l'orientation relatives entre la centrale inertielle et une caméra sphérique (cf. fig.4.4-a). Les sys-



(a) Capteur hybride (b) Systèmes de coordonnées

FIG. 4.4: L'approche de *Hol et al.*

tèmes de coordonnées utilisés dans ce système représentés dans la figure 4.4-b comprennent :

- Le repère **e** qui correspond au repère associé au monde ou la terre. La pose de la caméra est estimée vis à vis de ce repère ;
- Le repère **c** qui correspond au repère attaché à la caméra en mouvement ;
- Le repère **b** qui correspond au repère attaché à la centrale inertielle.

Etant donné que la caméra et la centrale inertielle sont attachées rigidement, la calibration estime a^b et φ^{ab} qui sont respectivement, la position relative de la centrale dans le repère caméra et la rotation entre le repère de la centrale et le repère caméra. L'idée est d'estimer les paramètres de la pose relative notés Θ à partir de $Z = u_1, ..., u_m, y_1, ..., y_n$ tel que les u_i sont les données fournies par la centrale inertielle et y_j les mesures fournies par la vision. L'estimation de ces paramètres se base sur une approche de prédiction d'erreurs définie par :

$$\varepsilon_t(\Theta) = y_t - \hat{y}_{t|t-1}(\theta) \tag{4.5}$$

Où $\hat{y}_{t|t-1}(\theta)$ est la prédiction de l'erreur. Les paramètres sont estimés en minimisant la différence entre les mesures estimées et les mesures prédites. La grandeur à minimiser est la norme des erreurs de prédiction :

$$V_N(\theta, z) = \frac{1}{N}\sum_{t=1}^{N}\frac{1}{2}\varepsilon_t^T\theta\Lambda_t^{-1}\varepsilon_t(\theta) \tag{4.6}$$

Λ est ici la matrice de covariance obtenue avec le modèle de prédiction de l'erreur. Le modèle de prédiction se base sur le mouvement de la caméra déduit à partir des mesures fournies par la vision. Ces mesures représentent des appariements entre les points 2D extraits de l'image et leurs correspondants 3D dans le monde réel. Pour simplifier ce processus, les auteurs se basent sur une mire de calibration. Au final, un filtre de Kalman étendu est utilisé pour la prédiction. Selon les résultats présentés dans [Hol et al., 2008], l'estimation de l'orientation avec cette approche est obtenue avec une précision inférieure à $0.49°$. Pour ce qui est de la translation, celle-ci varie entre $0.3mm$ et $3.6mm$.

4.1.5 Aron et al.

Dans les travaux d'[Aron et al., 2007], les auteurs proposent d'estimer la matrice de rotation X (cf. fig.4.5) qui permet de déduire les mouvements de rotations relatives de la caméra notées B à partir des matrices de rotations relatives A fournies par la centrale inertielle. La relation existante entre ces trois rotations est donnée par :

$$AX = XB \tag{4.7}$$

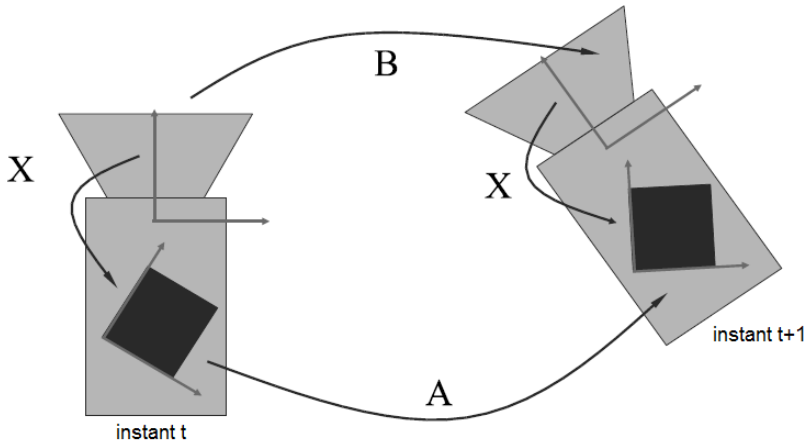


FIG. 4.5: Configuration utilisée dans [Aron et al., 2007] pour la calibration Hand-eye du capteur Inertiel/Caméra

A partir de cette équation (cf. eq.4.7), les auteurs estiment la rotation X en utilisant la méthode décrite par Park et Martin [Park et Martin, 1994]. L'approche présente une solution analytique au problème $AX = XB$ comme étant un ajustement linéaire des moindres carrés. Cette approche est simple et n'exploite pas d'hypothèses restrictives. La connaissance de cette rotation et de l'orientation relative donnée par la centrale inertielle permet de déduire l'homographie qui permet de déduire la pose courante (cf. section.3.1.2.1 page 77).

4.1.6 Reitmayr et Drummond

Dans le système proposé par [Reitmayr et Schmalstieg, 2003], les données de la centrale inertielle ont besoin d'être transformées dans le repère associé à la caméra afin de les fusionner. A chaque rotation caméra R_{CW}, orientation du repère caméra C dans le repère monde W, correspond une rotation $R_{SW'}$ qui définit l'orientation de la centrale inertielle dans son propre repère monde noté W'. La relation existante entre les deux rotations est donnée par :

$$R_{CW} = R_{CS} R_{SW'} R_{W'W} \tag{4.8}$$

La transformation reliant la caméra avec la centrale inertielle est subdivisée en deux rotations : une rotation R_{CS} entre le repère local de la centrale inertielle S et le repère associé à la caméra C et une rotation $R_{W'W}$ entre le repère monde W et le repère dit monde de la centrale inertielle W'. Les deux rotations sont estimées en utilisant la méthode de [Baillot et al., 2003].
Dans cette approche, la détermination des deux rotations se base sur le calcul de mouvements relatifs. En effet, dans la configuration des repères, nous avons les repères C et S qui sont rigidement liés, idem pour les repères W et W'. Entre deux instants donnés, nous pouvons obtenir les mouvements (i.e. rotations) relatifs effectués par chaque repère. Ainsi pour chaque configuration rigide, nous nous retrouvons dans le cas décrit dans [Aron et al., 2007]. Pour chaque rotation, nous définissons un système sous la forme $AX = XB$. Pour réaliser la phase de calibration, l'utilisateur doit se positionner à un endroit prédéfini et aligner la vue avec des objets connus dans l'environnement.
Dans les résultats présentés par [Baillot et al., 2003], l'erreur présentée sur l'estimation de $R_{W'W}$ est entre $0.05m$ (en X et Z) et $0.15m$ (en Y) en position. En orientation, l'erreur obte-

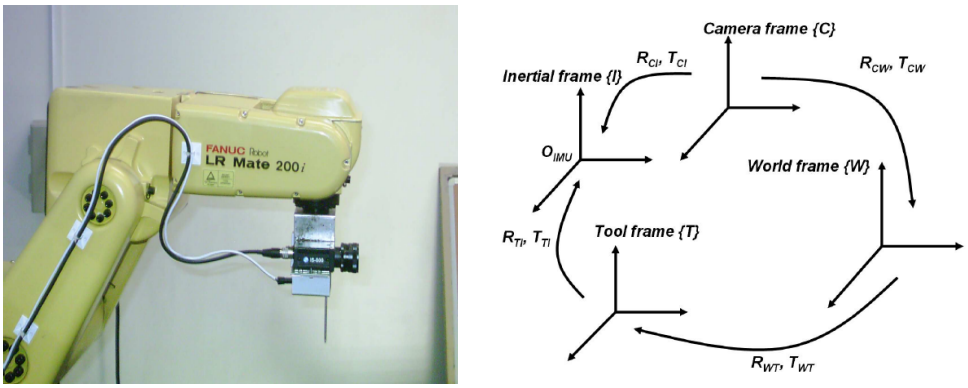
nue est entre 0.5° (en X et Y) et 1.2° (en Z). En ce qui concerne la rotation R_{CS}, les erreurs obtenues sont similaires avec une erreur en position entre $0.35m$ et $0.7m$ et une erreur en orientation entre 0.4° et 1.2°. Pour information, les rotations sont estimées en utilisant 7 positions de calibration.

4.1.7 Maidi et al.

Dans leurs travaux, *Maidi et al.* présentent deux approches pour calculer la relation entre la caméra et la centrale inertielle. La première approche utilise un bras robotique comme référentiel. La seconde approche exploite une fonctionnalité proposée par la centrale inertielle utilisée.

4.1.7.1 Première approche

Dans [Maidi et al., 2005], les auteurs utilisent un robot à 6 degrés de liberté pour exploiter l'exactitude du positionnement du robot dans le processus de calibration (cf. fig.4.6-a). Le robot utilisé est un bras manipulateur qui est articulé avec 6 axes. Son outil, qui représente le sixième axe, est le référentiel de mouvement utilisé. En effet, le robot fournit la position et l'orientation de son outil par rapport au référentiel utilisateur (par défaut, c'est la base du robot). De ce fait, pour déterminer la position d'un nouveau repère outil dans un nouveau repère utilisateur, il est nécessaire d'effectuer une phase d'apprentissage des nouveaux repères (outil et utilisateur).



(a) (b)

FIG. 4.6: (a) Capteur hybride monté sur un bras de robot (b) Représentation des rotations entre les repères utilisés dans [Maidi et al., 2005]

La transformation qui fait correspondre le repère associé à la centrale inertielle$\{I\}$ avec le repère caméra $\{C\}$ est composée de trois rotations : R_{CW} rotation du repère monde $\{W\}$ par rapport au repère associé à la caméra $\{C\}$, R_{WT} la rotation du système de coordonnées associé à l'outil $\{T\}$ par rapport au repère monde $\{W\}$ et enfin, R_{TI} la rotation du repère inertiel $\{I\}$ par rapport au repère associé à l'outil $\{T\}$. Ce qui donne :

$$R_{CI} = R_{CW}R_{WT}R_{TI} \tag{4.9}$$

En ce qui concerne la translation T_{CI}, il est important de connaître exactement la position du repère de la centrale inertielle par rapport au repère de l'outil noté T_{TI}. Cette translation

est calculée à partir des mesures effectuées. Une fois T_{TI} connue, nous pouvons calculer T_{WI} tel que :

$$T_{WI} = R_{WT} T_{TI} + T_{WT} \tag{4.10}$$

Ceci permet d'estimer la translation entre les repères des deux capteurs qui est exprimée comme suit :

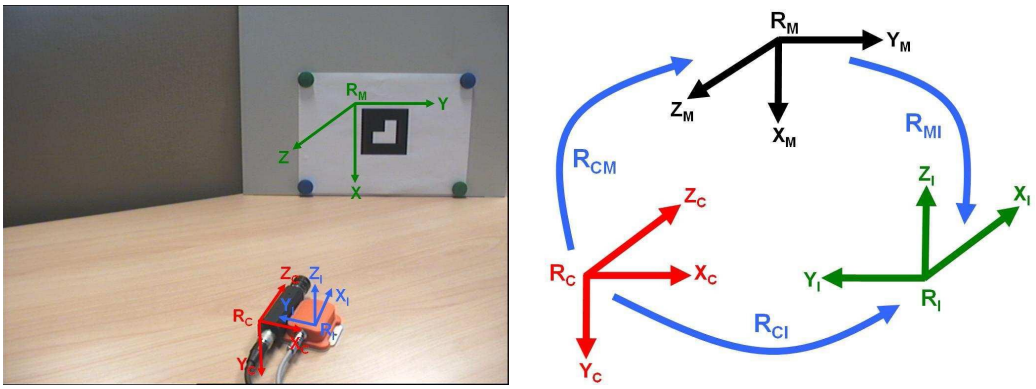
$$T_{CI} = R_{CW} T_{WI} + T_{CW} \tag{4.11}$$

Les résultats obtenus par les auteurs présentent une erreur moyenne en rotation de l'ordre de $(MRE_\psi = 0.32°, MRE_\theta = 0.30°, MRE_\phi = 0.18°)$ et une erreur en translation $(MTE_x = 1.5mm, MTE_y = 1.5mm, MTE_z = 1.2mm)$.

4.1.7.2 Deuxième approche

La deuxième approche présentée dans [Maidi et al., 2009] permet de se détacher de l'utilisation d'un bras de robot pour effectuer la calibration du couplage Inertiel/Caméra.

Dans le système présenté par [Maidi et al., 2009] (cf. fig.4.7-a), le système de vision permet d'estimer la rotation notée R_{CM} du repère associé au monde R_M par rapport au repère caméra R_C. Simultanément, la centrale inertielle fournit des rotations R_{MI} du repère R_I local attaché à la centrale inertielle dans le repère monde R_M. Cependant, par défaut, les rotations de la centrale inertielle sont exprimées par rapport à un repère global qui est défini par le repère du champ magnétique terrestre. Néanmoins, il est possible de redéfinir ce repère. En effet, la centrale inertielle utilisée dispose d'une procédure d'initialisation pour définir un autre repère global.



(a) (b)

FIG. 4.7: (a) Représentation des repères utilisés (b) Représentation des rotations entre les repères utilisés dans [Maidi et al., 2009]

En redéfinissant le repère global comme étant le repère monde, et à partir de cette configuration de système de coordonnées (cf. fig.4.7-b), la relation entre le repère de la caméra et celui de la centrale inertielle est représenté par une rotation R_{CI} exprimée par :

$$R_{CI} = R_{CM} R_{MI} \tag{4.12}$$

La rotation R_{CI} est calculée en utilisant les quaternions car ils ne présentent pas de singularités et donnent des solutions uniques pour l'orientation. La solution finale est donnée

à partir d'une linéarisation autour du quaternion moyen :

$$Q_{CI} = \frac{1}{N}\sum_{1}^{N} Q_{CI_i} \tag{4.13}$$

La seconde approche proposée par *Maidi et al.* présente des erreurs en rotation de l'ordre de $(MRE_\psi = 0.3674°, MRE_\theta = 0.1274°, MRE_\phi = 0.0775°)$

4.1.8 Bleser et al.

Dans les travaux présentés dans [Bleser, 2009], l'auteur propose de calibrer le système multi-capteurs en se basant sur la configuration donnée à la figure 4.8.

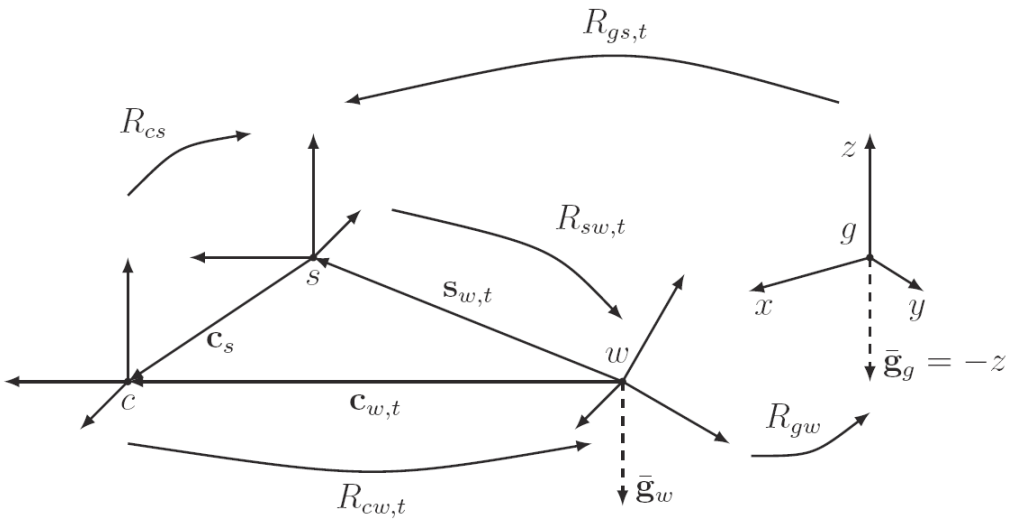


FIG. 4.8: Systèmes de coordonnées utilisées dans [Bleser, 2009]

Sachant que les deux capteurs (caméra et centrale inertielle) sont rigidement liés, la relation liant les deux capteurs qui permet de convertir la pose fournie par la centrale inertielle en une pose caméra est donnée par les équations :

$$R_{cw,t} = R_{cs} R_{sw,t} \tag{4.14}$$

$$c_{w,t} = s_{w,t} + R_{ws,t} \times c_s \tag{4.15}$$

Où $R_{cw,t}$ (resp $R_{sw,t}$) est la rotation entre le repère caméra c (resp le repère centrale inertielle s) et le repère monde w à l'instant t. Aussi, $c_{w,t}$ (resp $s_{w,t}$) représente la translation entre le repère caméra c (resp repère centrale inertielle s) et le repère w. La relation rigide entre deux capteurs est donnée par la rotation R_{cs} et la translation c_s entre le repère c et le repère de la centrale inertielle s. Le repère local de la centrale inertielle s est fixé à la centrale inertielle et les orientations fournies par celle-ci sont exprimées dans le repère global g de la centrale. Ce repère global est défini tel que l'axe des x pointe vers le nord magnétique local et son axe z dans le sens opposé de la gravité.

La rotation R_{cs} est calculée à partir d'un ensemble de vecteurs de gravité $(g_{c,t}, g_{s,t})$ tel que :

$$g_{c,t} = q_{cs} \odot g_{s,t} \odot q_{sc} \tag{4.16}$$

Où q_{cs} et q_{sc} sont les orientations sous forme de quaternion et $\odot$ le produit quaternion. La solution optimale est obtenue à partir du vecteur propre correspondant à la plus grande valeur propre à partir de la somme des carrés résiduels :

$$\sum_{t=1}^{N} \|g_{c,t} - q_{cs} \odot g_{s,t} \odot q_{sc}\| \tag{4.17}$$

Où $g_{c,t} = R_{cw,t} g_w$. Lors de la phase de calibration, la mire utilisée est placée de sorte que l'axe z du repère w est vertical donc dans le sens inverse de la gravité ce qui donne $g_w = g_g$. La translation c_s représente la distance entre l'origine du repère associé à la caméra c par rapport à l'origine du repère associé à la centrale inertielle s. Celles-ci sont fournies respectivement par la calibration de la caméra et par le constructeur.

Il n'est pas suffisant d'estimer (R_{cs}, c_s). En effet, il faut aussi déterminer R_{gw}. En connaissant R_{cs}, cette rotation est obtenue en calculant la moyenne à partir d'un ensemble de mesure $R_{gs,t}$ et $R_{cs}^T R_{cw,t}$.

4.2 Synthèse et étude comparative

Dans ce qui a précédé, nous avons présenté quelques approches qui s'intéressent à la calibration du couplage centrale inertielle et caméra. Maintenant, nous allons effectuer une synthèse de ces approches. Le tableau suivant (cf. tab.4.1) présente les différentes approches, leur principe, leur précision et leurs performances obtenues. Il faut noter que par N.I, nous voulons dire Non Indiqué.

Approche	Principe	Précision
You et al.	Utilise la relation entre les vitesses angulaires	0.4°
Alves et al. et Lobo et Dias	Suppose que les 2 capteurs observent la direction verticale	0.219°
Lang et Pinz	Relation basée sur la différence de rotations (i.e. Mouvement rotationnel relatif)	0.85°
Hol et al.	Utilise un modèle de prédiction avec un EKF	0.49°
Aron et al.	Relation sous la forme $AX = XB$ basée sur le mouvement rotationnel relatif	N.I
Reitmayr et Drummond	Relation sous forme $AX = XB$ en divisant en deux sous-systèmes de coordonnées liés rigidement	0.5° en X 0.5° en Y et 1.2° en Z
Maidi et al (1)	Utilise un bras de robot	$(MRE_\psi = 0.32°, MRE_\theta = 0.30°, MRE_\phi = 0.18°)$
Maidi et al (2)	Redéfinit le repère monde comme étant le repère globale de la centrale inertielle	$(MRE_\psi = 0.36°, MRE_\theta = 0.12°, MRE_\phi = 0.07°)$
Bleser et al.	Basée sur la relation entre les vecteurs de gravité	N.I

TAB. 4.1: Comparatif des approches de Calibration Inertiel/Camera

Tout d'abord, nous constatons que du point de vue de la précision, les approches sont équivalentes avec des erreurs qui varient entre 0.07° et 1.2°. La différence entre ces approches réside essentiellement dans les hypothèses utilisées pour calculer la relation entre les deux capteurs.

Nous remarquons que plusieurs travaux se basent sur le calcul du mouvement relatif des deux capteurs pour estimer la relation existante entre eux. Nous retrouvons ce principe dans les approches de [Lang et Pinz, 2005], [Aron et al., 2007] et [Reitmayr et Drummond, 2006]. Les approches décrite dans [Aron et al., 2007] et [Reitmayr et Drummond, 2006] se basent sur le même principe qui consiste à formaliser les relations entre les deux capteurs sous la forme $AX = XB$. Ces approches mesurent le mouvement relatif de chaque capteur entre deux instants différents. Une fois le couplage Inertiel/Caméra calibré, le mouvement rotationnel de la caméra peut être déduit du mouvement rotationnel fourni par la centrale inertielle. Ceci peut être intéressant pour les approches qui se basent sur l'estimation du mouvement pour mettre à jour l'estimation prédite. Cependant, ce type d'approche est souvent sujet à des dérives causées par l'accumulation des erreurs au fil du temps.

Nous retrouvons aussi d'autres méthodes qui se basent sur les vitesses angulaires comme l'approche proposée dans [You et al., 1999] ou sur les vecteurs de gravité comme l'approche proposée par [Bleser, 2009]. Cependant, si les vitesses angulaires sont fournies par la centrale inertielle, pour la vision il faut les calculer en se basant sur la dérivée du modèle sténopé et faire de même pour les vecteurs de gravité.

La première approche proposée par [Maidi et al., 2005] propose d'utiliser un bras de robot. Ceci impose aussi une phase d'apprentissage des repères outil et utilisateur. En ce qui concerne la seconde approche [Maidi et al., 2009], celle-ci suppose que le repère global de centrale inertielle est confondu avec le repère monde. Ceci est obtenu en redéfinissant le repère monde comme étant le repère global. Certes l'approche est simple mais elle ne peut être appliquée que sur des environnements à petite échelle où il est possible de placer la centrale inertielle sur le repère monde et de redéfinir le repère global.

Il nous faut une méthode de calibration qui soit en adéquation avec l'environnement extérieur. De plus, elle doit estimer la relation qui permet de déduire l'orientation de la caméra à partir de l'orientation de la centrale inertielle. Nous allons à présent proposer une solution de ce type.

4.3 Capteur Inertiel/Caméra : Modélisation et calibration

Le but du couplage Inertiel/Caméra est d'avoir une estimation permanente de l'orientation du point de vue. Cette dernière représente la rotation du repère caméra C par rapport au repère monde W.

Nous avons fixé la centrale inertielle sur la caméra de manière à obtenir une transformation rigide entre les deux repères. L'orientation est obtenue des données de la caméra en se basant sur le modèle sténopé décrit dans l'annexe A. Voyons les données fournies par la centrale inertielle.

4.3.1 Centrale inertielle

La centrale inertielle utilisée dans notre système est une centrale de type MTi fabriquée par Xsens [Technologies, 2007]. La MTi est une unité de mesures inertielles miniature avec des magnétomètres 3D intégrés (compas 3D) ainsi qu'un processeur embarqué à faible puissance. Ce dernier fonctionne avec un algorithme de fusion de données qui permet de calculer l'orientation absolue dans l'espace 3D à partir des gyroscopes, accéléromètres et magnétomètres. Brièvement, l'algorithme de fusion consiste à utiliser la mesure de gravité fournie par les accéléromètres et le nord magnétique donné par les magnétomètres afin de compenser l'augmentation des erreurs, c'est-à-dire la dérive, causée par l'intégration des vitesses angulaires pour calculer les orientations. Ce type de compensation est connu sous le nom attitude et cap de référence utilisé dans des systèmes AHRS (Attitude and Headings Reference System).

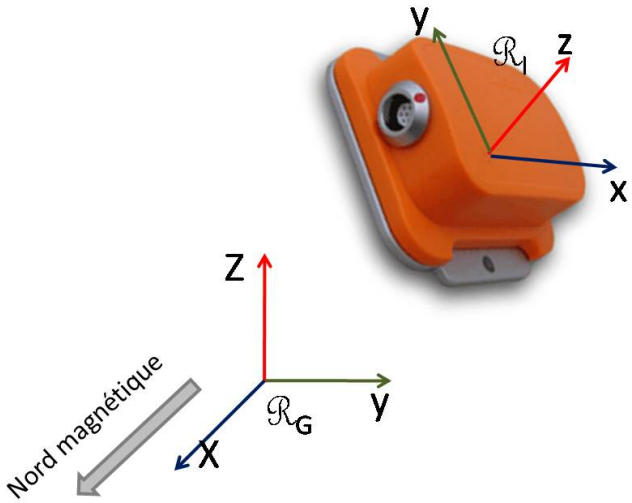


FIG. 4.9: Centrale Inertielle

Les accélérations, les vitesses angulaires et le champ magnétique sont définis dans le repère cartésien direct qui est attaché au boitier du capteur (cf. fig.4.9). Les orientations 3D sont calculées entre le repère lié au boitier qu'on notera I et un repère lié à la terre noté G. Par défaut le repère local fixé à la terre est défini comme un repère cartésien direct tel que :

- l'axe X pointe vers le nord magnétique local ;
- l'axe Y selon le repère main droite ;
- l'axe Z pointe vers le haut.

Le repère global utilisé par la centrale inertielle MTi est défini selon le nord magnétique local qui est différent du nord géographique. La déviation entre les deux varie selon la localisation sur la terre : c'est la déclinaison magnétique qui peut être grossièrement obtenue à partir de divers modèles de champ magnétique terrestre en fonction de la latitude et la longitude. Selon les besoins, le repère global peut être redéfini à l'aide d'une fonction d'initialisation fournie avec la centrale inertielle.

Les orientations peuvent être fournies sous forme d'un quaternion, d'angles d'Euler ou d'une matrice d'orientation. De plus, les mesures sont horodatées ce qui facilite la synchro-

nisation des données entre les différents capteurs utilisés.

4.3.2 Modélisation

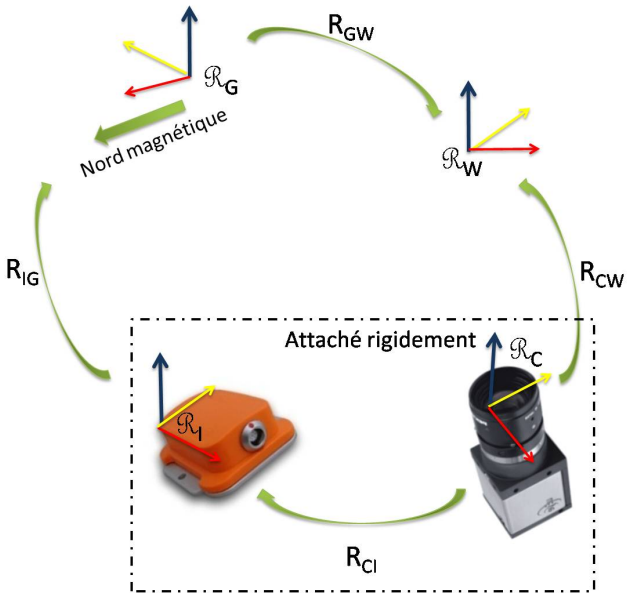


FIG. 4.10: Configuration des repères

Nous avons d'une part la centrale inertielle qui fournit des données concernant son repère local $\mathscr{R}_I$ par rapport à son repère global $\mathscr{R}_G$. D'autre part, la vision fournit les données relatives à son repère $\mathscr{R}_C$ par rapport à son repère monde $\mathscr{R}_W$. Ces différents repères sont illustrés à la figure 4.10. Etant donné que le couplage Inertiel/Caméra est utilisé pour estimer l'orientation, les transformations prises en compte sont les rotations entre les différents repères. De plus, étant donné que les deux capteurs sont rigidement liés, lorsqu'un capteur effectue un mouvement rotationnel, l'autre capteur effectue le même mouvement.

Dans ce qui va suivre, nous représentons les rotations sous forme matricielle puisque les angles d'Euler souffrent de perte de degré de liberté (gimbal lock) et l'utilisation des quaternions donne une formule non linéaire. De ce fait, à partir de la configuration des repères présentés dans la figure 4.10, nous définissons les transformations suivantes :

- R_{CW} la matrice de rotation du repère associé au monde par rapport au repère de la caméra ;
- R_{IG} la rotation entre le repère local de la centrale inertielle $\mathscr{R}_I$ et le repère global de la centrale inertielle ;
- R_{CI} la rotation qui relie le repère de la caméra $\mathscr{R}_C$ et le repère local de la centrale inertielle $\mathscr{R}_I$;
- R_{GW} la rotation entre le repère monde $\mathscr{R}_W$ et le repère global de la centrale inertielle $\mathscr{R}_G$.

L'objectif de notre sous-système est de pouvoir déduire, à tout instant, R_{CW} à partir de R_{IG}. Pour cela, la relation entre les deux capteurs est déterminée par une transformation f telle que :

$$R_{CW} = f(R_{IG}) \tag{4.18}$$

À partir de la configuration de nos repères, la relation entre les différents repères est donnée par l'équation suivante :

$$R_{CW} = R_{CI} R_{IG} R_{GW} \tag{4.19}$$

De ce fait, nous constatons que la relation entre les deux orientations (i.e. entre les deux capteurs) est définie par les matrices de rotations R_{CI} et R_{GW}. Donc, la phase de calibration consiste à déterminer ces deux rotations.

4.3.3 Calibration

Nous nous retrouvons avec la même configuration de système de coordonnées que le système proposé par Reitmayr et Drummond [Reitmayr et Drummond, 2006] (ou plutôt dans [Baillot et al., 2003]) ainsi que le système de [Bleser, 2009]. Comme nous l'avons vu précédemment, la calibration proposée repose sur le calcul du mouvement relatif à partir de points prédéfinis. L'approche proposée par [Bleser, 2009] se base sur le calcul des vecteurs de gravité. Nous voulons une procédure de calibration qui soit peu contraignante, requérant des hypothèses raisonnables.

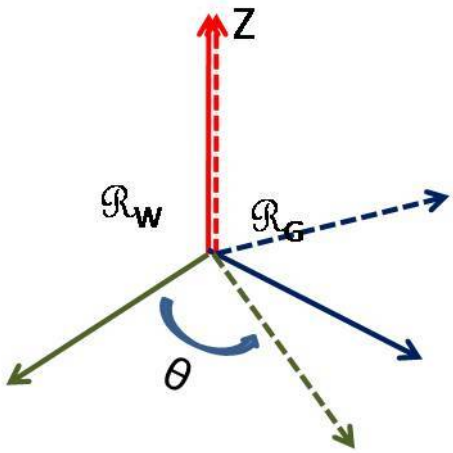


FIG. 4.11: Relation entre le repère R_G et le repère R_W

Nous savons que l'axe des Z du repère global de la centrale inertielle $\mathscr{R}_G$ est vertical et dirigé vers le haut. Nous allons exploiter cette information pour ainsi simplifier le calcul pour estimer une rotation et déduire la seconde. Pour cela, nous imposons que l'axe des Z du repère monde soit colinéaire avec cet axe. Ceci constituera notre hypothèse de départ. Les deux axes sont parallèles. La rotation R_{GW} s'exprime comme une rotation d'un angle θ autour de l'axe Z (cf. fig.4.13). R_{GW} s'exprime alors sous la forme :

$$R_{GW} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4.20}$$

Posons :

$$R_{CW} = (r_{ij}^{CW}), R_{CI} = (r_{ij}^{CI}), R_{IG} = (r_{ij}^{IG}) \tag{4.21}$$

Notre hypothèse, permet de réécrire l'équation (cf. eq.4.19), ce qui nous donne :

$$R_{CW} = \begin{pmatrix} R_{CI}r_1^{IG} & R_{CI}r_2^{IG} & R_{CI}r_3^{IG} \end{pmatrix} \begin{pmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4.22}$$

r_i^{IG} étant la $i$ème colonne de la matrice R_{IG} et $c = \cos(\theta)$, $s = \sin(\theta)$. Après calcul, nous obtenons :

$$R_{CW} = \begin{pmatrix} R_{CI}r_1^{IG} \begin{pmatrix} c \\ s \end{pmatrix} & R_{CI}r_2^{IG} \begin{pmatrix} -s \\ c \end{pmatrix} & R_{CI}r_3^{IG} \end{pmatrix} \tag{4.23}$$

A partir de l'équation (4.23), nous pouvons déduire que :

$$r_3^{CW} = R_{CI}r_3^{IG} \tag{4.24}$$

où r_3^{CW} est la troisième colonne de R_{CW}. En réécrivant l'équation (4.24) sous la forme linéaire ($Ax = b$), nous estimons R_{CI} à partir de trois ensembles de données en utilisant les moindres carrées. Une fois R_{CI} est estimée, R_{GW} est déduite à partir de l'équation (4.19) :

$$R_{GW} = R_{IG}^T R_{CI}^T R_{CW} \tag{4.25}$$

Au final, notre méthode de calibration consiste en l'estimation de la matrice de rotation R_{CI} et la déduction de la seconde matrice de rotation R_{GW}. Connaissant ces deux rotations, nous pouvons exprimer l'orientation donnée par la centrale inertielle dans le repère associé à la caméra.

4.4 Capteur GPS/Caméra : modélisation et calibration

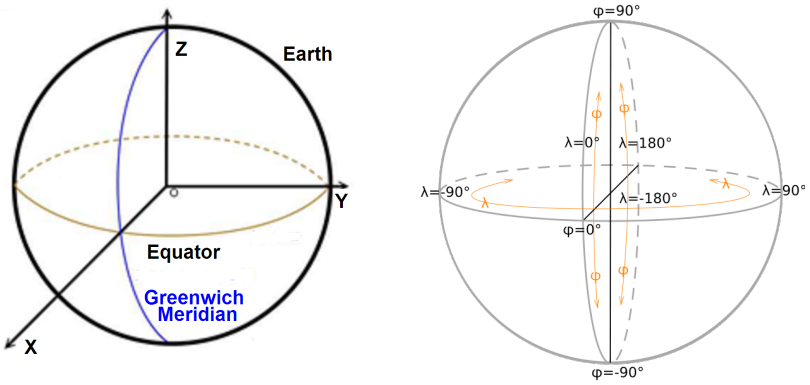
Le couplage GPS/Caméra vient en complément du couplage Inertiel/Caméra qui permet d'estimer en permanence l'orientation. En effet, le GPS est combiné avec la caméra pour fournir en continu la position de l'utilisateur dans l'environnement. Le GPS est le capteur le plus adéquat en environnement extérieur. Son utilisation permet de pallier le problème rencontré lors de l'utilisation des accéléromètres pour calculer la position. Avant de présenter la procédure de calibration, voyons les données fournies par ce capteur.

4.4.1 GPS

Dans notre système de localisation, nous utilisons un récepteur GPS ProXT de la firme Trimble. Les données fournies par ce récepteur sont transmises en utilisant la norme NMEA (National Marine Electronics Association). La norme NMEA, plus précisément NMEA-0183, est un protocole de communication qui définit comment les données sont transmises entre les instruments et équipements électroniques liés au GPS.

Sans rentrer dans les détails, les données sont transmises sous forme de paquets toutes les secondes. Chaque paquet de données comprend plusieurs types de trames. Ces trames englobent essentiellement l'identifiant du récepteur (par exemple GP pour " Global Positioning System"), le temps d'acquisition, les coordonnées géographiques (longitude, latitude et altitude), le nombre de satellites visibles, un indicateur sur la précision de positionnement appelé DOP (Dilution Of Position), l'élévation et l'azimut * du premier satellite, etc.

*L'azimut est l'angle horizontal entre la direction d'un objet et une direction de référence. Wikipédia

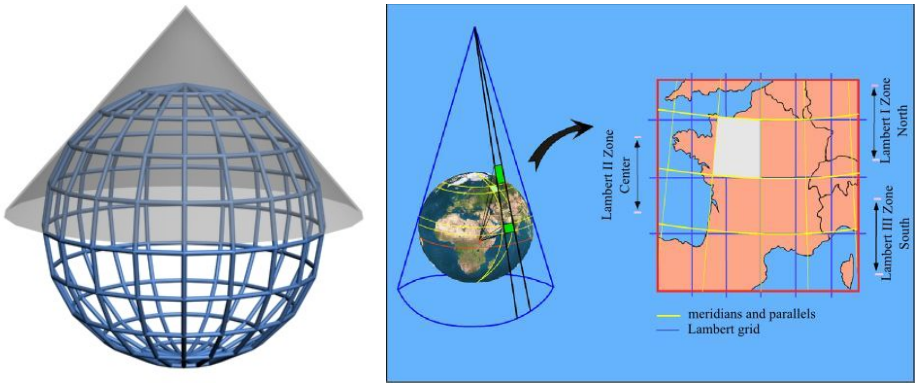


(a) (b)

FIG. 4.12: (a) repère défini par le système WGS84 (b) Coordonnées géographiques : longitude et latitude

La donnée qui nous intéresse est la position sous forme de coordonnées géographiques. Elles permettent de se localiser sur la surface de la terre en latitude et en longitude (cf. fig.4.12-a) et sont déterminées par un système géodésique. Il permet de définir une ellipsoïde de référence pour caractériser la terre en déterminant la valeur de son demi grand axe et le rapport grand axe sur petit axe. Il existe plusieurs systèmes géodésiques qui sont en général en adéquation avec l'endroit où ils sont utilisés. Le GPS utilise le système WGS84 qui constitue une bonne approximation à l'échelle globale. La figure 4.12-b illustre le système de coordonnées défini par le système WGS84.

Les longitudes et latitudes étant exprimées en degrés, il nous est nécessaire de les réexprimer et de les convertir dans un repère métrique de référence. Ceci consiste à faire le passage de coordonnées géographique à des coordonnées cartographiques. Etant donné que la surface de la terre n'est pas uniforme, il existe plusieurs repères cartographiques. En ce qui nous concerne, puisque nous sommes en France, nous utiliserons le repère cartographique basé sur une projection conique conforme de Lambert. Celui-ci est usuellement utilisé pour établir les cartes concernant le territoire français.



(a) (b)

FIG. 4.13: (a) Projection conique (b) Projection conique conforme Lambert : cas de la France

La projection conique est une projection cartographique qui conserve les distances. Elle se base sur un cône superposé à la sphère (la terre dans notre cas) (cf. fig.4.13-a). Le sommet du cône appartient à l'axe des pôles et est tangent à un ellipsoïde de référence en un point défini par un méridien de référence et un parallèle de référence de latitude ϕ_0 qui est aussi l'angle au sommet du cône. Le cône est ensuite développé sur un plan, sans glissement. Le repère associé $\mathscr{R}_L$ à la projection conique est défini tel que l'axe Y est colinéaire au méridien et l'axe X est la tangente à la sphère appartenant au cône. La figure 4.13-b illustre cette projection appliquée à la France.

4.4.2 Modélisation

La position fournie par le GPS doit représenter la position de la caméra par rapport au repère associé à la scène réelle $\mathscr{R}_W$. Lorsque le repère monde est un repère local et est différent du repère utilisé par le GPS, il nous faut estimer la transformation rigide (rotation et translation) qui relie les deux repères $\mathscr{R}_W$ et $\mathscr{R}_L$. Elle permet de déduire la position de la caméra à partir de la position du GPS.

Pour cela, à chaque position GPS, notée p_{gps}, nous associons une position caméra, notée p_{cam}, et obtenue par estimation de pose sachant que $p_{cam} = -R_{CW}^T t_{CW}$. La relation entre ces deux positions est donnée par :

$$p_{cam} = R_{WL} p_{gps} + t_{WL} \tag{4.26}$$

4.4.3 Calibration

A partir de différentes mesures, cette transformation, représentée par (R_{WL}, t_{WL}) est obtenue en minimisant le critère suivant :

$$\sum_i^n \| p_{cam}^i - R_{WL} p_{gps}^i + t_{WL} \|^2 \tag{4.27}$$

Pour obtenir la transformation optimale, nous définissons des vecteurs normalisés $\overrightarrow{N}^i_{gps}$ et $\overrightarrow{N}^i_{cam}$ associés respectivement à p_{gps}^i et p_{cam}^i et définis comme suit :

$$\overrightarrow{N}^i_{gps} = \frac{p_{gps}^j - p_{gps}^i}{\| p_{gps}^j - p_{gps}^i \|} \tag{4.28}$$

$$\overrightarrow{N}^i_{cam} = \frac{p_{cam}^j - p_{cam}^i}{\| p_{cam}^j - p_{cam}^i \|} \tag{4.29}$$

Avec $i = 1..\frac{n}{2}$ et $j = \frac{n}{2} + 1..n$. La relation reliant ces vecteurs normalisés est une rotation telle que :

$$\sum_i^n \| \overrightarrow{N}^i_{cam} - R_{WL} \overrightarrow{N}^i_{gps} \|^2 \tag{4.30}$$

La figure 4.14 représente la relation existante entre les positions et les vecteurs normalisés obtenus. À partir de l'équation (4.30), nous pouvons estimer la rotation R_{WL}. Nous nous basons sur l'approche décrite par **Radu Horaud et Olivier Monga** dans le chapitre 7 de leur livre *Vision par ordinateur* [Horaud et Monga, 1995].

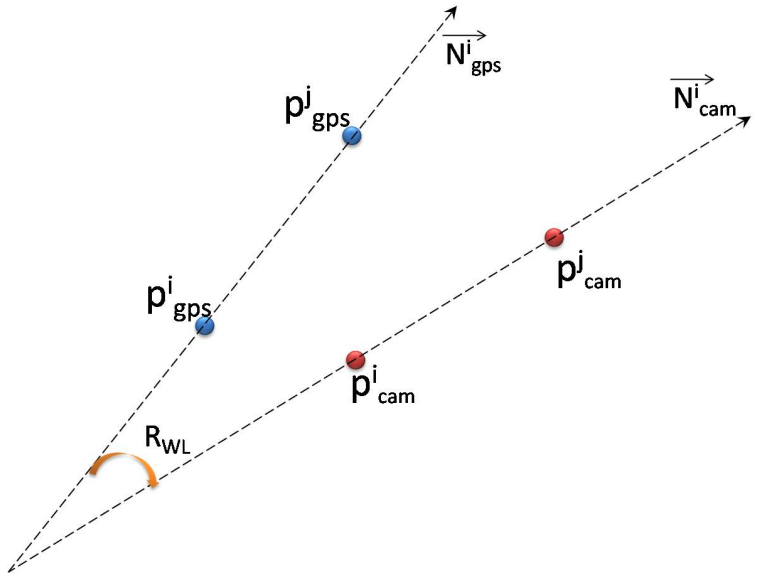


FIG. 4.14: Calcul de l'orientation entre les positions GPS et postion caméra

En utilisant la représentation d'axe et d'angle de rotation $(\overrightarrow{n}, \theta)$, nous avons les propriétés géométriques suivantes :

$$\overrightarrow{n}.(\overrightarrow{N}^i_{gps} - \overrightarrow{N}^i_{cam}) = 0 \tag{4.31}$$

$$(\overrightarrow{N}^i_{gps} + \overrightarrow{N}^i_{cam}).(\overrightarrow{N}^i_{gps} - \overrightarrow{N}^i_{cam}) = 0 \tag{4.32}$$

Nous en déduisons la relation suivante :

$$[\overrightarrow{N}^i_{gps} + \overrightarrow{N}^i_{cam}]_x \overrightarrow{\Omega} = \overrightarrow{N}^i_{gps} - \overrightarrow{N}^i_{cam} \tag{4.33}$$

Avec $\overrightarrow{\Omega} = \tan(\theta/2).\overrightarrow{n}$ et $[.]_x$ est la matrice antisymétrique équivalente au produit vectoriel. En résolvant l'équation (4.33), nous estimons $\overrightarrow{n} = \frac{\overrightarrow{\Omega}}{\|\overrightarrow{\Omega}\|}$ et $\theta = 2\arctan(\|\overrightarrow{\Omega}\|)$.

Une fois la rotation optimale estimée, la translation est déduite à partir de :

$$t_{WL} = \overline{p}_{cam} - R\overline{p}_{gps} \tag{4.34}$$

Où $\overline{p}_{cam} = \frac{1}{n}\sum_i^n p^i_{cam}$ et $\overline{p}_{gps} = \frac{1}{n}\sum_i^n p^i_{gps}$.

Le GPS est utilisé pour une localisation 2D (longitude et latitude). En ce qui concerne la hauteur, celle-ci est déterminée par la taille moyenne de l'utilisateur ainsi que l'élévation par rapport à l'origine. L'élévation peut être fournie par le modèle numérique de terrain (MNT) en connaissant la position 2D dans l'environnement.

4.5 Expérimentations et résultats

Après avoir décrit les capteurs utilisés, ainsi que les procédures de calibration mises au point, nous allons nous intéresser, d'une part, à la fiabilité et la précision des données fournies par ces capteurs, d'autre part, nous allons quantifier la précision des procédures de calibration que nous avons proposées. Ceci englobe la précision de l'estimation des

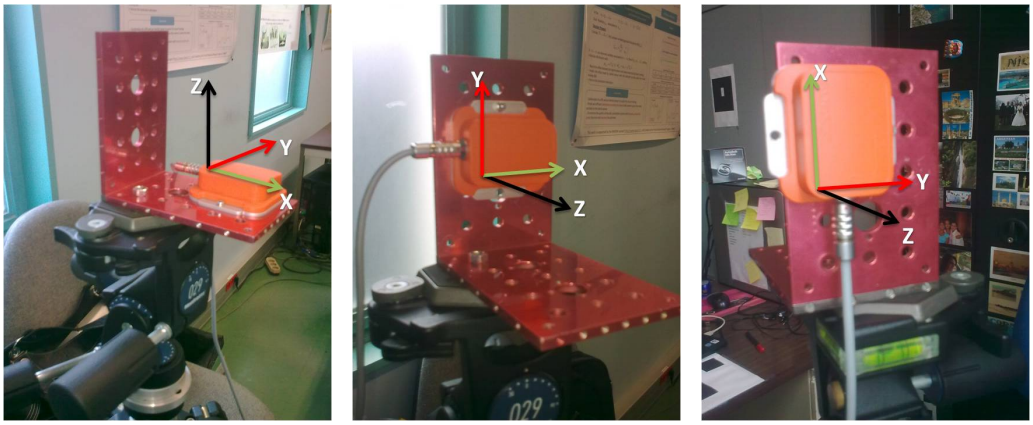
transformations ainsi que la précision des données déduites en utilisant ces transformations. Plusieurs expérimentations on été conduites dans ce sens. Voici une description de ces protocoles ainsi que les résultats obtenus dans chaque cas.

4.5.1 Caractérisation de la centrale inertielle

Les premières séries d'expérimentations concernent la centrale inertielle. Elles sont menées dans le but de caractériser le capteur que nous utilisons dans notre système. Cela nous permettra de quantifier la précision des données fournies par la centrale inertielle.

Pour cela, nous fixons notre centrale inertielle sur un trépied doté de trois axes de rotation. Un axe permet de réaliser des rotations de 360° autour de l'axe verticale. Les deux autres permettent d'avoir des rotations entre $-30°$ et 90°. La centrale inertielle est fixée selon trois configurations différentes tel que :

1. Configuration 1 : axe Z à la verticale pointant vers le haut (cf. fig.4.15-a)
2. Configuration 2 : axe Y à la verticale pointant vers le haut (cf. fig.4.15-b)
3. Configuration 3 : axe X à la verticale pointant vers le haut (cf. fig.4.15-c)



(a) Configuration 1 (b) Configuration 2 (c) Configuration 3

FIG. 4.15: Les différentes configurations pour la précision des orientations de la centrale inertielle

Pour chaque configuration, nous effectuons plusieurs orientations dans chaque axe. Ces orientations varient d'une amplitude de 15°. Pour chaque rotation, nous récupérons les données fournies par la centrale inertielle à savoir : les accélérations linéaires, les vitesses angulaires, les champs magnétiques et les orientations. Les orientations fournies par la centrale inertielle sont récupérées sous forme de quaternions. Cependant, nous les mettons sous forme matricielle (cf. annexe A). Pour calculer la différence entre deux rotations, nous optons pour le calcul du mouvement relatif entre elles. Puis, l'angle de différence entre deux rotations est calculé comme suit :

$$\theta_i = \arccos\left(\frac{trace(R_{ref}^T R_i) - 1}{2}\right) \tag{4.35}$$

telle que R_{ref} est la matrice de rotation de référence acquise au début de l'expérimentation et R_i la rotation à l'instant i.

Nous comparons l'angle de rotation obtenu avec des angles de rotation de référence. Ceux-ci correspondent aux angles fournis par le trépied. La figure 4.16 compare ces deux orientation. Le tracé rouge représente les angles de rotation entre les orientations de la centrale inertielle et l'orientation initiale. En bleu, ce sont les rotations de références.

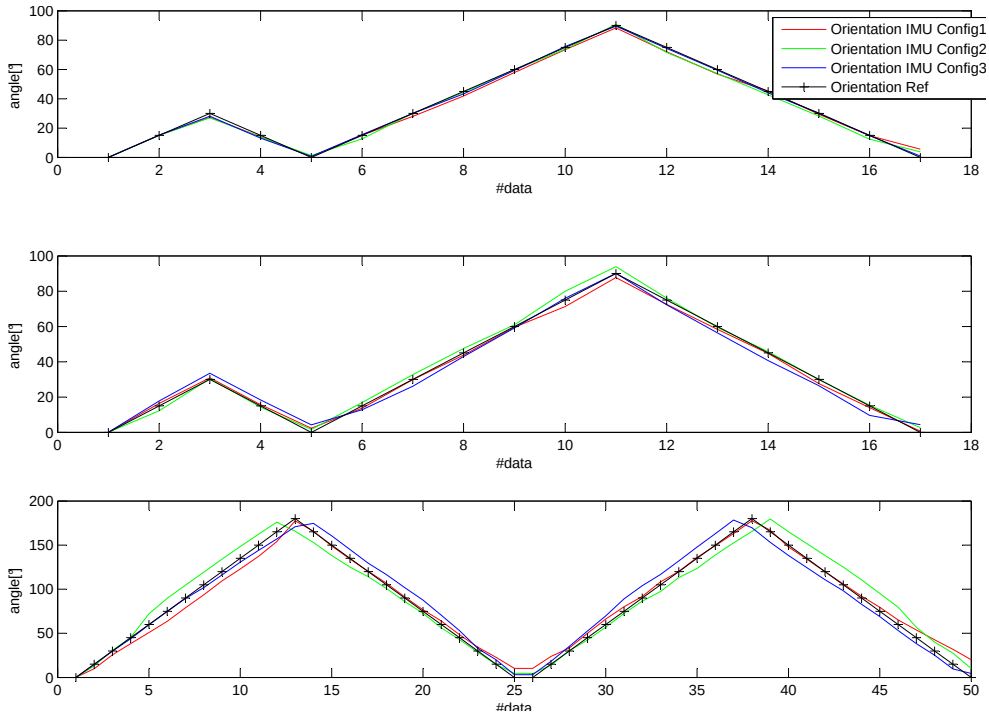


FIG. 4.16: Orientation centrale inertielle (rouge, vert et bleu) vs. orientation de référence (noir)

Du point de vue numérique, le tableau 4.2 présente les résultats obtenus en donnant la moyenne et les écarts types des différences observées.

	configuration 1		configuration 2		configuration 3	
	moyenne	écart type	moyenne	écart type	moyenne	écart type
Axe X	1.6366	1.5060	1.6111	1.2551	7.2911	4.2412
Axe Y	1.3493	1.0256	9.1402	6.2405	2.8156	1.5760
Axe Z	5.4312	4.8442	1.6374	1.4119	0.7286	0.5897

TAB. 4.2: Performances des orientations fournies par la centrale inertielle

Nous avons aussi calculé les rotations relatives entre deux orientations successives. Ceci nous a donné des rotations avec un angle minimal égale à 13.87° et un écart type de l'ordre de 3.12° et un angle maximal de 14.78° avec un écart type de 1.49°. Sachant que la rotation relative de référence est de 15°, ceci correspond à une erreur entre 1.13° et 0.22°. Ceci peut s'expliquer, d'une part, par une influence sur le champ magnétique qui existe dans l'environnement où les expérimentations ont été effectuées et, d'autre part, par l'imprécision qui peut être induite par les graduations du trépied. Pour information, ces expérimentations ont été effectuées au départ avec une tourelle. Cependant, les moteurs de cette dernière influencent énormément les mesures fournies par la centrale inertielle.

D'après ce que nous avons obtenu, la centrale inertielle donne des résultats assez précis. Entre les 3 configurations, la première configuration avec le l'axe des Z à la verticale donne de meilleurs résultats essentiellement en rotation autour de Z qui donne une erreur de l'ordre de 5.4312°. La précision que nous obtenons représente la précision des orientations relatives. La précision absolue ne peut pas être quantifiée directement dans notre cas puisque nous utilisons le repère global défini par défaut et que nous n'avons pas de données de références. Selon les constructeurs, la centrale inertielle présente une erreur inférieure à 0.5 °.

4.5.2 Calibration Inertiel/Caméra : évaluation de la précision de l'estimation des rotations

La seconde série d'expérimentations impliquant la centrale inertielle concerne la procédure de calibration Inertiel/Caméra. Elle a pour objectif de caractériser les performances de la procédure de calibration que nous proposons. Dans ce protocole, nous nous intéressons à la précision des données déduites à partir des orientations de la centrale inertielle et des transformations obtenues à la suite de la calibration. Ces données sont comparées aux orientations obtenues avec la vision. Dans ces expérimentations, nous allons nous baser sur une comparaison entre les orientations fournies par la centrale inertielle et la vision. Pour quantifier ces performances, nous évoluons dans notre environnement en effectuant plusieurs orientations. Nous calculons d'une part les orientations fournies par la vision et celles déduite à partir des données de la centrale inertielle et des rotations obtenues lors de la phase de calibration réalisée au préalable. En comparant les deux orientations, nous calculons la différence d'angle en utilisant l'équation 4.35. Ceci nous donne une erreur moyenne égale à 7.7897° avec +/- 4.4985°.

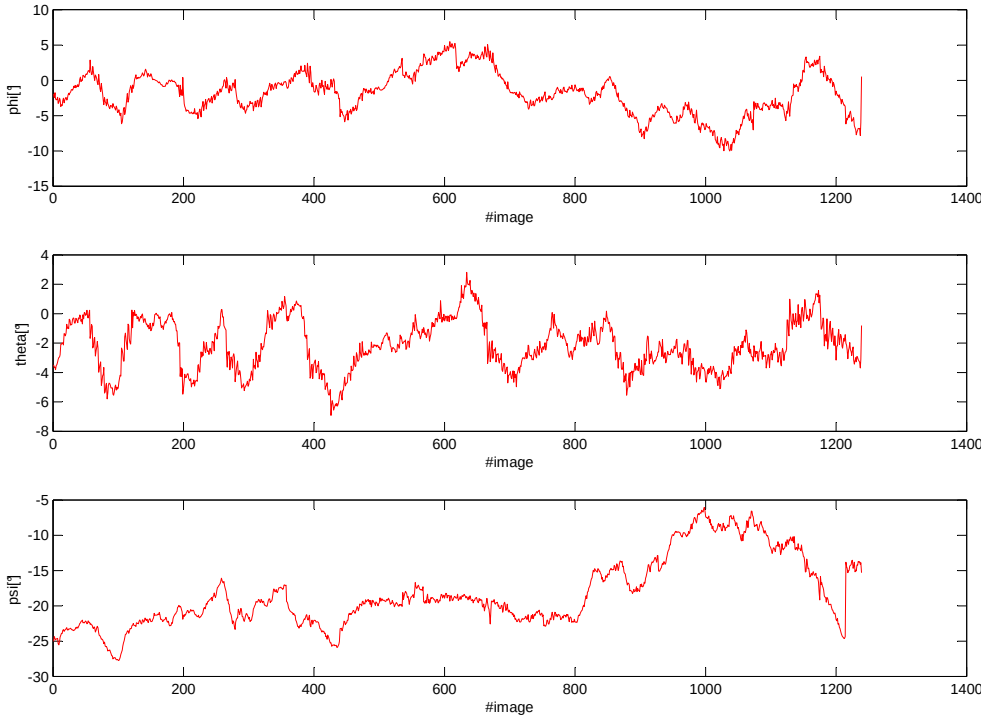


FIG. 4.17: Variation du R_{GW} : en angle d'euler

Nous constatons que l'erreur entre les deux orientations est assez élevés avec une erreur moyenne qui de l'ordre de 7.7°. Cette erreur peut être due à plusieurs facteurs. D'un côté ceci est du aux erreurs qui peuvent être introduites par la vision. D'autre part, celle-ci est aussi causée par l'influence de conditions extérieures sur le repère global dont l'axe X est défini selon le nord magnétique. Nous nous sommes intéressés aux variations de la rotation entre le repère globale de la centrale inertielle et le repère monde. Pour cela, nous recalculons cette rotation pour chaque vue. La figure 4.17 présente les rotations obtenues exprimées sous la forme d'angles d'Euler ce qui permet de constater les variations de cette rotation. Ceci représente une variation de l'ordre de $(2.3924°, 1.7301°, 6.7769°)$. Nous constatons que cette variation est plus marquée sur l'axe Z ce qui est tout à fait normal étant donné que les deux repères sont définis par une rotation d'un angle θ autour de l'axe Z.

Dans la figure 4.18 présente un tracé des rotations sous forme d'angles d'Euler pour visualiser la variation.

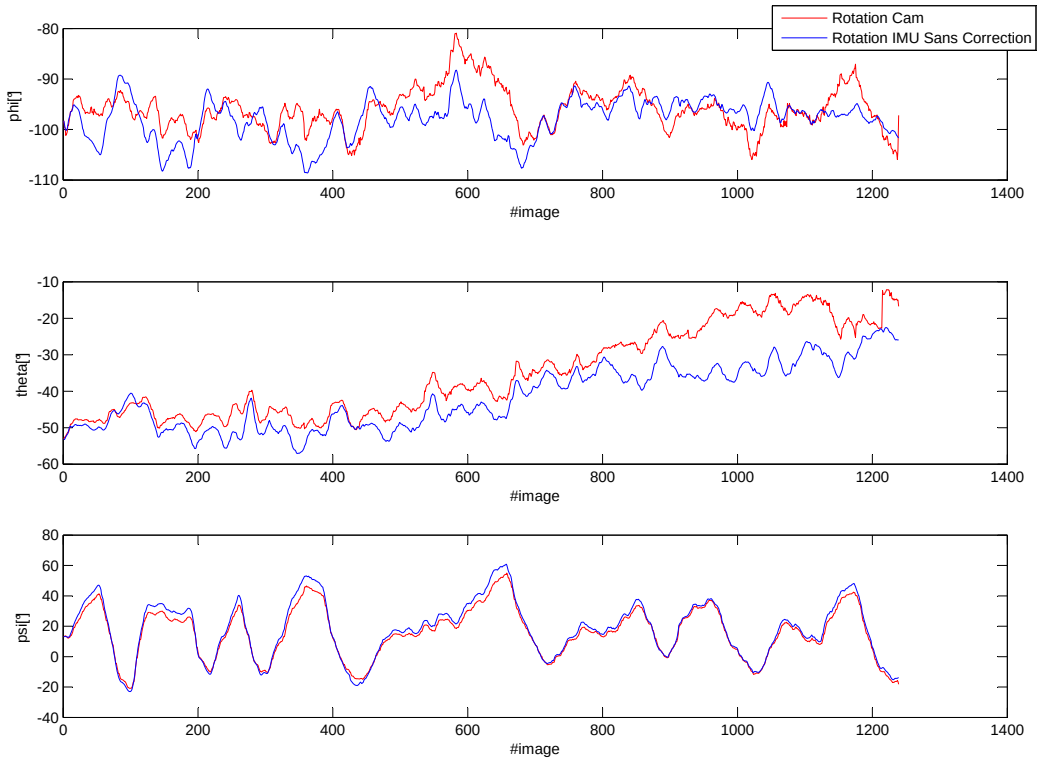


FIG. 4.18: orientation inertielle (en angle d'euler) (bleu) vs orientation caméra (rouge)

Ceci nous amène, pour pallier cette variation, à recalculer en ligne le R_{GW} à chaque nouvelle image et d'utiliser cette rotation pour l'image suivante. Avec cette correction, les erreurs obtenues précédemment sont réduites. En effet, nous obtenons une erreur moyenne égale à 0.9745 ° avec un écart type de l'ordre de 0.7054 °. Nous pouvons visualiser dans la figure 4.19 cette amélioration en observant les angles déduits des rotations de la centrale inertielle corrigée (bleu) en comparaison à celles de la vision (en rouge).

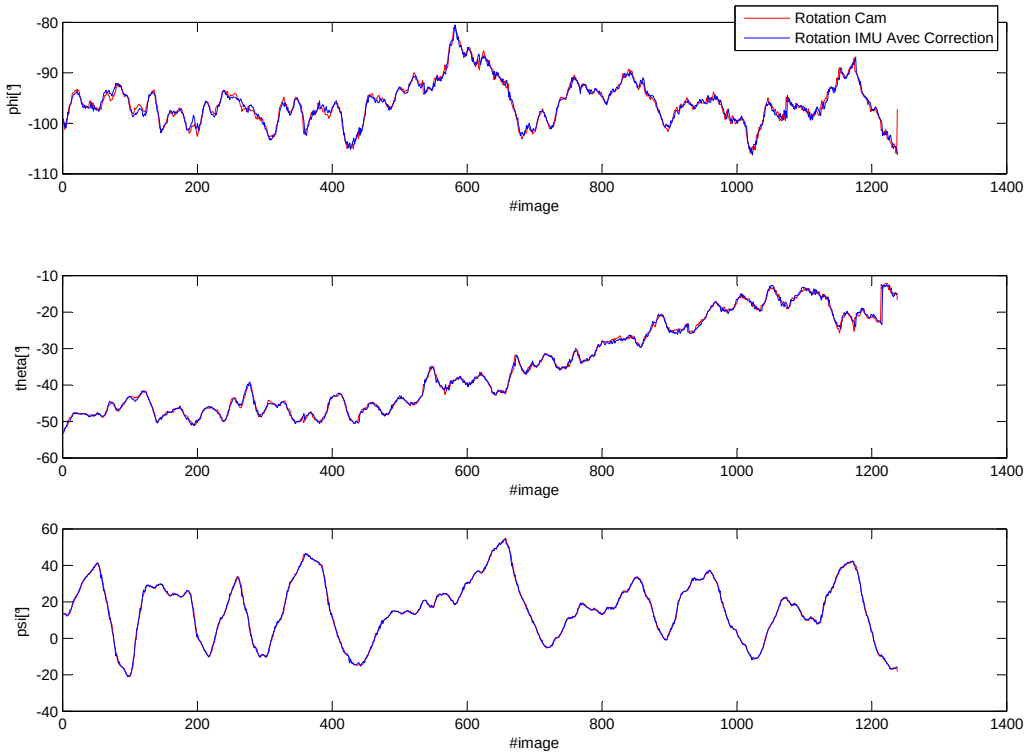


FIG. 4.19: orientation inertielle avec correction (en angle d'Euler) (bleu) vs orientation caméra (rouge)

Voyons ce que ceci donne point de vue recalage. Nous avons d'un côté le résultat de recalage avec les orientations sans correction et d'un autre côté le recalage obtenu avec les orientations de la centrale inertielle corrigée. La figure 4.20 présente quelques images obtenues où nous pouvons observer la différence entre les deux résultats de recalage avec l'amélioration obtenue avec la correction du R_{GW}.

4.5.3 Caractérisation du récepteur GPS

La deuxième partie de notre système est constituée du couplage GPS/Caméra en vue d'estimer en permanence la position du point de vue. Le GPS est un des capteurs destiné uniquement aux environnements extérieurs. Dans la partie qui suit, nous allons mener deux séries d'expérimentations. Les premières expérimentations caractérisent le GPS seul. Nous cherchons à connaitre la précision réelle du capteur. Ces tests se basent d'une part sur les données brutes et d'autre part sur les données obtenues après conversion en coordonnées cartographiques.

Nous choisissons d'évoluer sur un chemin en ligne droite. Il servira de référentiel pour mesurer les performances du récepteur. La droite est échantillonnée en plusieurs points. En chaque point, nous récupérons les coordonnées géographiques qui sont transformées par la suite dans le repère cartographique. Il faut signaler que seule la longitude et la latitude sont prises en compte. En effet, l'altitude n'est pas très précise. La hauteur sera fixée à la taille moyenne de l'utilisateur ou bien à la hauteur retournée par la vision additionnée

(a) image0030 : Sans Correction

(b) image0030 : Avec Correction

(c) image0165 : Sans Correction

(d) image0165 : Avec Correction

(e) image0720 : Sans Correction

(f) image0720 : Avec Correction

(g) image1250 : sans Correction

(h) image1250 : Avec Correction

FIG. 4.20: Comparaison entre le recalage obtenu avec les orientations de la centrale inertielle avec et sans correction

à l'élévation obtenue par le MNT. La droite que nous utilisons comme référence est une ligne qui existe sur le sol, elle est échantillonnée en plusieurs points équidistants entre eux. La distance entre deux points successifs est de l'ordre de 1 mètre. A chaque position, nous récupérons les positions fournies par le récepteur GPS.

Dans un premier temps, nous allons nous intéresser aux coordonnées géographiques (i.e. longitude et latitude). La figure 4.21 présente un tracé des coordonnées géographiques (données en degrés). Nous utilisons le fait que la trajectoire sur laquelle nous évoluons soit une droite pour quantifier l'erreur. Pour cela, nous calculons l'équation de la droite en utilisant une régression linéaire à partir de l'ensemble des données. Chaque point de la ligne doit appartenir à cette droite. L'erreur que nous obtenons est très faible et est en moyenne de l'ordre de 0.010 secondes d'arc avec un écart type de l'ordre de 0.011 secondes d'arc. Cependant, ce sont les coordonnées cartographiques qui nous intéressent. Nous calculons ces coordonnées en utilisant la projection Lambert 2. La figure 4.21 présente les positions obtenues dans un espace métrique. Comme pour les coordonnées géographiques, nous recalculons l'équation de la droite à partir des coordonnées métriques. Nous obtenons une erreur moyenne de l'ordre de $0.321m$ avec un écart type égale à $0.334m$. Cependant, nous retrouvons à certaines positions des erreurs supérieures au mètre et une erreur maximale de l'ordre de $1.693m$.

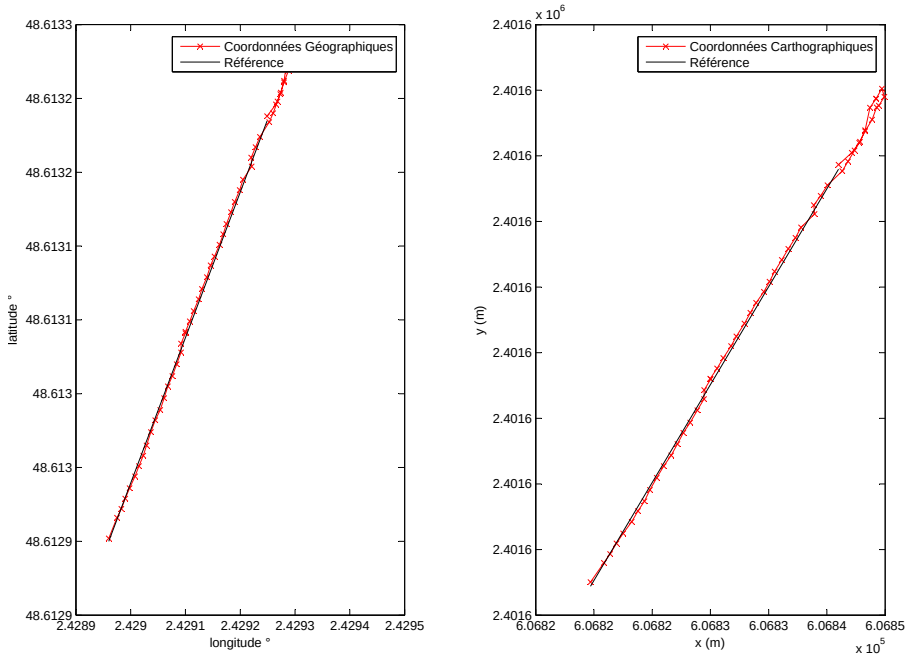


FIG. 4.21: Positions exprimées en coordonnées géographique (à gauche) et cartographique (à droite)

Ensuite, nous calculons les distances entre deux positions successives. En moyenne, nous obtenons une distance de l'ordre de 1.062 avec un écart type $0.213m$ sachant que la distance entre deux points de la ligne est égale à 1 mètre. Les variations enregistrées dans cette expérimentation sont dues d'une part à la distorsion induite par la projection et d'autre part à l'incertitude induite lors des mesures. Il reste à voir l'erreur qui est introduite par la calibration.

4.5.4 Calibration GPS/Caméra : évaluation de la précision de l'estimation de la position

La seconde partie des expérimentations concerne la précision induite par la calibration. Nous ne pouvons pas connaitre la transformation réelle entre les deux repères. Toutefois, les expérimentations conduites nous permettent de savoir si la transformation optimale calculée avec la procédure de calibration altère ou non la précision de la position déduite à partir de celle fournie par le récepteur GPS et de la transformation calculée. Pour cela, nous nous intéressons aux positions globales par rapport au repère monde (repère local défini dans l'environnement). Nous avons déployé deux protocoles différents : le protocole de "**la ligne droite**" et celui des positions aléatoires. Ces deux protocoles ont été déjà utilisés pour quantifier les performances de la vision en termes de localisation (cf. section 2.6.2.2 page 58).

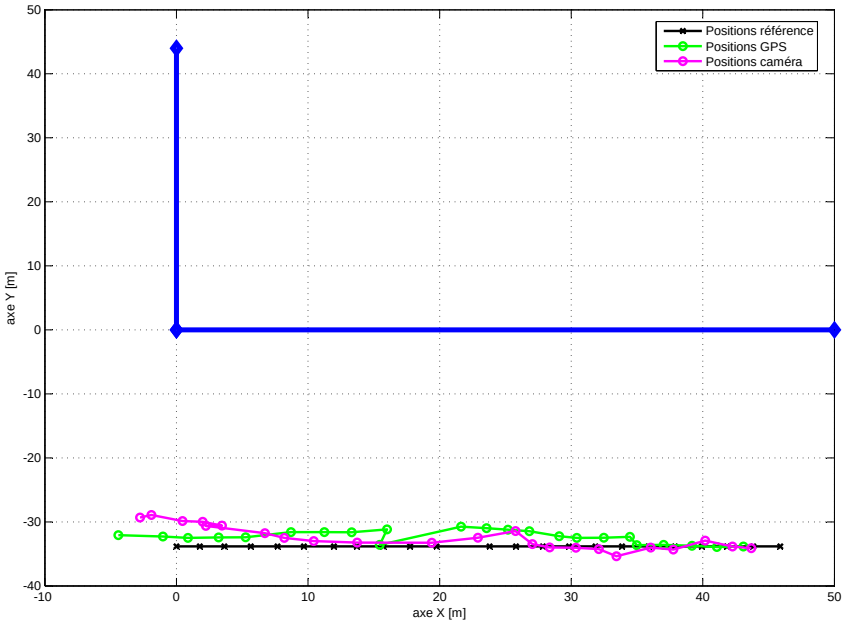


FIG. 4.22: Tracé de la trajectoire du GPS (vert) vs. caméra (magenta) vs. réelle (noir) sur une ligne droite

Le protocole de "**la ligne droite**" consiste à évoluer le long d'une ligne droite utilisée comme référence dont nous connaissons les coordonnées par rapport au repère local. Cette ligne droite est échantillonnée en plusieurs points. A chaque point de cette ligne, nous récupérons les positions GPS transformées dans le repère local ainsi que les positions obtenues par la vision (cf. section 2.3 page 48). Puis, nous allons comparer les positions GPS, d'une part, avec les positions réelles et, d'autre part, avec les positions fournies par la vision. Dans le figure 4.22 nous traçons des trajectoires obtenues à partir des positions GPS illustrées en vert, comparées à la trajectoire de la caméra tracée en magenta et à la trajectoire réelle tracée en noir.

Les résultats obtenus sont donnés dans le tableau 4.3. Nous retrouvons dans ce tableau l'erreur moyenne sur l'axe X (μ_x) et l'axe Y (μ_y) avec les écarts types associés (respectivement σ_x et σ_y). Nous présentons dans ce tableau les erreurs obtenues entre les données GPS et les les données de référence (GPS/Ref) ainsi qu'avec les données issues de la vision (GPS/Camera).

	μ_x	μ_y	σ_y	σ_y
GPS/Ref	1.8374m	1.4810m	1.1094m	0.9597m
GPS/Camera	1.7321m	1.4702m	1.8314m	1.0116m

TAB. 4.3: Temps de calcul par phase pour le calcul de pose

Nous constatons que sur l'axe X, l'erreur obtenue est légèrement plus élevée que sur l'axe Y. Si nous nous intéressons aux erreurs maximales, nous obtenons un maximum de 4.4112m (0.2318m au minimum) sur l'axe X et un maximum de 3.0807m sur l'axe Y (0.0268m au minimum). Cette différence peut être incombé à la précision des poses basée vision utilisées dans le processus de calibration. Comparons nos résultats avec ceux déjà présenté dans la littérature. Par exemple dans [Reitmayr et Drummond, 2007], les auteurs présentent une erreur moyenne égale approximativement à $(4m, 10m)$ avec un écart type de l'ordre de $(1.9m, 10m)$. Ces résultats sont obtenus à partir des données cartographique alors que les notre sont obtenus à partir des données transformées dans le repère locale. Le fait de transformer ces positions dans le repère local associé à l'environnement permet de réduire l'erreur de positionnement du GPS. Cependant, cette précision est fortement liée d'une part à la précision de la calibration et précisément des données utilisée dans la calibration (positions caméra et positon GPS), de plus elle dépend des données à transformer.

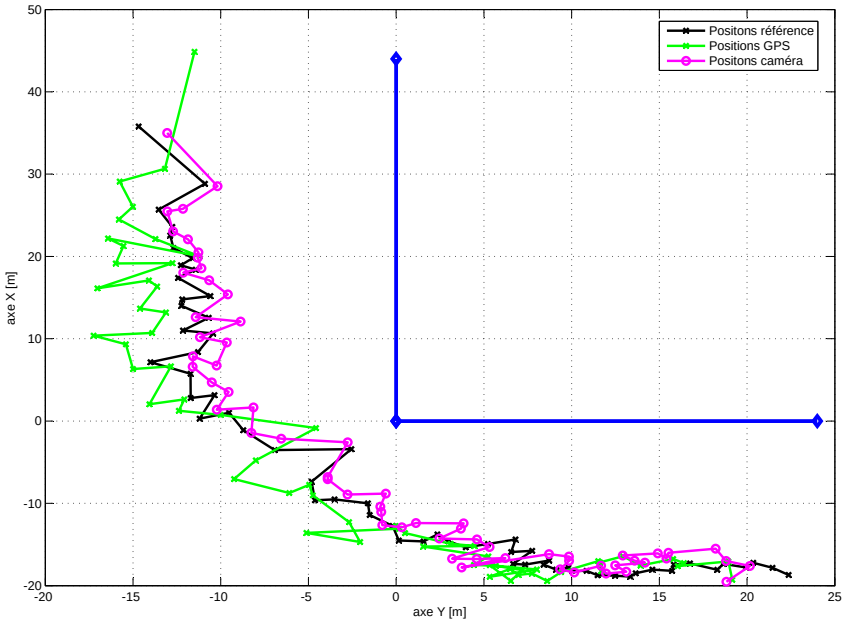


FIG. 4.23: Positions GPS (vert) vs. caméra (magenta) vs. réelle (noir)

Pour consolider les résultats obtenus lors du premier protocole, nous avons réalisé pour le second protocole des mesures sur des positions choisies aléatoirement dans l'environne-

ment où nous évoluons. A chaque position de référence, nous allons comparer, d'une part, les positions fournies par le récepteur GPS et la vision avec, d'autre part, les données de référence. Nous pouvons visualiser à la figure 4.23 les trajectoires obtenues à l'aide des différentes mesures (GPS en vert, caméra en magenta et positions de référence en noir). Visuellement, les positions GPS sont plus éloignées que les positions de vision par rapport aux données de référence. Pour mieux interpréter les résultats obtenus, nous calculons les erreurs moyennes sur chaque axe et nous les reportons sur le tableau qui suit :

	μ_x	μ_y	σ_y	μ_y
GPS/Camera	2.434m	0.773m	1.5488m	0.6023m
GPS/Ref	3.4474m	0.6399m	1.2283m	0.5131m

TAB. 4.4: Temps de calcul par phase pour le calcul de pose

A l'instar des résultats obtenus précédemment, l'erreur sur l'axe X est toujours aussi élevée que celle obtenue sur l'axe Y. Dans ce protocole, les erreurs sur l'axe Y sont en moyenne inférieures au mètre. Cependant cette erreur diffère selon où nous nous trouvons dans l'environnement. En effet, sur la partie à droite du bâtiment (cf. fig.4.23), l'erreur moyenne est égale à (2.88m,2.47m). Ceci est du essentiellement à la faible couverture GPS dans cette zone en raison de la proximité du bâtiment et la présence d'arbres. Cependant, ces résultats restent corrects.

Du point de vue du recalage, nous pouvons observer dans la figure 4.24 quelques résultats obtenus à différentes positions et avec différents point de vue. Ce recalage est réalisé à partir des positions fournies par le GPS et des orientations de la vision. Pour mieux visualiser les performances du recalage, nous projetons le modèle filaire avec les poses fournies par la vision (en rouge) et celles obtenues avec le GPS (en vert). Les deux modèles ne se superposent pas. Cependant, nous constatons qu'avec les données du GPS nous obtenons un recalage dans le voisinage de celui réalisé avec la vision. En termes de précision numérique, l'erreur que nous obtenons est égale à 64.7935 pixels avec un écart type égale à 38.7278 pixels. Certes c'est une grande erreur, mais elle est obtenue à partir de la position GPS. Donc, en pensant à un moyen pour corriger la position GPS pour le rendre plus proche de la vision, nous pourrions réduire cette erreur de recalage et ainsi obtenir un meilleur résultat. Nous y reviendrons plus tard sur cette correction (cf. section 5.2.2 page 123).

4.6 Conclusion

La calibration est une phase importante pour les systèmes multi-capteurs. Celle-ci permet de déterminer les transformations entre les différents capteurs selon un modèle définissant la relation entre eux.

Dans ce chapitre, nous avons présenté quelques approches proposées dans la littérature pour la calibration du couplage caméra et centrale inertielle. Ces approches se basent sur des hypothèses simplificatrices ce rend ces méthodes applicables que dans certains scénarios. Ceci nous a amené à proposer une approche en adéquation avec notre système.

L'approche de calibration que nous proposons exploite une des caractéristiques autour de la définition des repères utilisés par la centrale inertielle. Sachant que la relation entre les deux capteurs exploite deux rotations intermédiaires. Cette l'hypothèse permet de réduire

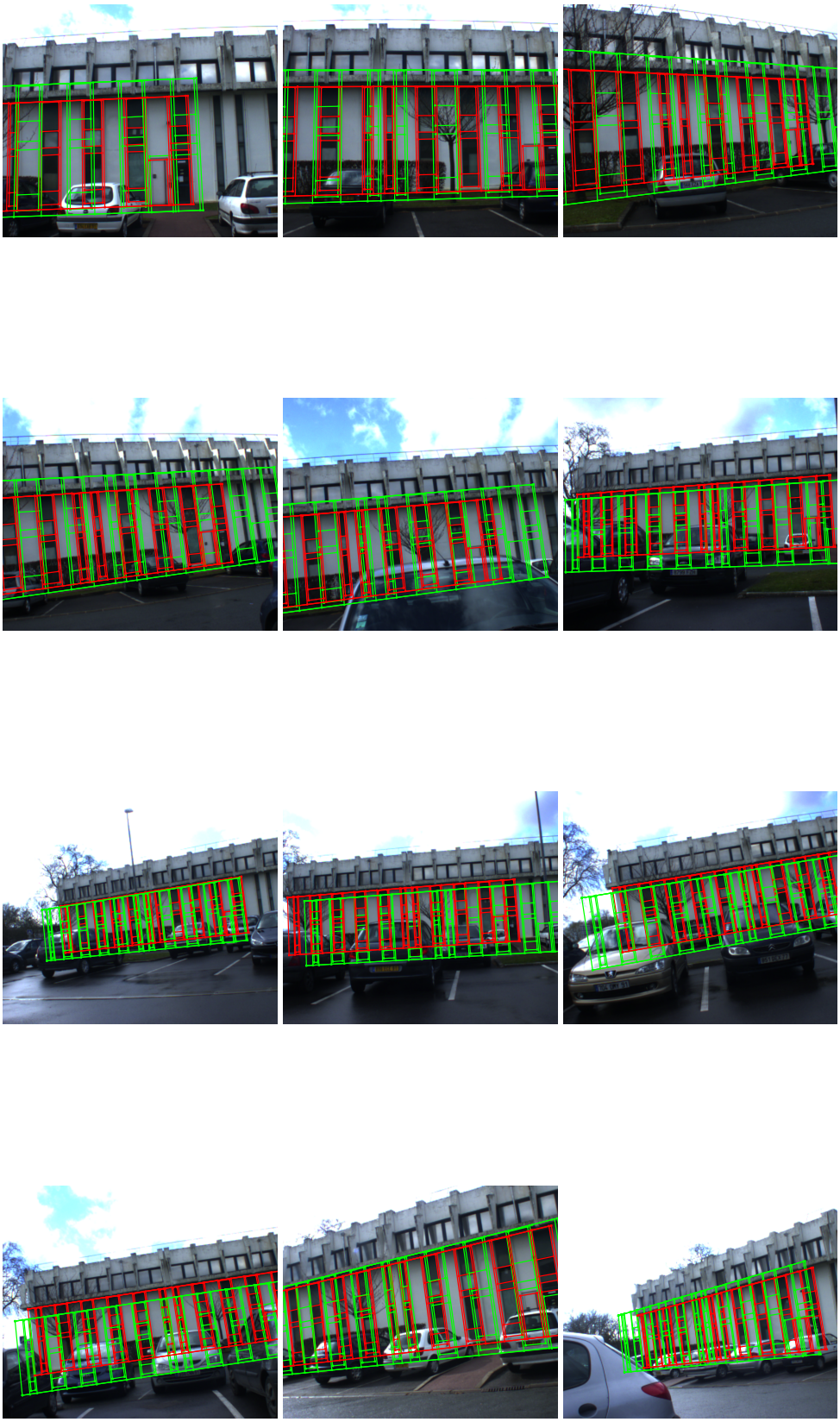


FIG. 4.24: Résultat de recalage avec les positions GPS (vert) vs. la vision (rouge).

la phase de calibration au calcule d'une rotation et la déduction de la seconde.

En plus de l'approche de calibration Inertiel/Caméra, nous avons proposé une approche pour calibrer le capteur hybride composé du récepteur GPS et de la caméra. Ceci est motivé par le fait que les positions fournies par le GPS sont définies par rapport au repère terre, alors que les positions estimées par la vision sont déterminées par rapport à un repère local. L'approche que nous proposons estime une transformation rigide entre les deux repères en se basant sur une représentation axe et angle.

Les approches que nous proposons sont simples à mettre en œuvre et ne requièrent pas de lourdes hypothèses. De plus, elles ont l'avantage de fonctionner dans différents systèmes. En effet, en ce qui concerne par exemple l'approche de calibration Inertiel/Caméra, peut être utilisée aussi bien dans le cas où nous voulons déduire les orientations absolues que relatives.

Suivant les expérimentations conduites pour caractériser les deux approches de calibration, les résultats que nous avons obtenus sont décalés par rapport à ce que donne la vision. Cependant, ceci nous permet d'entrevoir une procédure de correction afin de rapprocher les estimations fournies par la paire GPS/Inertiel à celles fournies par la vision.

Dans le chapitre qui suit, nous allons voir comment la correction est réalisée et essentiellement quelles sont les données utilisées pour la correction. De plus, nous allons présenter en détail notre système de localisation multi-capteurs ainsi que son fonctionnement.

Chapitre 5

Localisation basée suppléance multi-capteurs

Dans le chapitre 2, nous avons introduit notre sous-système de vision. Nous avons présenté une approche de localisation utilisant des points d'intérêts extraits de la scène. Cette méthode contient une approche d'initialisation semi-automatique qui permet de retrouver l'ensemble des appariements 2D/3D. Ces derniers sont maintenus avec un suivi 2D/2D afin de calculer la pose au fil des images.

Nous avons donné un descriptif global de notre système dans le chapitre 3 (page 85) ainsi que les problématiques à traiter (cf. sec.3.4 page 86). Par la suite, dans le chapitre 4, nous avons présenté les procédures de calibration du capteur hybride composé d'une caméra, une centrale inertielle et un GPS. Cette étape de préparation réalisée hors ligne permet d'unifier les données fournies par les différents capteurs et de les exprimer dans le même référentiel.

Jusqu'à présent nous n'avons pas encore abordé le fonctionnement propre de notre système et la manière dont les différentes parties interagissent. Ce chapitre présente le rôle exact du sous-système d'assistance composé de la centrale inertielle et du GPS et détaille les améliorations qu'il apporte au sous-système de vision décrit auparavant. Nous nous intéresserons aux traitements ajoutés dans les différents sous-systèmes afin d'assurer l'interaction entre eux et améliorer la qualité des estimations. Dans la quatrième partie du chapitre, nous allons aborder les premières expérimentations conduites pour tester le système de localisation dans sa globalité. Ayant déjà présenté quelques performances des différentes parties du système (à savoir le sous-système de vision et la calibration), Ces tests permettent d'analyser le comportement du système face à différentes situation. De plus, quelques résultats concernant les traitements complémentaires ajoutés feront l'objet d'une brève étude.

Avant de conclure ce chapitre, nous allons nous intéresser à un cas concret d'application du système que nous proposons. Il s'agit du projet RAXENV. Nous allons présenter un bref descriptif de ce projet, les solutions adoptées ainsi que quelques résultats obtenus.

5.1 Description du système

Notre système est composé d'une caméra combinée avec un récepteur GPS couplé à une centrale inertielle. La localisation fonctionne selon un schéma de suppléance ou d'assistance. Ceci revient à subdiviser le système de localisation en deux parties : un sous-système principal et un sous-système d'assistance.

Dans notre cas, le sous-système principal s'appuie sur la vision. En effet, en utilisant la méthode sans marqueurs décrite dans le chapitre 2, la pose de la caméra est estimée à partir du flux d'images. L'utilisation de la caméra seule peut être suffisante sauf dans le cas où celle-ci est dans l'incapacité d'estimer la pose à partir des données dont elle dispose. Dans ce cas, le sous-système d'assistance qui comprend le récepteur GPS et la centrale inertielle a pour rôle de remplacer le sous-système principal pour fournir une estimation de la localisation jusqu'à ce que ce dernier soit capable de reprendre la main.

Pour pouvoir basculer d'un sous-système à autre, le système de localisation doit disposer de critères pour détecter la défaillance du sous-système de vision. Ceci nous amène à incorporer au niveau du sous-système de vision un test pour vérifier la cohérence de la pose estimée. Le résultat de ce test permettra au système, soit d'exploiter la vision ou donner la main au sous-système d'assistance.

Dans le cas de défaillance de la vision, le sous-système d'assistance estime la pose à partir des données fournies par le récepteur GPS et la centrale inertielle. Les données fournies par ces deux capteurs sont, bien évidemment, transformées dans le repère de référence choisi en utilisant les paramètres obtenus lors de la phase de calibration.

Par ailleurs, le sous-système d'assistance à la localisation (AL) ne se contente pas de remplacer le sous-système de vision mais les poses qu'il fournie sont aussi utilisées pour réaliser le rendu du modèle filaire de la scène. De plus, à partir de la position et de l'orientation fournie par le sous-système d'assistance, nous pouvons définir un critère qui permet de valider la pose estimée par le sous-système de vision.

Toutefois, les données fournies par le sous-système d'assistance ne sont pas aussi précises que l'estimation fournie par la vision. De ce fait, le système de localisation doit être capable de corriger les poses du sous système d'assistance pour approximer les poses fournies par la vision. Ainsi, le sous-système AL incorpore un module de prédiction/correction. Il permet de prédire les erreurs produites par le système de localisation en se basant sur un apprentissage en ligne de l'erreur obtenue entre les mesures fournies par les deux sous-systèmes. Le module AL utilise cette erreur pour corriger et raffiner la position et l'orientation fournies par le GPS et la centrale inertielle.

De plus, lorsque le système de localisation a détecté la défaillance du sous-système de vision et a basculé vers le sous-système d'assistance, il doit être capable de reprendre le suivi visuel lorsque ce dernier redevient fonctionnel. Ceci nécessite la mise au point d'une approche de réinitialisation qui retrouve les appariements 2D/3D pour re-estimer la localisation à partir de la vision.

La figure 5.1 illustre le flux de données dans notre système de localisation. Ce schéma reprend les différents modules décrits et illustre les échanges entre eux.

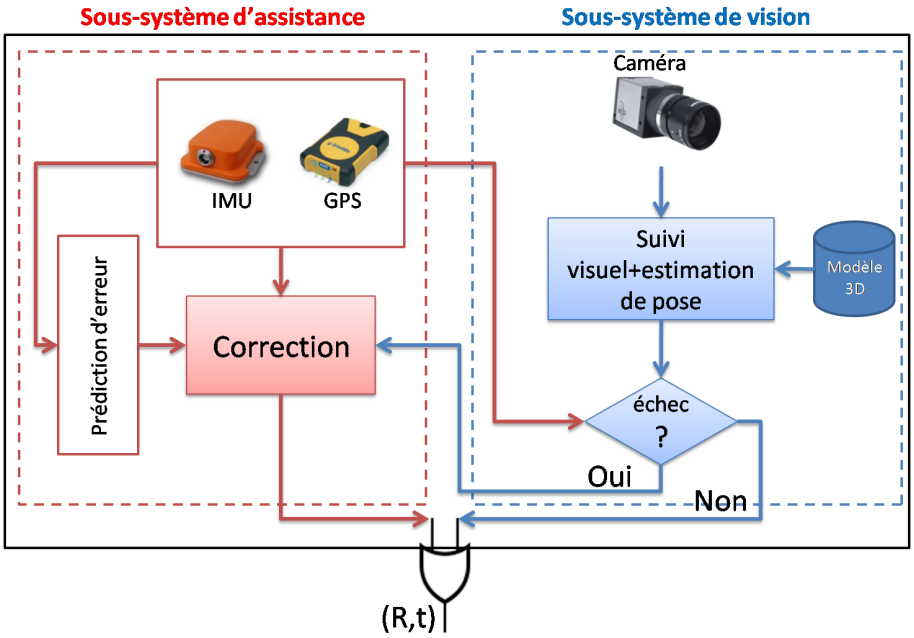


FIG. 5.1: flux de données

Avant de décrire le fonctionnement propre du système, nous allons tout d'abord compléter la description des modules additionnels à savoir les tests de validation et la procédure de prédiction et de correction.

5.2 Composants nécessaires à l'intégration des deux sous-systèmes

Dans ce qui va suivre, nous allons présenter en détail les traitements ajoutés pour compléter le descriptif de notre système de localisation donné auparavant. Nous allons décrire essentiellement le mécanisme d'interaction entre les deux sous-systèmes.

5.2.1 Critères de validation

Une des fonctionnalités qu'offre notre système de localisation est la gestion des défaillances induites par le sous-système de vision. Pour cela, nous avons développé un système de décision qui se base sur un certain nombre de critères afin d'indiquer si le système de localisation accorde sa confiance aux mesures fournies par le sous-système de vision ou plutôt fait appel au sous-système d'assistance à la localisation pour le remplacer. L'idée est de définir des indicateurs permettant de déterminer si le sous-système de vision a échoué ou non, c'est-à-dire si la pose estimée est correcte.

Ces indicateurs sont définis en connaissant les causes qui provoquent l'échec des méthodes basées vision. Celles-ci sont imputables à plusieurs facteurs essentiellement les occultations, les mouvements brusques et/ou le changement de luminosité. Ces conditions influent sur le suivi visuel ce qui perturbe l'ensemble des appariements 2D/3D et ainsi fausse l'estimation de pose. De ce fait, les indicateurs que nous avons sélectionnés sont :

- Le nombre de points suivis ;
- L'erreur de reprojection ;
- Les intervalles de confiance.

Si un de ces critères n'est pas vérifié, la pose calculée est rejetée. Dans ce cas, le système de localisation bascule vers le sous-système d'assistance.

5.2.1.1 Le nombre de points suivis

Le nombre de points constituant l'ensemble des appariements 2D/3D utilisés pour le calcul de la pose influence la précision de l'estimation de pose. En effet, plus nous avons de bons appariements 2D/3D plus la pose estimée est précise. Il suffit donc de définir un nombre minimum de points acceptés pour calculer la pose à partir de l'équation (2.4 page 49). En théorie, il suffit de disposer de 4 points au minimum pour résoudre cette équation. Etant donné que nous sommes dans un environnement large, il nous faut un nombre de points minimum plus élevé qui varie entre 10 à 20 points.

5.2.1.2 L'erreur de reprojection

Le nombre d'appariements utilisés pour le calcul de la pose n'est pas un critère suffisant pour détecter la défaillance du sous-système de vision. Pour la méthode utilisant les points, nous nous basons aussi sur l'erreur de reprojection. Cette erreur de reprojection représente la moyenne quadratique de la différence entre les points 2D extraits des images et la projection des points 3D avec la pose estimée. Sachant que (m_i, M_i), $i = 1..n$, sont des couples d'appariements obtenus à partir du suivi, et (R_{CW}, t_{CW}) est la pose estimée à partir de cet ensemble d'appariements, l'erreur de re-projection se présente comme suit :

$$\varepsilon = \frac{1}{n}\sum_{i=1}^{n} \|(m_i - K * (R_{CW}M_i + t_{CW}))\| \tag{5.1}$$

Avec K la matrice des paramètres intrinsèques. Si cette erreur est grande, i.e. supérieure à un seuil fixé, la pose est jugée erronée. Dans nos tests, nous avons choisi un seuil de l'ordre de 100 $pixels^2$

5.2.1.3 Intervalles de confiance

Jusqu'à maintenant, nous n'avons vu que des indicateurs issus de la vision. Cependant, parallèlement au sous-système de vision, nous disposons d'un autre sous-système qui fournit aussi une information de localisation. Bien évidemment, nous parlons du sous-système d'assistance à la localisation. Les mesures fournies par ce module d'assistance étant moins précises que celles fournies par la vision, ces données peuvent malgré tout servir d'indicateur de validation de la cohérence des mesures obtenue par la vision. En effet, ces données peuvent être utilisées pour définir des intervalles de confiance. Ils permettent de juger de la validité de la pose obtenue par le sous-système de vision en les confrontant avec celles issues du sous-système AL.

Concrètement, à partir de la position fournie par le GPS et transformée en utilisant les paramètres obtenus avec la procédure de calibration GPS/Caméra décrit dans la section 4.4 (page 103), nous définissons une ellipse d'erreur. Elle est définie par son centre qui n'est autre que la position 2D obtenue et par ses axes dont la largeur est en fonction de l'écart type de l'erreur entre la position obtenue avec la vision et celle déduite du GPS. Plus précisément, ils sont égaux à trois fois cet écart type ($3 * \sigma$). Pour valider la position déduite de la vision, il suffit juste de vérifier si cette position appartient à l'intervalle de confiance calculé à partir de la position courante du GPS.

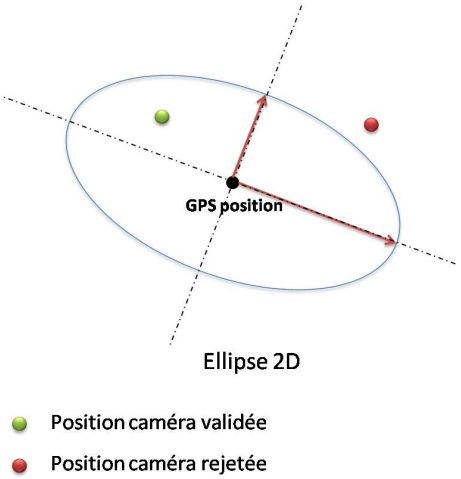


FIG. 5.2: Intervalle de confiance défini pour valider les positions estimées par la caméra.

Pour l'orientation, la validation est réalisée autrement. En effet, chaque orientation fournie par la vision est comparée à l'orientation déduite des données fournies par la centrale inertielle. Le système estime la différence entre les deux rotations qui est représentée par ΔR tel que :

$$\Delta R = R_{CW}^{T} f(R_{GI}) \tag{5.2}$$

En calculant la trace de la différence, nous pouvons déduire l'angle θ, l'angle de rotation entre les deux orientations tel que :

$$\theta = \arccos \frac{Trace(\Delta R) - 1}{2} \tag{5.3}$$

Nous partons du principe que si les deux rotations sont identiques, le résultat de ΔR doit être égale à la matrice identité telle que ceci nous donne une trace égale à trois et de ce fait un angle $\theta = 0$. De ce fait, le test de validité consiste à estimer la trace de la matrice de différence entre les deux rotations. Si cette trace est inférieure à un certain seuil, l'orientation est jugée valide sinon l'estimation est rejetée. Dans notre système, le seuil choisi est égale à 2.9 ce qui correspond à 18°.

5.2.2 Prédiction et correction d'erreur

A partir des résultats présentés dans le chapitre 4, les poses fournies par le sous-système AL sont moins précises que celles obtenues par le sous-système de vision. Ceci est du à plusieurs facteurs qui sont essentiellement la précision des capteurs utilisés dans le système d'assistance et de la précision de la procédure de calibration. De ce fait, l'estimation de l'erreur produite par le sous-système AL est importante dans le processus de localisation. En effet, identifier cette erreur permet de quantifier la qualité des mesures et ainsi améliorer l'estimation de la localisation. L'erreur de localisation est obtenue par différenciation entre la pose estimée par la vision et la pose fournie par le sous-système AL. En connaissant cette différence, la pose fournie par l'assistance peut être corrigée. Cependant, le problème se pose lorsque le sous-système d'assistance prend le relais. Dans ce cas, cette erreur ne peut pas être calculée directement étant donné que les données de la vision ne sont pas disponibles. De ce fait, nous avons besoin de prédire cette erreur pour corriger approximativement ce que fournit à cet instant le sous-système AL.

Pour cela, nous nous orientons vers des outils de prédiction. Par définition, la prédiction consiste à déterminer une mesure dans l'avenir à partir des mesures observées dans le passé. De plus, il faut connaître les lois qui déterminent cet avenir en fonction de son passé. Les différents outils de prédiction qui sont proposés se basent en générale sur le concept de la régression linéaire. Elle regroupe toute méthode modélisant la relation entre une donnée Y et une ou plusieurs variables telle que le modèle dépend linéairement de paramètres inconnues à estimer à partir d'un ensemble de données d'observation. La régression linéaire correspond au modèle dont la moyenne conditionnelle de Y donnée par X est une fonction affine de X. La régression linéaire se base sur une distribution de probabilité conditionnelle de Y donné par X plutôt que sur la distribution de probabilité conjointe de X et Y. Dans le but de prédire, la régression linéaire est utilisée pour ajuster un modèle prédictif à partir d'un ensemble de données d'observation de valeur Y et X sachant que $Y = f(X)$. A partir du modèle construit, si une nouvelle valeur de X est donnée sans connaitre sa valeur Y, le modèle ajusté permet de fournir une prédiction de la valeur Y.

Suivant le principe de régression, les réseaux de neurones peuvent être utilisés pour prédire les données. Par définition, un réseau de neurones artificiels est un modèle mathématique qui tente de simuler les structures et les fonctionnalités des aspects des réseaux de neurones biologiques. Il représente un ensemble de neurones artificiels et de processus d'information interconnectés. Les réseaux de neurones ont la particularité d'être des systèmes adaptatifs selon les informations (internes ou externes) qui circulent dans le réseau lors de la phase d'apprentissage. En ce qui concerne le neurone artificiel, celui-ci représente une fonction mathématique conçue telle un modèle brut qui a une ou plusieurs entrées et les combine pour fournir une sortie appelée synapse. Les sorties de chaque neurone sont par la suite passées vers une fonction non-linéaire appelée fonction de transfert ou d'activation. Pareil que la régression linéaire, le réseau de neurone cherche à ajuster les coefficients associés à chaque donnée en entrée appelée poids synaptique afin de minimiser une fonction d'erreur. Le processus itératif qui permet de les définir représente la phase d'apprentissage.

Les filtres particulaires sont des techniques d'estimation de modèle basée sur la simulation et sont utilisés généralement pour l'estimation des modèles Bayésiens. Le filtre particulaire estime la séquence des paramètres x_k, $k = 0,1,2,...$ en se basant sur des données observées y_k pour $k = 0,1,2,....$ Toutes les estimations baysiennes de x_k suivent une distribution *a posteriori* $P(x_k|y_0,y_1,,y_k)$. De même, le filtre de Kalman utilise des mesures observées bruitées et fournit des valeurs qui convergent vers les valeurs réelles des mesures et des valeurs calculées qui leur sont associées en prédisant la valeur, estimant l'incertitude et calculant les poids moyens des valeurs prédites et mesurées. Le filtre de Kalman est un filtre récursif qui estime l'état actuel à partir de l'état précédent et des mesures courantes. De ce fait, l'estimation n'a pas besoin d'historique ni des observations ni des estimations. La prédiction définie dans le filtre de Kalman utilise l'estimation calculée précédemment pour estimer l'état courant. Cette prédiction est connue comme état *a priori*.

Nous voulons utiliser un modèle de prédiction qui permet de réaliser un apprentissage en ligne de l'erreur produite et utilisent cet apprentissage pour prédire l'erreur qui est produite à cet instant pour ainsi corriger en ligne les mesures fournies. Les prédicteurs présentés précédemment sont écartés pour plusieurs raisons. En effet, certains n'offrent pas une possibilité d'effectuer l'apprentissage en ligne telle que les réseaux de neurones. De plus, ce type de prédicteur ne fournit de bon résultat que lorsque les données en entrées sont définies dans le voisinage des données utilisées pour l'apprentissage. Ceci exclut le fait de réaliser un apprentissage hors-ligne. Les autres types de prédicteurs soit n'ont pas besoin

d'un historique de l'évolution de l'erreur telle que le filtre de Kalman ou bien peuvent présenter un problème de temps de calcul.

Dans nos travaux, nous allons présenter un autre type de prédicteur qui a été déjà utilisé par [Reitmayr et Drummond, 2007] pour prédire l'erreur entre les positions GPS et les positions fournies par la caméra pour ainsi corriger les mesures fournies par le GPS et les rapprocher de la bonne estimation pour ainsi réinitialiser le suivi visuel. En effet, ces derniers ont eu recours au processus Gaussien [Williams, 1997] qui permet de faire un apprentissage en ligne de l'erreur produite et par la suite utiliser la covariance obtenue pour prédire l'erreur à l'instant courant. Les résultats présentés ont l'air satisfaisant. Dans ce qui va suivre, nous allons tout d'abord présenter le principe d'un processus gaussien et de son fonctionnement. Puis par la suite, nous allons exposer comment nous l'avons utilisé dans notre système.

5.2.2.1 La régression avec le processus Gaussien

La régression linéaire est utilisée pour prédire des données à partir d'un ensemble d'observations. Cette prédiction se base sur une phase d'apprentissage qui consiste à définir les paramètres du modèle utilisé. La prédiction peut être faite en utilisant un processus Gaussien qui est considéré comme un prédicteur basé sur les données *a priori* et non sur les paramètres estimés *a priori*. Pour utiliser le processus gaussien pour la prédiction, nous allons nous baser sur l'étude faite par dans [Williams, 1997] présentant un tutorial sur la régression avec le processus gaussien. Mais avant cela, nous commencerons par présenter quelques définitions qui accompagnent le concept de processus Gaussien.

Définition 5.2.2.1. *Un processus stochastique $X = (X_t)_{t\in T}$ est une famille de variables aléatoires X_t indexées par un ensemble T.*

Un processus stochastique ou processus aléatoire ou fonction aléatoire représente une évolution dans le temps d'une variable aléatoire. Si T est un ensemble fini, le processus est un vecteur aléatoire. Si $T = \mathbb{N}$ alors le processus est une suite de variables aléatoires.

Définition 5.2.2.2. *Étant donné un processus stochastique $(X_t)_{t\in T}$, les lois finies dimensionnelles de X sont les lois de tous les vecteurs $(X_{t_1}, X_{t_2}, ..., X_{t_n})$ pour $t_1, ..., t_n \in T$ et $n \in \mathbb{N}$*

L'ensemble des lois finies dimensionnelles caractérise la loi $\mathbb{P}_X$ du processus X. Parmi les processus stochastiques, nous retrouvons le processus gaussien. Le processus gaussien est défini comme suit :

Définition 5.2.2.3. *Un processus est dit gaussien si toute ses lois fini dimensionnelles $\mathfrak{L}(X_{t_1}, ..., X_{t_n})$ sont gaussienne $(\forall n \in \mathbb{N}, \forall t_1, t_2, ..., t_n \in T)$*

En d'autres termes, on dit qu'un processus noté $X = (X_t)_t$ est gaussien si toutes combinaison linéaire $a_1X_{t_1} + ... + a_nX_{t_n}$ suit une loi gaussienne pour tout $n \in \mathbb{N}$, $t_1, ..., t_n \in T$ et $a_1, ..., a_n \in \mathbb{R}$. De plus, la loi d'un vecteur gaussien $(X_{t_1}, ..., X_{t_n})$ est connue à travers sa fonction caractéristique par le vecteur moyenne $(E[X_{t_1}], ..., E[X_{t_n}])$ et la matrice de covariance $(cov(X_{t_i}, X_{t_j})_{1\leq i,j\leq n})$. De ce fait, la loi gaussienne est connue dès qu'on se donne la fonction moyenne $a_(t) = E[X_t]$ et l'opérateur de covariance $K(s,t) = cov(X_s, X_t)$. Ainsi, la loi finie dimensionnelle de $(X_{t_1}, ..., X_{t_n})$ est alors la loi normale de dimension n $\mathcal{N}(a_n, K_n)$ avec $a_n = (a(t_1), ..., a(t_n))$ et $K_n = (K(t_i, t_j))_{1\leq i,j\leq n}$. Les fonctions a et K définissent donc toutes les lois fini dimensionnelles de X et donc aussi sa loi. Parmi les propriétés des processus gaussiens. Nous avons :

- Toutes les lois marginales d'un processus gaussien sont gaussiennes ;
- Toutes combinaison linéaire de lois marginales d'un processus gaussien est aussi gaussienne.

Pour résumer, un processus gaussien est un processus stochastique qui permet de générer un échantillonnage à partir d'une distribution de probabilité a priori, i.e. la fonction d'inférence bayesienne.
Si nous modélisons le problème de la prédiction en utilisant les processus gaussien, il se présente comme suit :

- Soit un ensemble de données $D = \{x_n, y_n\}_{n=1..N}$, tel que $y_n = f(x_n) + \varepsilon_n$.
- Le but est de prédire la distribution de y^*, étant donné un nouveau point x^* et la probabilité conditionnelle $P(y^*|x^*, D)$;
- Cette prédiction est obtenue par régression tel que $\hat{y}(x^*) = E[t^*|x^*, D]$

Nous avons si $P(f(x_1), ..., f(x_N)) = \mathcal{N}(0, K_N)$ et $P(\varepsilon_n) = \mathcal{N}(0, \sigma^2)$ alors $P(y_1, ..., y_N) = \mathcal{N}(0, K_N + \sigma^2.I)$. De plus, la prédiction :

$$P(y^*|y_n) = \frac{P(y^*, y_n)}{P(y_n)} = \mathcal{N}(\hat{y}^*, \sigma^{*2}) \tag{5.4}$$

En reprenant le formalisme présenté dans [Williams, 1997], le processus gaussien qui dérive de la régression linéaire et utilisé pour la prédiction se présente comme suit :

Soit $(x_1, x_2, ..., x_n)$ un ensemble de données associé à $(y_1, y_2, ..., y_n)$ telle que $y_i = f(x_i)$. Cet ensemble de données est considéré comme ensemble d'apprentissage. Nous voulons prédire la valeur y_{n+1} associé à la nouvelle donnée x_{n+1}.

Pour cela, nous considérons $(Y_1, ..., Y_{n+1})$ un ensemble de $n+1$ variables aléatoires qui suivent une distribution gaussienne de moyenne nulle et de matrice $(n+1)$x$(n+1)$ de covariance Σ_{n+1} qui se présente comme suit :

$$\Sigma_{n+1} = \begin{pmatrix} \Sigma_n & \kappa \\ \kappa^T & \kappa_{n+1} \end{pmatrix} \tag{5.5}$$

Sachant que Σ_n est une matrice nxn, κ est un vecteur nx1 et κ_{n+1} est un scalaire. Si nous avons $y_1, ..., y_n$ observations associées à $x_1, ..., x_n$, alors la distribution conditionnelle $P(Y_{n+1}|Y_1, ..., Y_n)$ suit une distribution gaussienne telle que :

$$\mu_{Y_{n+1}} = \kappa^T \Sigma_n^{-1} y^n \tag{5.6}$$

$$\sigma^2_{Y_{n+1}} = \kappa_{n+1} - \kappa^T \Sigma_n^{-1} \kappa \tag{5.7}$$

Où $y^n = (y_1, ..., y_n)^T$, $\kappa_{n+1} = cov(y_{n+1}, y_{n+1})$, $\kappa_i = cov(y_{n+1}, y_i)$ et $\Sigma_{ij} = cov(y_i, y_j)$. La covariance entre y_i et y_j est une fonction de x_i et x_j telle que :

$$cov(x_i, x_j) = cov(x_j, x_i) = \frac{1}{N - |i-j|} \sum_{n=1}^{N-|i-j|} x_n x_{n+|i-j|} \tag{5.8}$$

Voyant maintenant comment nous appliquons ce modèle.

5.2.2.2 Application au système d'assistance de localisation

Dans notre système, les x_i représentent les positions GPS et les orientations déduites de la centrale inertielle. Les y_i sont les erreurs estimées entre les poses fournies par le sous-système AL et les poses calculées par le sous-système de vision. Lors de la phase

d'apprentissage, on associe à chaque position GPS et angles d'orientations enregistrée l'erreur obtenue. Lorsque le processus de suivi échoue, le système prédit l'erreur pour ainsi corriger la position GPS et l'orientation de la centrale inertielle et converger vers la pose de la caméra. En utilisant le modèle présenté auparavant, la moyenne $\mu_{Y_{n+1}}$ représente notre erreur prédite.

Concrètement, pour une position GPS $p_{gps} = (x_{gps}, y_{gps})$, le système prédit l'erreur $(\mu_{x_{gps}}, \mu_{y_{gps}})$ en utilisant le processus gaussien. La correction de la position p'_{gps} se présente comme suit :

$$\begin{cases} x'_{gps} = x_{gps} + \mu_{x_{gps}} \\ y'_{gps} = y_{gps} + \mu_{y_{gps}} \end{cases} \tag{5.9}$$

Concernant la correction des orientations fournies par la centrale inertielle transformée dans le repère de la caméra R_{IMU}, celles-ci sont mises sous forme d'angle d'Euler $(\alpha_x, \alpha_y, \alpha_z)$ afin de pouvoir les utiliser dans le processus Gaussien et estimer une prédiction des erreurs moyennes $(\mu_{\alpha_x}, \mu_{\alpha_y}, \mu_{\alpha_z})$. Lors de la phase de correction, les angles d'Euler sont convertis sous forme matricielle. Si ΔR est la matrice obtenue, l'orientation déduite de la centrale inertielle est corrigée comme suit :

$$R'_{IMU} = \Delta R R_{IMU} \tag{5.10}$$

Au final la pose de la caméra fournie par le sous-système d'assistance est représentée par $(R'_{IMU}, -R'_{IMU} p'_{gps})$.

5.2.3 Réinitialisation automatique

Contrairement à l'approche semi-automatique décrite dans la section 2.4 (page 50), l'utilisateur n'intervient pas. Cette approche est utile pour reprendre le suivi. Son principe est le suivant : nous associons à chaque point 3D des descripteurs qui sont extraits autour de chaque projection 2D d'un point 3D du modèle. Pour cela, nous définissons une zone centrée autour du point projeté dans laquelle nous détectons des points caractéristiques. Nous avons opté pour le détecteur SURF [Bay et al., 2008].

(a) (b)

FIG. 5.3: (a) Les appariements des points SURF (b) Surpression des *outliers* avec le calcul de l'homographie

Ce détecteur se caractérise par sa robustesse et son invariance face à la rotation et aux changements d'échelle. Brièvement ce détecteur est une approximation de l'opérateur

SIFT. Il possède la même robustesse mais il est plus rapide. En effet, l'opérateur SURF se base sur une approximation de la Hessienne. Dans [Bay et al., 2008], les auteurs proposent d'utiliser des filtres box pour représenter les noyaux des filtres qui approchent le calcul des dérivées secondes de l'image. Le calcul de la Hessienne est effectué sur l'image intégrale qui correspond à la somme des valeurs entre chaque point et l'origine. L'utilisation de l'image intégrale permet d'accélérer les calculs. Les points d'intérêts correspondent aux maxima locaux. Le SURF associe à chaque point détecté un descripteur. Il décrit la répartition des intensités au sein d'une échelle dépendant du voisinage de chaque point détecté. Le fait d'apparier uniquement les points SURF définis autour des projections des points 3D permet de retrouver indirectement les appariements de ces points 3D. La question est comment retrouver ces appariements à partir de l'ensemble des appariements SURF ?

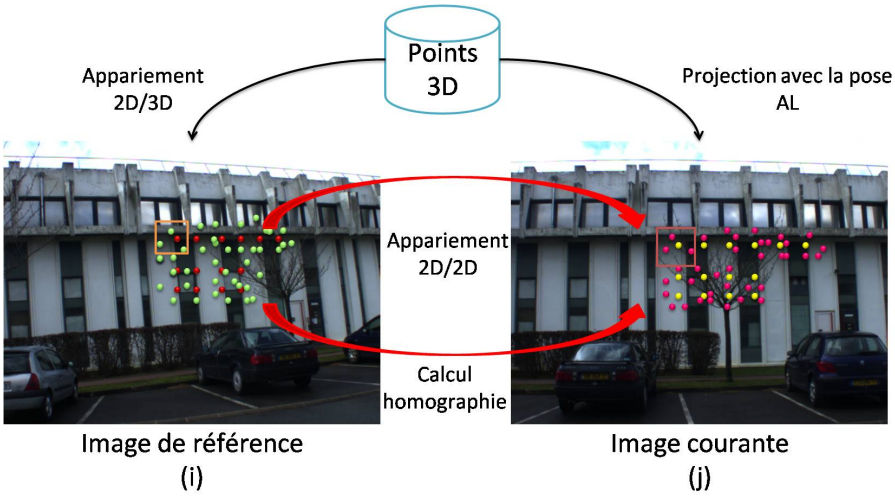


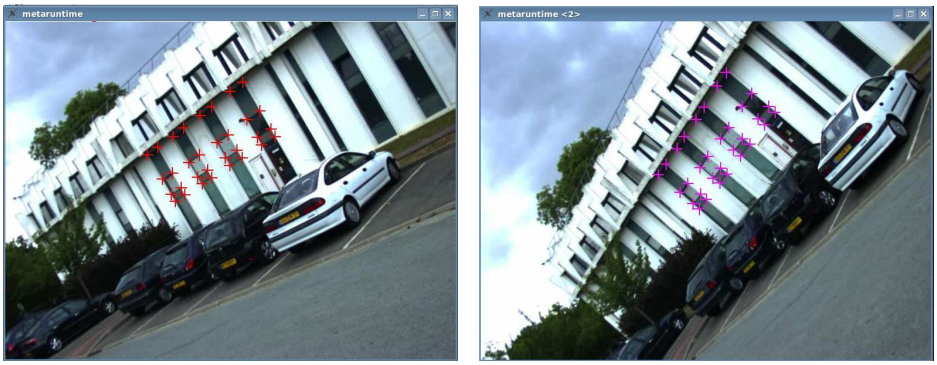
FIG. 5.4: Réinitialisation automatique : schéma illustratif

Pour retrouver les correspondants des projections des points 3D, nous décidons de chercher la transformation reliant deux images. Elle est exprimée sous la forme d'une homographie H_i^j qui définit une relation entre un couple d'appariement (m_i, m_j) telle que :

$$m_j = H_i^j m_i \tag{5.11}$$

Identifier cette homographie permet de transformer un point m_i appartenant à l'image i dans l'image j en le point p_j. Cette homographie est calculée à partir d'un ensemble d'appariements, dans notre cas à partir des appariements obtenus avec le SURF. Cette homographie permet de retrouver les correspondants des points 3D en transformant leur projection dans l'image i dans le repère de l'image j (cf. fig.5.5-b). De cette manière, si on connait les appariements 2D/3D à l'instant i, nous pouvons les retrouver à l'instant j. Pour rendre l'appariement robuste et pour éliminer les appariements aberrant, nous utilisons l'algorithme RANSAC [Fischler et Bolles, 1981]. Le calcul de l'homographie est détaillé dans l'annexe A. La figure 5.4 reprend le principe de la méthode.

Afin d'accélérer les calculs, nous utilisons les poses fournies par le sous-système d'assistance pour définir des zones de recherche dans l'image courante. Ces zones sont définies autour des projetés 2D des points 3D du modèle. Ceci a pour effet de réduire la zone d'extraction des points SURF dans l'image courante et ainsi réduire le temps de la phase d'appariement. De plus, en réduisant la zone de recherche le nombre de faux appariement diminue.



(a) (b)

FIG. 5.5: (a) Image de référence (b) Projection des points avec l'homographie calculée

5.3 Fonctionnement et réalisation du système

Après avoir présenté les différents composants du système que nous proposons, nous allons maintenant décrire son fonctionnement. Comme notre système suit un schéma de suppléance, ceci peut se traduire par le passage du système d'un état nominal du fonctionnement à un autre. De ce fait, nous pouvons identifier quatre états du système global à savoir :

1. un état d'*initialisation* : le système est défini dans cet état lorsque lors de la phase d'initialisation semi-automatique où le système effectue l'appariement 2D/3D ;
2. un état de *prédominance vision* : lorsque le système est dans cet état, cela veut dire que c'est la méthode basée vision qui est utilisée pour la localisation ;
3. un état de *prédominance AL* : dans cet état, le système fait appel au sous-système d'assistance à la localisation ;
4. un état de *réinitialisation* : en passant à cet état, le système tente de réinitialiser le sous-système de vision après son échec.

Le système basculera d'un état à un autre si un ensemble de condition est vérifié. Pour mieux schématiser ces transitions, nous utilisons le formalisme d'automate à états finis. Par définition, ce dernier est un modèle théorique composé d'un nombre fini d'états et de transitions entre ces états. Ce formalisme est principalement utilisé dans la théorie de la calculabilité et des langages formels. En utilisant les états présentés précédemment comme état définissant notre automate, les transitions décrites par l'automate permettent de contrôler notre système de localisation. L'état initial est défini par l'état d'*initialisation*. La figure 5.6 illustre l'automate que nous avons défini pour modéliser les états de notre système et ses transitions.

Suivant l'automate décrit dans cette figure (cf. fig.5.6), notre système de localisation fonctionne comme suit :

1. Initialement, le système se trouve à l'état d'*initialisation*. A cette étape, le système recherche les appariements 2D/3D en utilisant la procédure d'initialisation décrite dans le chapitre 2 (page 33) dans la section 2.4 (page 50). Pour rappel, un rendu du modèle filaire représentant l'environnement est réalisé à partir d'une pose prédéfinie. Ce rendu est aligné manuellement par l'utilisateur. Une fois validée, le système recherche les appariements. Les poses prédéfinies peuvent être déterminées hors ligne ou elles peuvent être fournies par le système d'assistance à la localisation ;

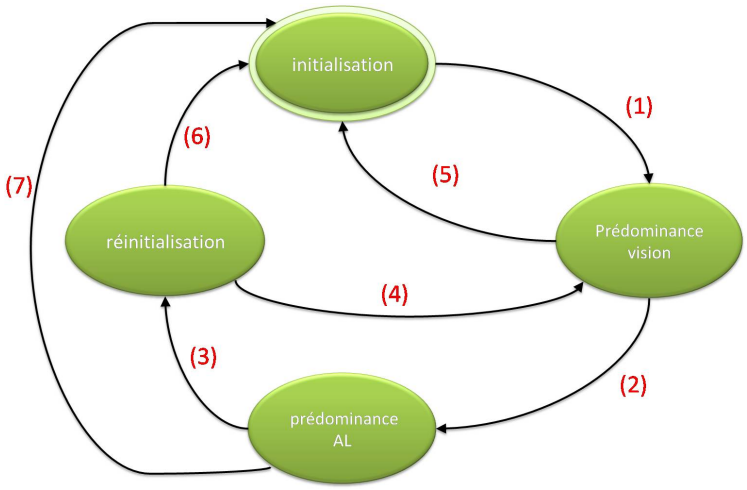


FIG. 5.6: fonctionnement du système hybride de localisation

2. Une fois l'initialisation effectuée et validée, le sous-système de vision est utilisé. Le système bascule alors de l'état d'*initialisation* vers l'état de *prédominance vision* (transition (1)) ;

3. Lorsque le système est dans l'état de *prédominance vision*, le système de localisation se base sur la méthode basée vision décrite dans le chapitre 2. Dans cet état, le système calcule les poses à partir du flux d'images. En effet, à chaque image reçue, le système calcule le nouvel ensemble d'appariement 2D/3D pour estimer la pose. Le système valide chaque pose calculée en se basant sur les critères définis précédemment. Si la pose est validée, elle sera utilisée pour la phase de recalage. De plus, elle sera utilisée dans la phase d'apprentissage de l'erreur dans le processus gaussien. Tant que la pose calculée est validée, le système de localisation reste au niveau de l'état de *prédominance vision* ;

4. Lorsque la pose calculée n'est pas validée, c'est-à-dire l'un des critères définis n'a pas été satisfait, le sous-système d'assistance à la localisation remplace le sous-système de vision et le système de localisation bascule de l'état de *prédominance vision* vers l'état de *prédominance AL* (transition (2)) ;

5. Lorsque le système de localisation est dans l'état de *prédominance AL*, la pose est fournie par le sous-système d'assistance de localisation. Les positions fournies par le GPS et les orientations sont transformées dans le repère local en utilisant les transformations obtenues à partir des procédures de calibration. La pose obtenue est par la suite corrigée avec la prédiction ;

6. Après un laps de temps, le système tente de réinitialiser le sous-système de vision. De ce fait, le système bascule de l'état de *prédominance AL* vers l'état de *réinitialisation* (transition 3) ;

7. Une fois dans l'état de *réinitialisation*, le système utilise la procédure d'appariement automatique décrite auparavant (cf. section.5.2.3 page 127) pour retrouver les couples d'appariements.

8. Si la réinitialisation est réussie, ce qui se traduit par l'obtention d'un nombre suffisant de bon appariement, le système de localisation bascule de l'état de *réinitialisation* vers l'état de prédominance vision (transition (4)) ;

9. Si le système ne réussit pas à réinitialiser le sous-système de vision, le système repasse à l'état d'*initialisation* pour réinitialiser le système en utilisant la procédure semi-automatique (transition (7)) ;
10. le système offre la possibilité à l'utilisateur d'intervenir en faisant basculer le système n'importe quel moment de l'état où il se trouve vers l'état d'initialisation (transitions (5) et (6)). En effet, si l'utilisateur juge par lui-même que le système ne fonctionne pas correctement, il peut forcer le système à basculer pour réinitialiser le système de vision pour fonctionner correctement.

L'utilisation d'un automate à états finis a été guidée par l'architecture logicielle utilisée pour développer ce système à savoir ARCS.

5.3.1 ARCS : Augmented Reality Components System

Proposé et développé par J-Y. Didier [Didier et al., 2009], le système ARCS (*Augmented Reality Component System*) est un outil qui offre la possibilité de réaliser un prototypage rapide pour le développement d'applications de réalité augmentée. Le système se base sur le paradigme de la programmation orientée composants. Il a pour objectifs principaux de :

- Réutiliser des applications, des méthodes et/ou des algorithmes déjà développés ;
- Implémenter facilement de nouveaux algorithmes au sein d'anciennes applications ;
- Effectuer des tests rapides et faciles ;
- Faciliter le déploiement de nouvelles applications.

Nous allons présenter quelques éléments clés utilisés par le système.

5.3.1.1 Composants

Elément principal d'une application, le composant (objet contenant du code compilé) est constitué d'entrées et de sorties appelées respectivement des **slots** et des **signaux**. Ces composants peuvent communiquer entre eux de manière synchrone. Cette communication est possible lorsqu'un signal d'un composant est connecté à un slot d'un autre composant. L'activation d'un slot peut déclencher un ou plusieurs signaux. L'exécution d'un composant ayant émis le signal est interrompu jusqu'à ce que le slot appelé a achevé son exécution. De plus, chaque composant peut posséder des slots utilisés pour l'initialiser.

5.3.1.2 Les feuilles (*sheets*)

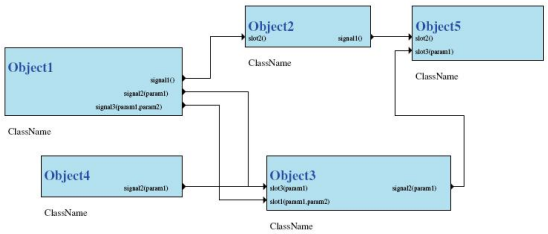


FIG. 5.7: Exemple de composition de composants pour la formation d'une feuille

La communication entre les composants s'effectue par des connexions entre les signaux et les slots. L'ensemble de composants et de connexions ainsi que la liste des initialisations est appelé feuille (ou *sheet*). La feuille résulte de la composition des composants. La figure 5.7 illustre un exemple d'une feuille représentant un ensemble de composants avec leurs différentes connections.

5.3.1.3 Automate

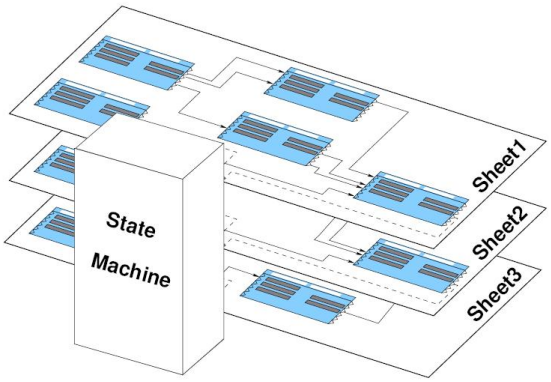


FIG. 5.8: Une application est la composition d'un automate et de plusieurs feuilles

Une application développée avec ARCS est une collection de feuilles. Ces feuilles représentent différents état du système. Les transitions représentant le passage d'une feuille à une autre sont régies par un automate à états finis (cf. fig.5.8). En effet, chaque état de l'automate correspond à une feuille. De ce fait, les actions d'une application sont fonction de l'état dans lequel l'automate se trouve. Ainsi le changement d'état est activé quand l'automate reçoit un jeton de la part de la feuille courante. Le jeton est envoyé par un des composants constituants la feuille active. Ceci active les changements de feuille.

En termes techniques, ARCS se base sur la bibliothèque Qt écrit en C++. Cette bibliothèque portable fournit une API qui permet de couvrir un bon nombre des fonctionnalités de base des systèmes d'exploitation. Ceci permet de garantir des composants qui peuvent être compilés sur plusieurs systèmes d'exploitation (Linux ou windows). Le grand avantage de l'utilisation de cette bibliothèque consiste dans le fait que le mécanisme signal/slot implémenté permet la connexion/déconnection à l'exécution. Si QT permet de décrire les composants d'une application, le langage XML (eXtensible Markup Language) est utilisé pour décrire les feuilles répertoriant les connections entre les différents composants développés pour l'application ainsi que l'automate associé à cette application. La figure 5.9 présente l'organisation d'un fichier XML avec la description associé aux différents composants d'une application ARCS (**object** pour composant, **sheet** pour feuille, **statemachine** pour l'automate).

Une fois que les bibliothèques comprenant les composants compilés sont créées, le ficher XML décrivant l'application établi, l'application est exécutée à l'aide d'un moteur d'exécution (cf. fig.5.10). Ce dernier charge les composants et les faits communiqué entre eux conformément à la description fournie par les feuilles et l'automate. Le moteur d'exécution est composé de quatre éléments essentiels :

- Un parseur de fichier XML qui lit la description de l'application ;
- Un gestionnaire d'automate qui effectue le basculement des feuilles et l'arrêt de l'application si cela est nécessaire ;
- Un gestionnaire de communications qui connecte et déconnecte les composants à la demande ;
- Un gestionnaire de composants qui charge les composants à partir des bibliothèques et instancie et détruit ces derniers à la demande.

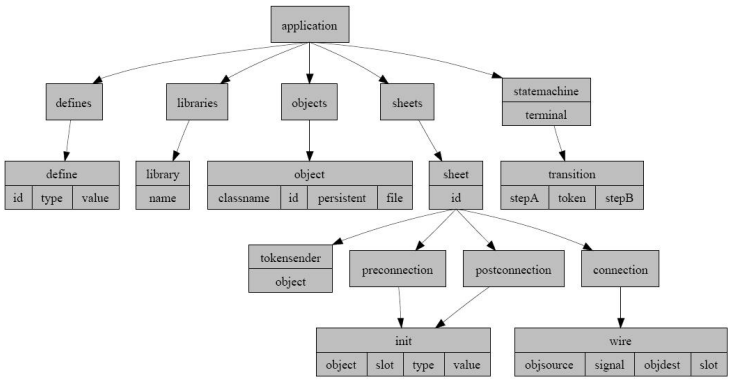


FIG. 5.9: Organisation d'un fichier XML pour la description d'une application avec ARCS

Lors du lancement de l'application, le moteur d'exécution lance la feuille associée à l'état initial de l'application. L'arrêt est effectué lorsqu'une feuille correspondant à un état terminal de l'application est atteinte.

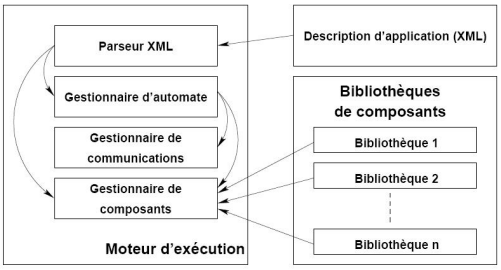


FIG. 5.10: Descriptif du moteur d'exécution

Un aperçu des composants développés avec ARCS sont présentés dans l'annexe E.

5.4 Expérimentations et Résultats

Dans les chapitres précédents (chapitres 2 et 4), nous avons présenté quelques protocoles d'expérimentations qui se sont intéressés aux performances d'une partie de notre système de localisation. Dans cette section, nous allons, dans un premier temps, nous concentrer sur les parties complémentaires que nous avons rajoutées en vue d'intégrer les deux sous-systèmes. Ainsi, nous présentons quelques résultats pour démontrer l'efficacité et les performances de l'approche de réinitialisation. Par la suite, quelques résultats sur le processus de prédiction et de correction de l'erreur seront exposés. Ceci nous permet de démontrer les améliorations apportées au système d'assistance à la localisation. La dernière partie des expérimentations concerne le fonctionnement propre de notre système de localisation. Suivant les différents tests effectués, nous allons montrer comment notre système se comporte devant les situations auxquels il est confronté.

5.4.1 Evaluation des performances de l'approche de réinitialisation

Afin de caractériser notre approche de réinitialisation, nous appliquons la méthode décrite dans la section 5.2.3 sur des paires d'images prises selon des points de vue variés.

La première image est définie comme image de référence et la seconde image est utilisée comme image courante. Nous sélectionnons un ensemble de points caractéristiques dans l'image de référence et nous cherchons leur correspondant dans l'image courante en utilisant notre approche. La figure 5.11 illustre quelques résultats obtenus sur les points définis dans l'image de référence (cf. fig.5.11-a (en rouge)) avec ce que nous obtenons pour chaque point de vue (cf. fig.5.11-(b-d) (en magenta)).

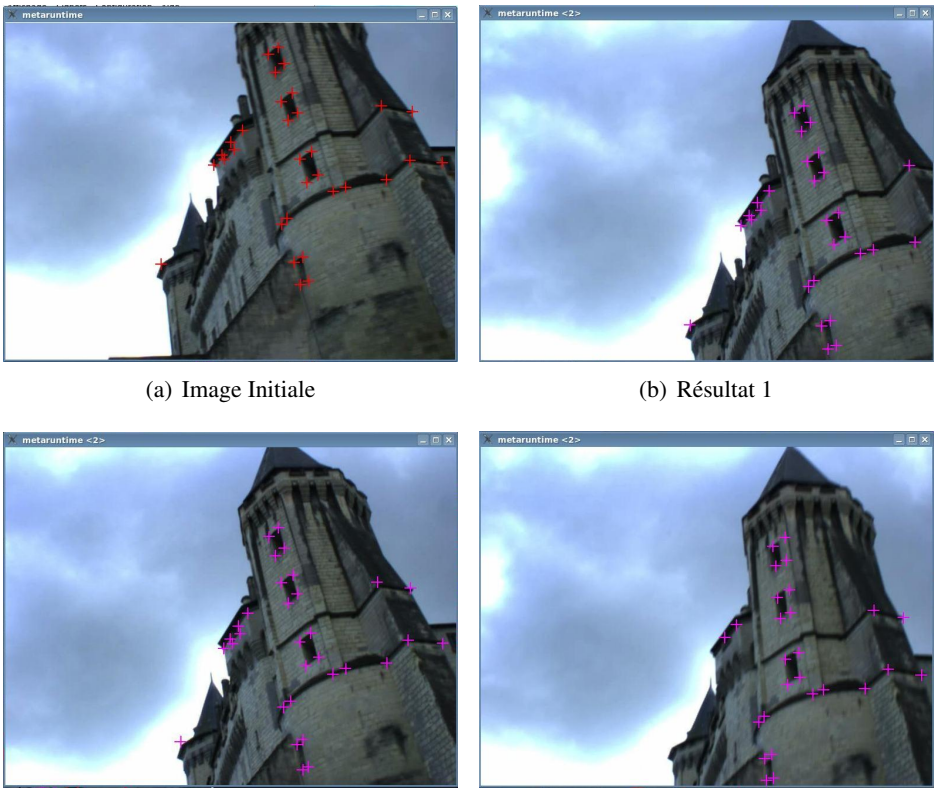
(a) Image référence (b) Résultat 1

(c) Résultat 2 (d) Résultat 3

FIG. 5.11: Résultats de l'utilisation de la réinitialisation : Exemple de points coplanaires

L'approche fonctionne bien. En effet, dans les 3 cas considérés, elle arrive à générer des bons appariements 2D/2D avec un taux de réussite de 100%. Dans cet exemple, l'ensemble des points de référence est coplanaire. Ceci laisserait entendre que l'approche ne fonctionnerait que pour ce type de points or ce n'est pas le cas. Pour vérifier ceci, nous testons notre approche sur un autre type d'environnement où les points caractéristiques ne sont pas coplanaires. Ceci est illustré dans la figure 5.12-a.

En observant les résultats obtenus (cf. fig.5.12(b-d)), nous constatons que notre approche donne aussi de bon résultats pour les points non coplanaires. En effet, le fait de prendre des points SURF autour des points caractéristiques permet d'avoir une bonne estimation d'homographie et de ce fait retrouver notre ensemble de points 2D qui ne sont autre que les projections des points du modèle 3D. Nous aurions pu chercher directement les appariements de ces points. Le problème se pose lorsque le déplacement entre la vue où le suivi décroche (vue de référence) et la vue courante est large. Ceci rend la méthode sensible aux faux appariements. De même, la méthode d'initialisation décrite dans la section 2.4 (page 50) n'est pas adaptée dans ce cas car elle fonctionne que si les projections des points 3D sont dans le voisinage proche de leurs correspondants.



(a) Image Initiale (b) Résultat 1

(c) Résultat 2 (d) Résultat 3

FIG. 5.12: Résultats de l'utilisation de la réinitialisation : Exemple points non-coplanaires

Pour analyser le principe de cette approche, nous calculons l'erreur entre les points obtenus avec notre approche et les points extraits de l'image. Au final, nous obtenons une erreur moyenne de l'ordre de 1.7823 pixels avec un écart type égal à 0.6634 pixels. Ce résultat est très satisfaisant.

Le problème qui peut se poser avec les approches de réinitialisation concerne les temps d'exécution. En effet, la phase de réinitialisation doit être faite de manière assez rapide et non perceptible par l'utilisateur lors de la phase de suivi. Le tableau 5.1 récapitule les temps d'exécution obtenus par phases. Pour indication, le tableau comporte les temps obtenus en utilisant la pose fournie par l'assistance à la localisation pour prédire la position des projections des points dans l'image courante ainsi que les temps obtenus sans cette prédiction. Nous constatons que l'incorporation de la prédiction a permis de diviser le temps de calcul par deux ce qui est une nette amélioration. Certes les temps obtenus sont assez élevés mais ceci reste acceptable. Par comparaison, l'approche de réinitialisation proposée par [Reitmayr et Drummond, 2007] présente un temps de calcul équivalent à 3 secondes (3000 ms).

5.4.2 Apport de la prédiction/correction

Avant d'aborder les expérimentations sur le système global, nous allons nous intéresser au processus de prédiction et analyser son apport vis à vis de la localisation. Pour cela, le protocole se déroule comme suit. Nous évoluons dans l'environnement en calculant

Etape [ms]	Sans Prédiction	Avec Prédiction
Extraction vue de référence	15	15
Extraction vue courante	370	240
Appariement	20	10
RANSAC	10	10
Total	415	275

TAB. 5.1: Comparaison des temps d'exécution de la réinitialisation : avec et sans prédiction

d'une part les positions avec la vision et d'autre part nous récupérons les positions GPS transformées dans le repère local. Puis, à partir de l'ensemble de données collectées, nous appliquons la prédiction pour retrouver les erreurs de la localisation avec le GPS.

Pour la prédiction de la données n, nous utilisons un ensemble de 5 données acquises précédemment. Ce nombre a été choisi à partir de plusieurs expérimentations où nous avons fait varier la taille de l'ensemble pour l'apprentissage puis observer l'influence sur l'erreur prédite. Nous avons constaté que l'augmentation de la taille de l'échantillonnage n'améliorer pas énormément la qualité de la prédiction et un apprentissage avec 5 données était amplement suffisant. Il faut savoir aussi que plus l'ensemble d'apprentissage est grand, plus la prédiction devient consommatrice en temps de calcul (la taille de la matrice de covariance augmente avec le nombre de données). De ce fait, la taille choisie représente un bon compromis entre efficacité et temps de calcul.

Les figure 5.13 et 5.14 illustrent un tracé comparatif entre les erreurs obtenues à l'issue de la prédiction (en bleu) avec les erreurs estimées (en rouge) entre les positions GPS et les positions calculées avec la méthode basée vision.

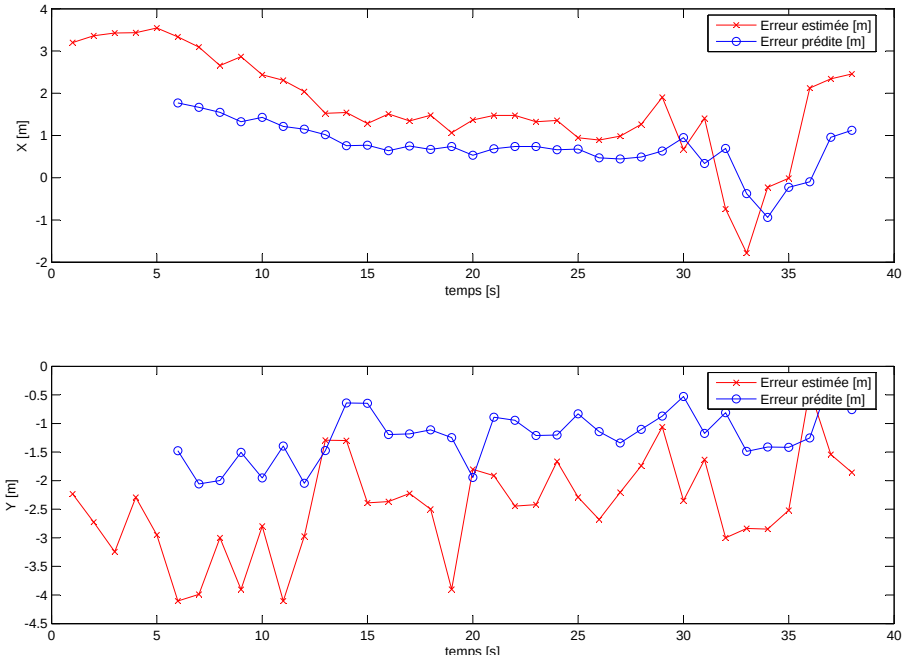


FIG. 5.13: Comparaison entre les erreurs prédites (en bleu) et les erreurs estimées (rouge) obtenue sur premier jeu de données

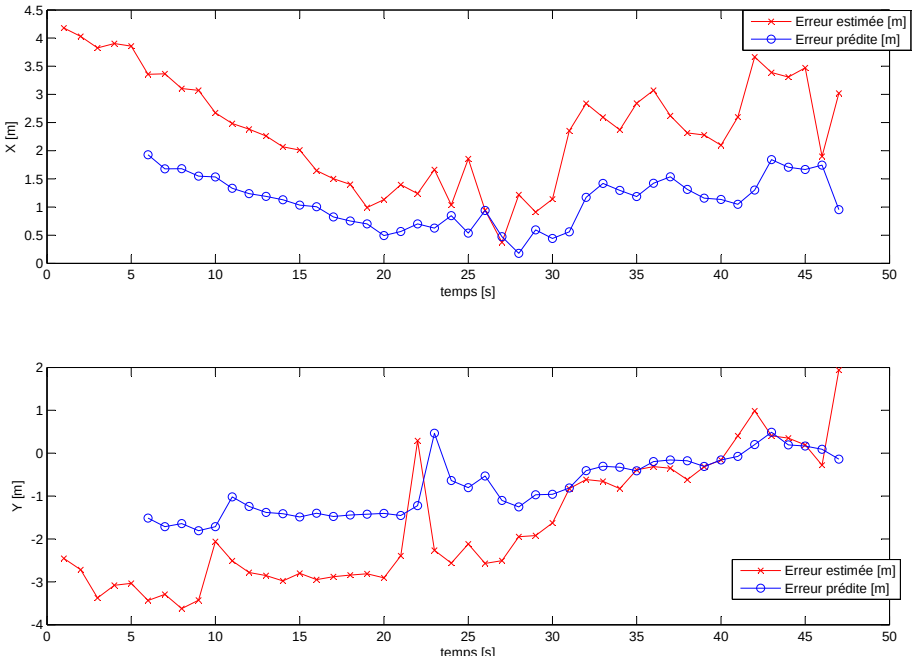


FIG. 5.14: Comparaison entre les erreurs prédites (en bleu) et les erreurs estimées (rouge) obtenue sur un second jeu de données

Pour mieux interpréter les résultats obtenus, nous calculons les différences entre les erreurs prédites et les erreurs estimées. Ces différences sont résumées dans le tableau 5.2. Nous constatons que l'erreur entre les deux positions est prédite avec un décalage de $1m$ en moyenne sur chaque axe avec un intervalle de confiance de +/- 0.7m.

		moyenne	Ecart type
Test 1	Axe X	0.9090	0.4572
	Axe Y	1.2710	0.7016
Test 2	Axe X	1.0893	0.5524
	Axe Y	1.0025	0.7348

TAB. 5.2: Les erreurs de prédiction

En observons les trajectoires obtenues (cf. fig.5.15), nous constatons qu'avec la prédiction, la trajectoire du GPS (en bleu) après correction (en noir) se rapproche de ce que donne la vision (rouge). Certes, nous avons toujours un certain décalage entre les deux positions mais celui-ci est réduit.

5.4.3 Evaluation du comportement du système de localisation

Après avoir étudié les performances liées aux différentes parties composants notre système, nous allons passer maintenant à son fonctionnement. L'objectif des expérimentations à venir est d'analyser son comportement face aux situations auxquels il peut être confronté. Ces situations peuvent être résumées essentiellement par :

1. L'occultation des points suivis : ceci peut être causé par des objets ou bien par le mouvement propre de la caméra ;

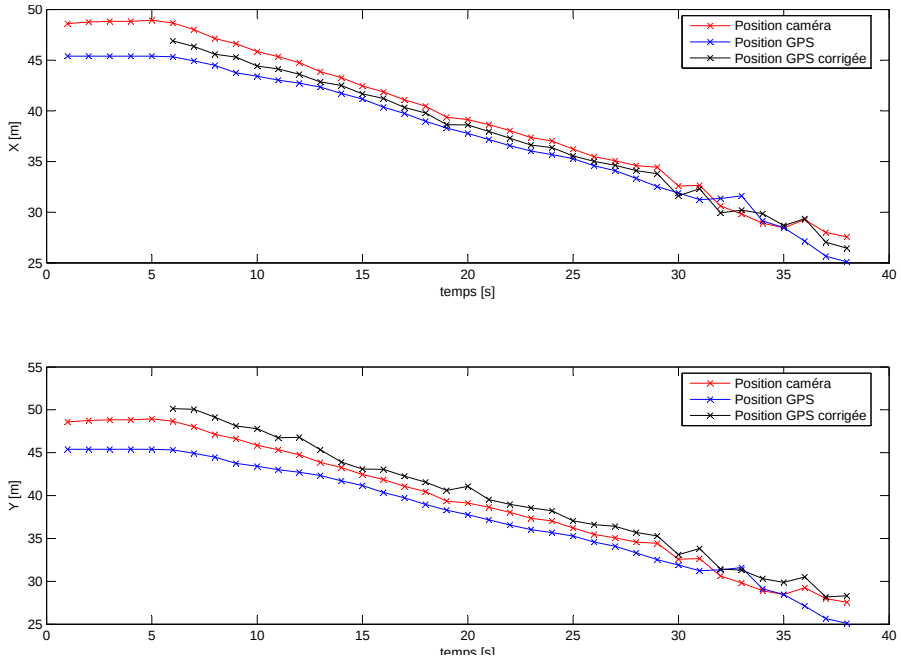


FIG. 5.15: Positions caméra (rouge) vs. positions GPS (bleu) vs. position GPS corrigée (noir) (obtenue sur le premier jeu de données)

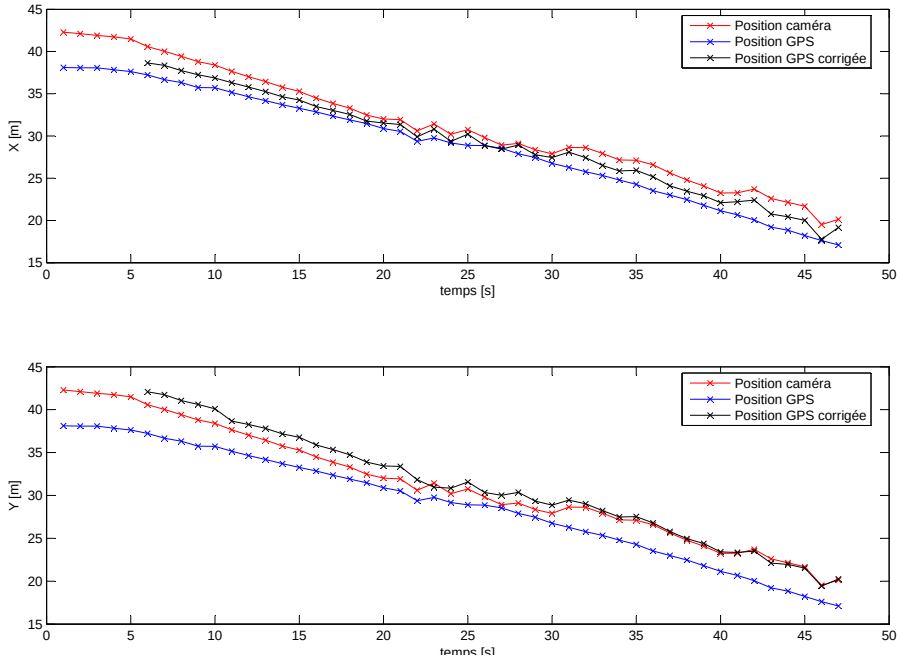


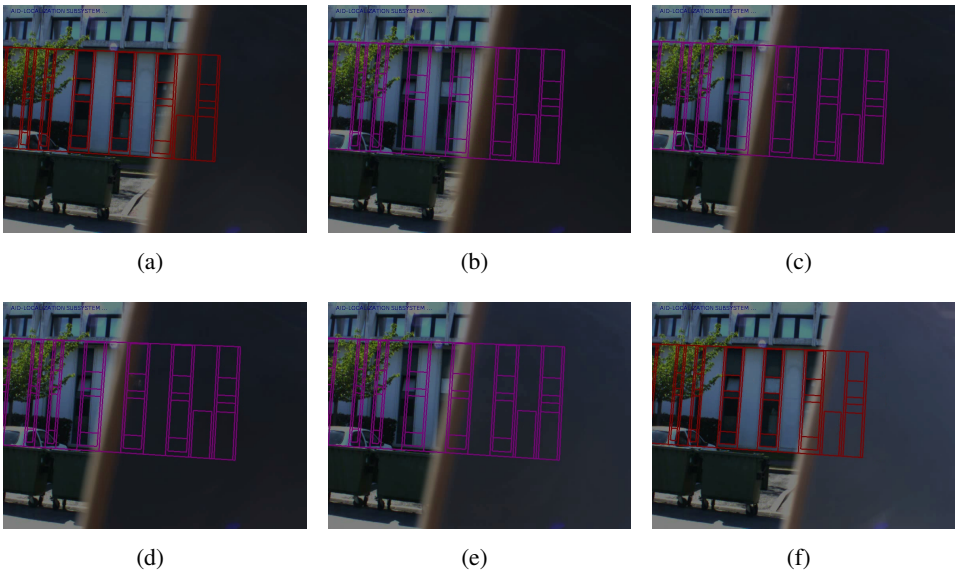
FIG. 5.16: Positions caméra (rouge) vs. positions GPS (bleu) vs. position GPS corrigée (noir) (obtenue sur le second jeu de données).

2. Les variations de luminosité ;
3. Les mouvements brusques et rapides de la caméra portée par l'utilisateur.

Nous allons tester notre système dans ces différentes situations. Pour cela, le système porté par un utilisateur évolue dans un environnement extérieur à grande échelle. Les mouvements de l'utilisateur ne sont pas contraints. Durant ce temps, le système estime la position et l'orientation de la caméra suivant la description faite auparavant. Pour visualiser ce que donne le système, nous allons nous baser sur le recalage obtenue avec la projection d'un modèle filaire représentant l'environnement (ou plutôt la construction faisant partie de l'environnement où évolue l'utilisateur). Nous optons pour un code de couleur afin de différencier entre les deux sous-systèmes opérationnels. Ceci se traduit sur la projection du modèle filaire dans le flux vidéo. Ainsi, si ce recalage est obtenu avec les poses fournies par le sous-système de vision, le modèle sera visible en rouge. Sinon si les poses sont calculées avec le sous-système d'assistance à la localisation, nous visualiserons le modèle en magenta.

5.4.3.1 Cas d'occultation partielle

Nous avons testé le comportement de notre système de localisation face à des cas d'occultations partielles. Le protocole expérimental se déroule comme suit. Lors du fonctionnement du système, nous provoquons des occultations de la scène en obstruant une partie des points utilisés pour le calcul de la pose et nous analysons la réaction du système.



(a) (b) (c)

(d) (e) (f)

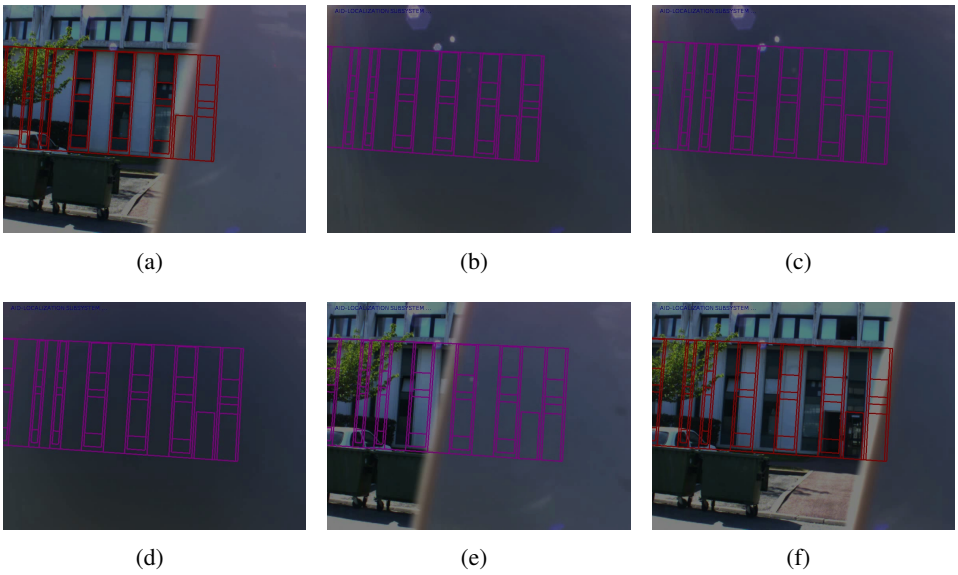
FIG. 5.17: Résultats obtenus dans le cas d'occultation partielle

Nous retrouvons dans la figure 5.17 un exemple de résultats obtenus. Nous pouvons observer que dans les images présentées dans 5.17-(b-e)n le système utilise le sous-système d'assistance pour recaler le modèle filaire. Concrètement, le système de localisation détecte qu'il n'y pas suffisamment de points pour calculer la pose en utilisant le sous-système de vision. Ainsi, le système bascule vers le sous-système d'assistance pour que celui-ci fournisse la pose nécessaire au recalage. Entre temps, le système de localisation tente de

reprendre la vision et lorsqu'il réussit à obtenir un certain nombre d'appariement, le sous-système de vision reprend son rôle comme nous pouvons le remarquer dans la figure 5.17-f. Nous avons effectué plusieurs fois ces tests et à chaque fois le système arrive à s'adapter à la situation. Point de vue du recalage, lorsque le système de localisation fait appel au sous-système d'assistance, nous pouvons voir que le modèle filaire se recale correctement sur la vue réelle. Certes, ce recalage n'est pas précis comparé à ce que donne la vision, mais ceci reste suffisant. De plus, les projections des points 3D sont dans le voisinage de leur correspondant, ce qui contribue dans la phase de réinitialisation.

5.4.3.2 Cas d'occultation totale

Dans ce cas tous les points 3D ne sont plus visibles. Nous laissons notre système fonctionner puis à un certain moment nous occultons totalement la façade du bâtiment et alors le suivi visuel perd les points 2D. Le système bascule vers le sous-système d'assistance tant que la façade reste cachée (cf. fig.5.18-(b-f)). Dans ces cas d'occultation totale, la précision du recalage est moins importante vu que nous ne voyons rien de la scène réelle. Ce qui est intéressant c'est de reprendre le suivi lorsque la façade devient visible. Nous pouvons observer que le système de vision arrive à reprendre le suivi comme le montre la figure 5.18-f.



(a) (b) (c)

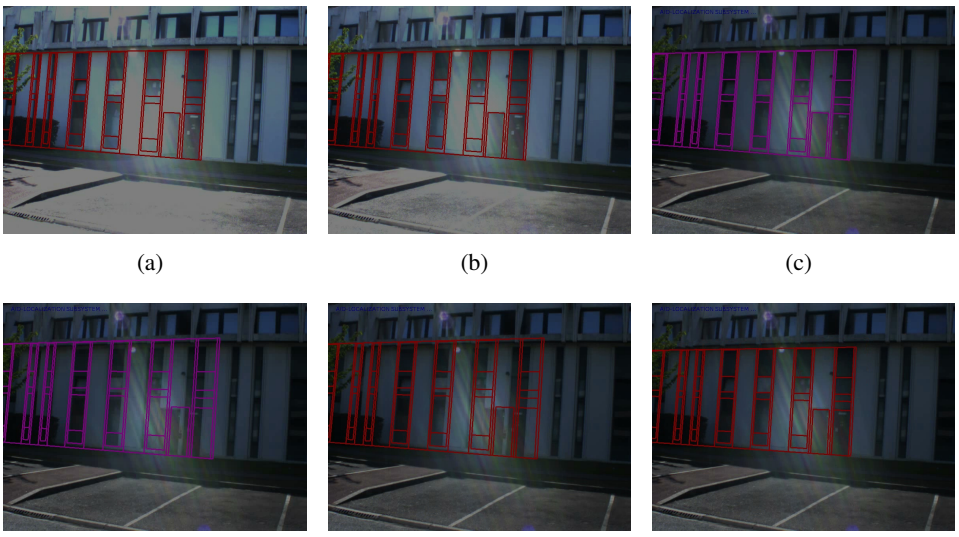
(d) (e) (f)

FIG. 5.18: Résultats obtenus dans le cas d'occultation totale

5.4.3.3 Cas de variations de luminosité

Autre cause qui peut influencer le comportement du système, les variations de la luminosité agissent directement sur le suivi visuel et peuvent engendrer des faux appariements. Nous confrontons notre système de localisation à une telle situation et nous observons son comportement. La figure 5.19 présente un exemple de variation de luminosité où nous observons que l'image devient plus sombre (cf. fig. 5.19-b et 5.19-c). En examinant les résultats obtenus, nous pouvons conclure que notre système à passer le test avec succès. En effet, lorsque la luminosité varie le suivi visuel échoue. Le sous-système d'assistance le remplace. Nous pouvons observer que l'approche de réinitialisation a réussit à retrouver les

appariements malgré qu'entre les différents basculements la luminosité à changer. Ceci est du à l'utilisation des descripteurs SURF qui ont l'avantage d'être invariants aux variations de luminosité.



(a) (b) (c)

(d) (e) (f)

FIG. 5.19: Résultats obtenus dans le cas de variation de luminosité

5.4.3.4 Cas de mouvements brusques

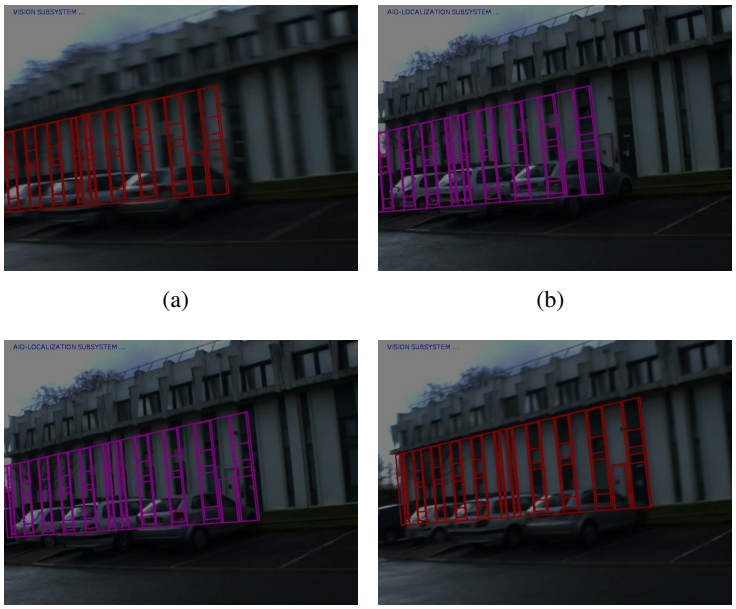
En situation de mobilité, les mouvements de l'utilisateur ne sont pas toujours lisses, uniformes et lents. En effet, ceux-ci peuvent être brusques et ainsi créent des images floues. En présence de ces dernières, le suivi visuel décroche. De plus, lorsque nous sommes face à des mouvements rapides, le déplacement image peut être important et de ce fait, le suivi visuel n'arrive pas à retrouver les appariements ou bien à tendance à engendrer des mauvais appariements. Nous avons par exemple dans la figure 5.20-b une image qui est floue à cause du mouvement rapide de la caméra engendré par la mobilité de l'utilisateur.

Notre système détecte la présence du flou généralement via le suivi visuel qui renvoie un nombre insuffisant de point ou bien via l'erreur de reprojection qui sera élevée. Comme nous le remarquons dans la figure 5.20, le sous-système d'assistance devient fonctionnel pour redonner ensuite la main au sous-système de vision (cf. fig.5.20-d). Toutefois, nous avons constaté dans certains cas que le recalage issu de l'assistance à la localisation n'est pas assez bon. Ceci est dû au fait que parfois en présence de mouvement brusque, le suivi visuel ne décroche pas rapidement, et du coup, ceci influence les mesures utilisées pour la correction des estimations fournie par le sous-système AL.

5.4.4 Bilan

Un récapitulatif des contributions faites tout au long de cette thèse peut être résumé comme suit :

– Un système de localisation qui s'adapte aux conditions régnant dans l'environnement et qui ont une influence sur les approches basées vision. Ceci a été possible en adoptant une approche basée suppléance ;



(a) (b)

(c) (d)

FIG. 5.20: Résultats obtenus dans un cas de mouvement brusque

– Un système palliatif à la vision composé d'un récepteur GPS et d'une centrale inertielle ;
– Deux approches de calibration du capteur hybride simples qui ne requièrent pas de lourdes hypothèses simplificatrices ;
– Deux procédures pour corriger les données fournies par le sous-système d'assistance à savoir la prédiction/Correction et le recalcule en ligne d'un des paramètres de calibration ;
– Une approche d'initialisation et une autre de réinitialisation qui utilise des descripteurs pour obtenir l'ensemble des appariements 2D/3D.

L'idée que nous avons poursuivie tout au long de nos travaux est de proposer une solution logicielle afin de créer un système "intelligent". Par cela, nous proposons un système qui a la faculté d'analyser les données dont il dispose pour s'adapter aux conditions extérieures en changeant son état intérieur. Ceci a été possible en divisant le système en deux sous-systèmes complémentaires. Ils fonctionnent de manière alternative en utilisant un modèle d'automate à états finis. L'utilisation de l'automate offre la possibilité au système de basculer d'un état à un autre selon des critères prédéfinis. L'utilisation de l'automate a été encouragée par l'architecture logicielle employée dans notre système de localisation.

Les différentes expérimentations conduites tout au long de ce manuscrit ont présentés les performances de notre système. Celles-ci sont jugées satisfaisantes. Certes, nous n'obtenons pas les mêmes performances que les approches déployés en milieux intérieur, mais nos résultats restent similaires à ceux présentés par d'autres systèmes fonctionnant en extérieur sachant que nous sommes dans un environnement à grandes échelles et que obtenir des erreurs autour du mètre est suffisant. Les expérimentations effectuées dans différentes situations, nous ont permis d'observer que le système réagit bien et arrive à être autonome. En effet, le système bascule d'un état à un autre (i.e. d'un système à un autre) sans l'intervention de l'utilisateur.

Nous avons comparé notre travail à ceux basées suppléance de données à savoir dans [Aron et al., 2007] et [Maidi et al., 2005]. D'un côté, nous avons trois capteurs réparties sur deux sous-systèmes. L'utilisation du GPS a été possible étant donné que nous évoluons dans un environnement extérieur. Ceci évite de supposer que le mouvement entre deux images peut être approximer à une rotation. De plus, le GPS est une solution à la dérive de la centrale inertielle pour le calcul de la position. M. Maidi a tenté d'estimer la translation à partir des accélérations mais celles-ci ne peut se faire que durant un court laps de temps. D'un autre côté, la centrale inertielle est utilisée pour déduire l'orientation absolue alors qu'Aron l'utilise pour retrouver l'orientation relative et ainsi estimer une homographie rotationnelle. Celui-ci suppose que lorsque la vision échoue, le mouvement entre deux images peut être approximé à une rotation pure. L'utilisation de l'homographie est aussi liée au fait que la vision se base sur l'approche basée plan de [Simon et al., 2000]. L'utilisation d'homographie peut engendrer un effet de dérive causé par un cumul d'erreurs.

Par rapport aux approches basées fusion, nous nous sommes contentés de comparer notre travail à celui présenté par [Reitmayr et Drummond, 2007]. Nous choisissons ce travail car par rapport aux autres approches, c'est celui qui se rapproche le plus de notre système. En effet, ils proposent d'utiliser une caméra avec une centrale inertielle et un GPS. Le récepteur GPS est utilisé uniquement dans les processus d'initialisation et de réinitialisation. Alors que dans notre cas, les positions GPS sont exploitées pour fournir des poses pour le recalage. La position et l'orientation ne sont fournies que par le couplage inertiel/caméra qui se base sur un modèle de fusion. Les orientations de la centrale inertielle sont exploitées pour prédire une pose pour projeter les contours du modèle 3D et calculer la pose. Le fait d'utiliser des contours leur permet de gérer les occultations partielles. L'utilisation des contours suppose que le mouvement entre deux images est petit. Ceci leur permet d'utiliser uniquement les orientations. Cependant, en présence de grands mouvements ceci n'est plus vrai. Ceci peut entrainer l'échec du système dans ces cas. Dans notre approche aucune hypothèse de ce genre n'est utilisée même si au niveau du sous-système de vision nous utilisons le KLT, celui-ci fournie de bons résultats même avec un mouvement large. Nous pouvons relever que dans leur approche il faut disposer d'un modèle de contours avec des textures alors que dans notre système il nous suffit uniquement d'avoir un ensemble de points 3D avec leur descripteur.

Nous avons démontré qu'un système de suppléance peut être utilisé en extérieur. Notre système a été testé dans le cadre d'un projet ANR. Ce projet est le sujet de la prochaine section.

5.5 Application au projet RAXENV

Le travail que nous venons de présenter, rentre également dans le cadre d'un projet national, le projet de RAXENV, un projet exploratoire soutenu par l'Agence de Recherche National sous le réseau "Technologie logicielle. L'objectif de ce projet est de démontrer la faisabilité d'un système utilisant le paradigme de la réalité augmentée pour des applications dédiées aux sciences et techniques de l'environnement. L'idée est de proposer un système qui soit en termes de technologies en adéquation avec les conditions de travail des utilisateurs finaux et ainsi être adopté par ces derniers.

L'enjeu de ce projet est de pouvoir proposer un système qui a la faculté de :
– Se localiser spatialement à l'aide d'un positionnement primaire en utilisant des cap-

teurs de types GPS et en s'aidant de données naturelles identifiées dans l'environnement et dont nous disposons d'une connaissance *a priori*. Cette connaissance se traduit par des modèles 3D géo-référencés, des Systèmes d'information géographiques (SIG) ou des modèles numérique de terrain (MNT) ;
- Offrir à l'utilisateur des fonctionnalités de visualisation et d'interrogations des données, locales ou distantes, de surfaces ou sous-sol en s'aidant des méthodes d'interaction adaptées aux terminaux mobiles.

En termes de techniques, le projet s'est intéressé à différents aspects à savoir : le recalage réel/virtuel, l'accès à des données distantes dans un contexte de mobilité ainsi que la visualisation et l'interaction. Dans le cadre de ce projet, nous nous sommes intéressés à mettre au point :
- Un système de localisation basé multi-capteurs pour un recalage satisfaisant de données virtuelles ;
- Une méthode de représentation d'objets de sous-sol (sondages, tuyaux, cavités, ...) pour les visualiser dans un système de réalité augmentée ;
- Des méthodes d'interaction en adéquation avec la mobilité de l'utilisateur.

Ces méthodes ont été testées dans différents scenarios correspondants à différents métiers issus des sciences d'environnement :
- Un scénario dit instrumenté représenté par le château de Saumur, où la problématique métier est la géotechnique et la gestion de risques dans le cadre du suivi de la reconstruction d'un rempart. D'un point de vue recalage, bien que l'utilisateur soit en extérieur, celui-ci ressemble à une configuration en intérieur vu que l'espace de travail est confiné.
- Un scénario en milieu urbain où la gestion de réseau enterré (l'assainissement) constitue l'élément essentiel. Le géoréférencement est assuré par un SIG dont il est nécessaire de prendre en compte son imprécision.

Dans ce qui va suivre, nous allons voir la mise en œuvre de ce système du point de vue matériel et logiciel.

5.5.1 Matériel employé

Point de vue matériel, le système développé constitue un outil d'acquisition et de restitution des données. Le choix du matériel employé a été effectué conformément aux besoins exprimés par les utilisateurs finaux à savoir les géologues et les agents de la Lyonnaise des eaux.

D'une part, le système a la faculté de se localiser spatialement en combinant trois capteurs : une caméra et un récepteur GPS couplé avec une centrale inertielle. D'autre part, le système offre à son utilisateur des fonctionnalités de visualisation via un terminal mobile de type tablette-PC. Le choix s'est porté sur les tablettes-PC en raison de la démocratisation de leur utilisation dans ces métiers et en raison de la sécurité qui ne peut être assuré avec d'autres types de dispositifs de visualisation tels que les casques. Les tablettes-PC présentent un compromis entre capacité de calcul et mobilité.

A cela s'ajoute la faculté d'interroger des données (locales ou distantes) de surface et du sous-sol à l'aide d'interactions adaptées aux terminaux utilisés. Pour cela, le système emploie des *phidgets* [Greenberg et Fitchett, 2001] (cf. fig.5.21). Choisis par analogie avec les *widgets*, les composants d'interface graphique, un *phidget* est un composant physique d'interface utilisateur. Les phidgets offrent la possibilité de créer des interfaces tangibles de manière facile. Les *phidgets* sont constitués de blocs (potentiomètre, joystick, boutons

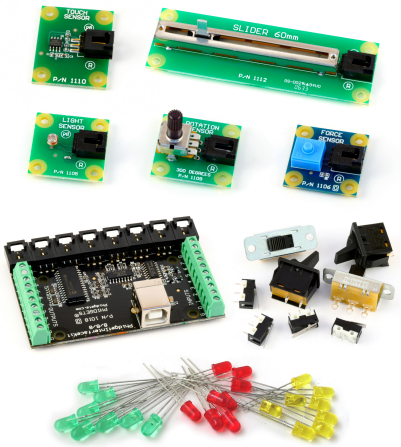


FIG. 5.21: L'ensemble de blocs composant le phidget

tactiles, accéléromètres, etc.) de construction qui sont connectés sur port USB et dotés d'une interface de programmation (API) unifiée.

5.5.2 Architecture logicielle

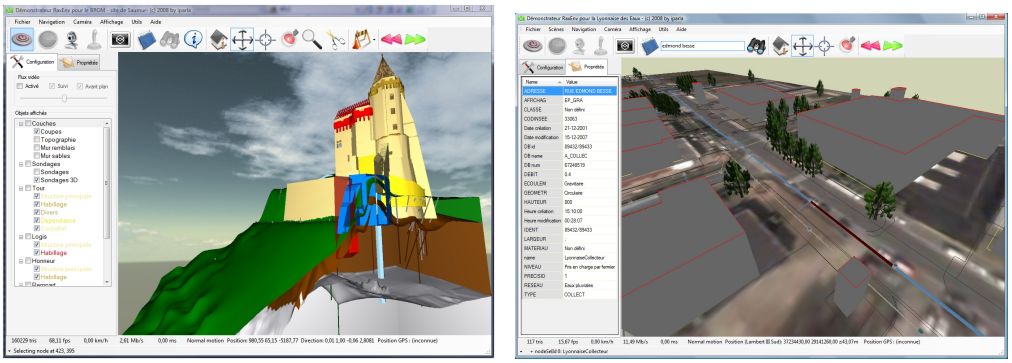
Si la bibliothèque ARCS a été utilisée pour le développement du module de localisation, en ce qui concerne l'interface de visualisation et d'interaction, nous avons eu recours à la plateforme `Elkano`. Elle offre la possibilité d'un développement facile et rapide de nouvelles solutions distribuées pour la visualisation à distance des modèles 3D. En effet, l'équipe **Iparla** du LaBri a procédé au portage de la plate-forme nommée `Magellan` sur des terminaux Mobiles communiquant (TMC) et l'a rebaptisé `Elkano`.

La plate-forme `Magellan` est une plate-forme de visualisation mise au point par J.-E. Marvie [Marvie, 2004]. Ecrite en C++, elle a pour but de faciliter le développement de nouvelles solutions de visualisation de scènes 3D interactives distribuées sur des machines hétérogènes. Cette plate-forme offre un ensemble de classes systèmes encapsulant des appels systèmes telle que les *sockets*, les *threads*, etc. afin d'assurer la portabilité. A cela, s'ajoute un méta-graphe de scène et des classes de nœuds pouvant être distribués. La plate-forme encapsule un système de modules offrant la possibilité d'enrichir la boucle principale interaction-rendu et un système de plug-ins permettant de décrire de nouveaux nœuds utilisables dans le graphe de scène. Le support du langage de description de scènes utilise VRML97. De plus, `Magellan` fonctionne sous différents systèmes d'exploitation (Windows / Linux / SunOS).

La plate-forme `Elkano` dispose de ces caractéristiques héritées de la plate-forme `Magellan` en incluant un support de la plate-forme Windows Mobile et donc la compatibilité avec les TMC. De plus, cette plate-forme incorpore un module de rendu distribué pour un groupe de PC reliés par un réseau TCP/IP. De ce fait, chaque machine se voit attribuée une partie de la zone d'affichage et une machine gère les interactions utilisateur et fait office de chef d'orchestre en assurant une barrière de synchronisation avant chaque passe de rendu. Cette technique simple mais efficace est en particulier utilisée pour effectuer du rendu temps-réel sur la grande surface d'affichage de la salle de réalité virtuelle du LaBRI, Hémicyclia 2. La plate-forme `Elkano` prend en charge des nœuds GeoVRML et un module d'interaction géoréférencé pour permettre la visualisation de scènes géoréférencées. De plus, le langage

de description de scènes X3D est utilisé dans cette version destinée essentiellement aux terminaux mobiles.

La figure 5.22 illustre l'utilisation de la plate-forme `Elkano` dans le cadre du projet RAXENV.



(a) Démonstrateur du château de Saumur (b) Démonstrateur urbain

FIG. 5.22: Aperçu de la plate-forme `Elkano` utilisée dans le projet RAXENV

5.5.3 Description de la plate-forme RAXENV

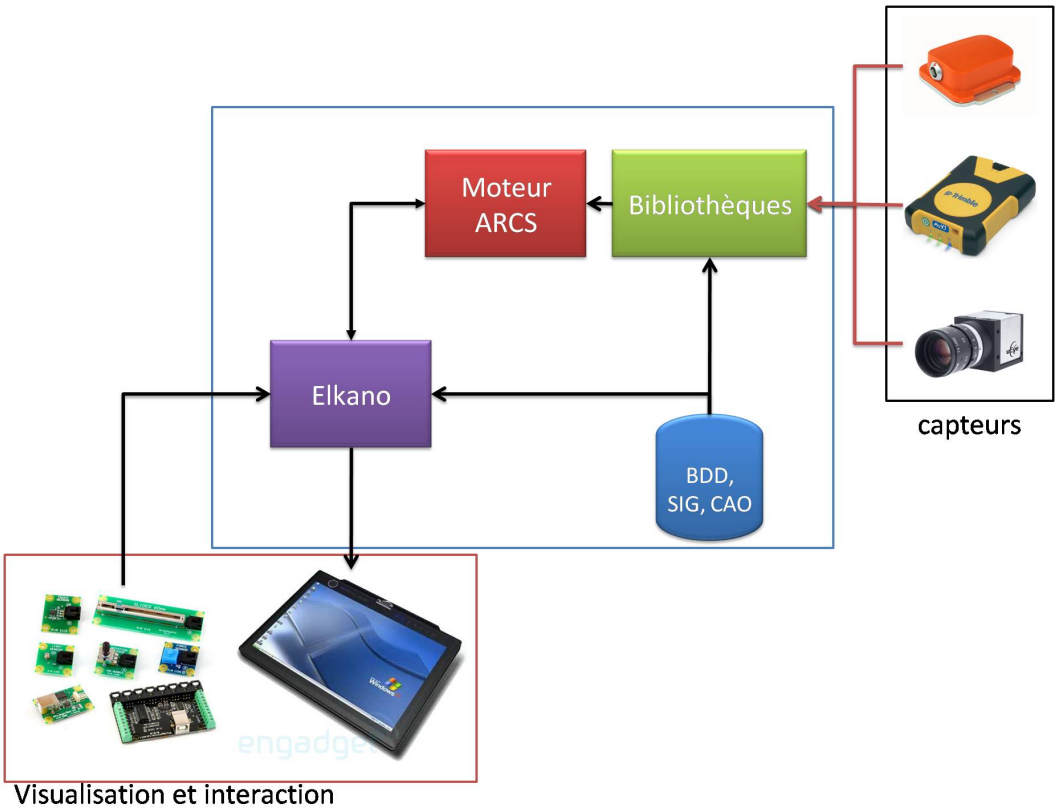


FIG. 5.23: Flot de données dans la plateforme RAXENV

Nous reprenons dans la figure 5.23 le flot de données de la plate-forme RAXENV. Celle-ci est calquée sur la description faite dans la section 1.2 (page 9) des systèmes de réalité augmentée en générale.

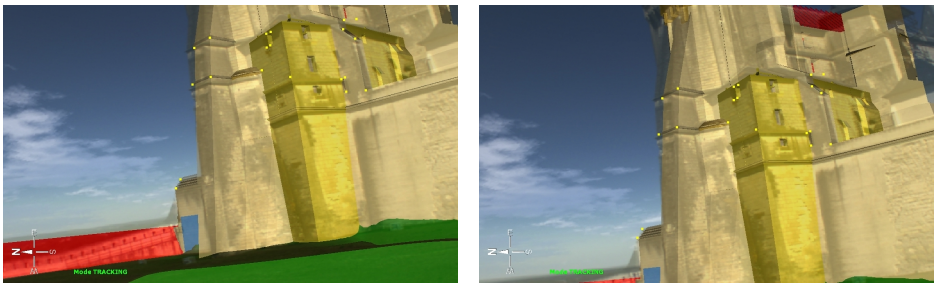
Côté capteurs, nous avons une caméra, un récepteur GPS et une centrale inertielle. Ceux-ci fournissent des informations sur l'environnement où évolue l'utilisateur. A cela s'ajoute la base de données qui est sous forme de modèle 3D représentant une partie de l'environnement réel. Ces capteurs et cette base de données sont connectés aux bibliothèques développées avec ARCS (cf. section 5.3.1 page 131). De plus, la base de données est connectée à la plate-forme `Elkano` pour extraire les données à visualiser.

Les bibliothèques développées sont composées essentiellement de la bibliothèque de localisation et des bibliothèques associées aux capteurs. La bibliothèque de localisation calcule la position et l'orientation à partir des données fournies par les capteurs et la base de données. Les autres bibliothèques interfacent les capteurs utilisés afin de récupérer les données fournies par ceux-ci et de les transmettre à la bibliothèque de localisation.

Ces données (i.e. pose estimée et données fournies par les capteurs) sont fournies à la plateforme `Elkano` qui se charge de réaliser un rendu de la scène augmentée. Celle-ci est visualisée via la tablette-PC qui est utilisée comme dispositif de restitution. De plus, à celle-ci sont connectées des *phidgets* qui offrent la possibilité d'interagir avec les données 3D. L'interface développée permet aussi d'envoyer des instructions au module de localisation lors de la phase d'initialisation semi-automatique par exemple.

5.5.4 Résultats

Le système RAXENV comprend différentes fonctionnalités qui sont offertes aux utilisateurs. Ces fonctionnalités ont été développées et conçu selon les attentes des utilisateurs finaux. La plus basique et principale est bien évidemment le recalage. Dans la figure 5.24, nous apercevons quelques exemples de recalage obtenu sur le site du château de Saumur lors d'une compagne de test. Dans cette figure, nous observons le modèle du château qui a été construit en utilisant un scanner 3D se superposer sur la vue réelle.

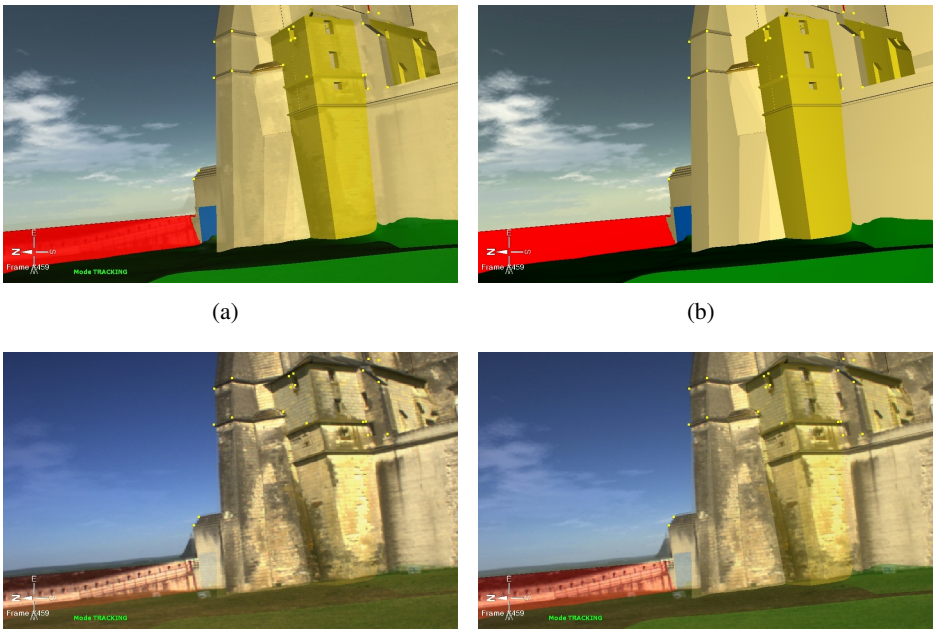


(a) (b)

FIG. 5.24: Exemple de recalage sur le site du château de Saumur

Parmi les options proposées à l'utilisateur, ce dernier a la possibilité de choisir la transparence des objets 3D recalés sur la vue réelle et ainsi privilégier de visualiser le château virtuel (cf. fig. 5.25-(a-b)) ou bien le réel (cf. fig. 5.25-(c-d)).

A cela s'ajoute la faculté de sélectionner une partie du modèle 3D en changeant de couleur (cf. fig.5.26-(a-b)) ou bien de supprimer des partie de ce modèle pour ne garder

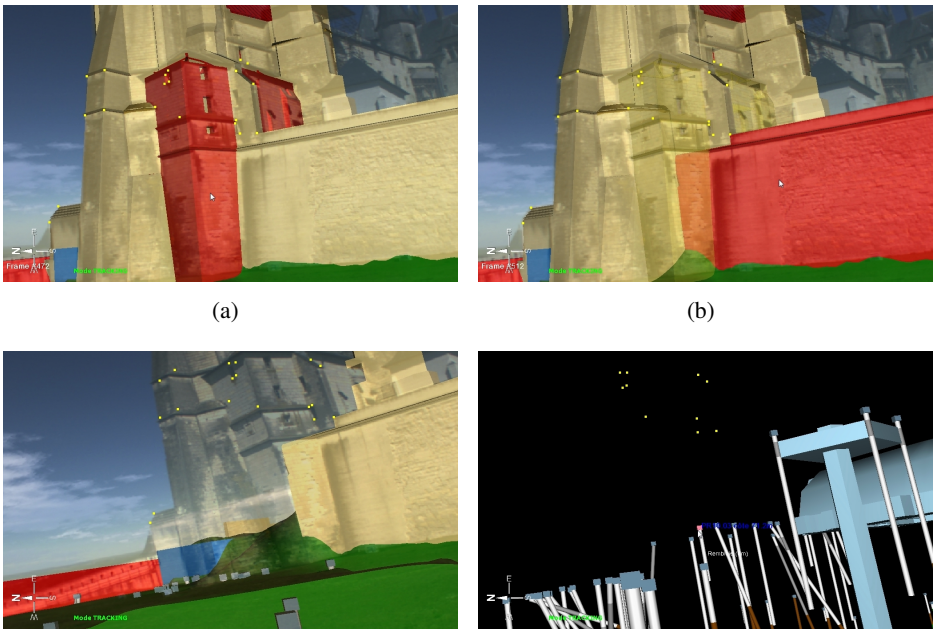


(a) (b)

(c) (d)

FIG. 5.25: Résultats de recalage avec différents degré de transparence

que la partie qui intéresse l'utilisateur (cf. fig.5.26-c-d).



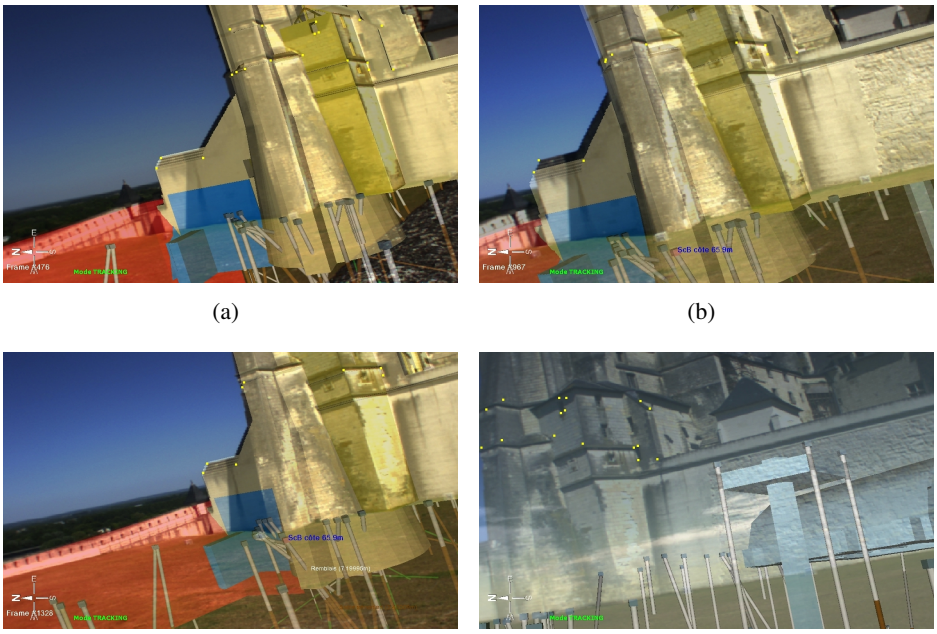
(a) (b)

(c) (d)

FIG. 5.26: Exemple de manipulation

En plus de visualiser le modèle 3D virtuel superposé sur ma vue réel, l'intérêt d'un tel système est de pouvoir visualiser ce qui n'est pas visible pour l'utilisateur. Dans le cas du projet RAXENV, l'utilisateur est plus intéresser par visualiser des sondes de sondages qui sont enfouies dans le sous-sol du site. Ceci lui permet de les localiser, et d'interroger

l'application sur ces têtes de sondages pour avoir divers information. La figure 5.27 présente quelques résultats obtenus durant les tests.

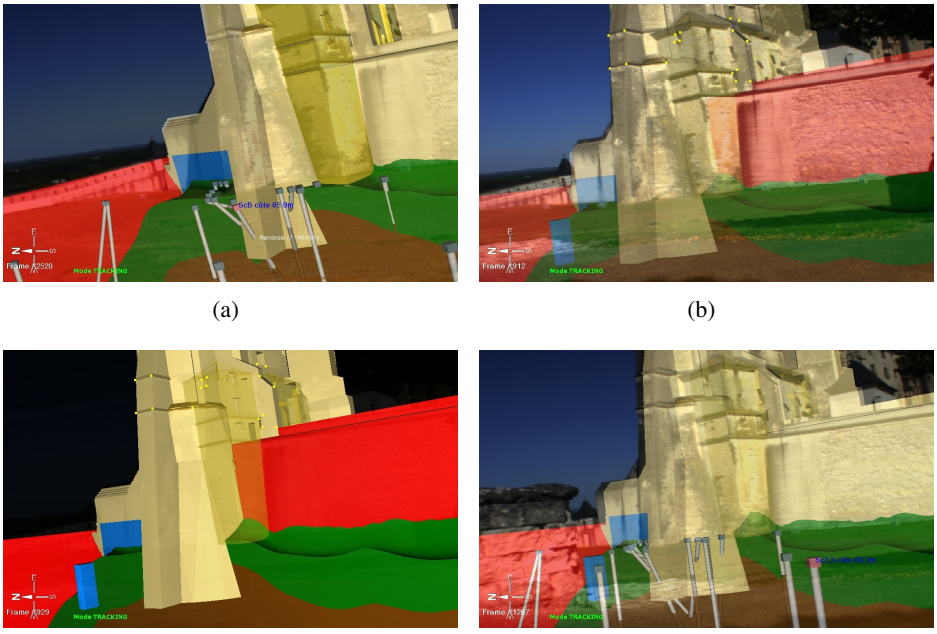


(a) (b)

(c) (d)

FIG. 5.27: Exemple de visualisation de sondes de sondage

A cela s'ajoute la faculté de faire des coupes dans le sol (cf.fig.5.28) et de réaliser ainsi en réalisant un clipping de pouvoir visualiser les couches du sous-sol, les sondages, etc.



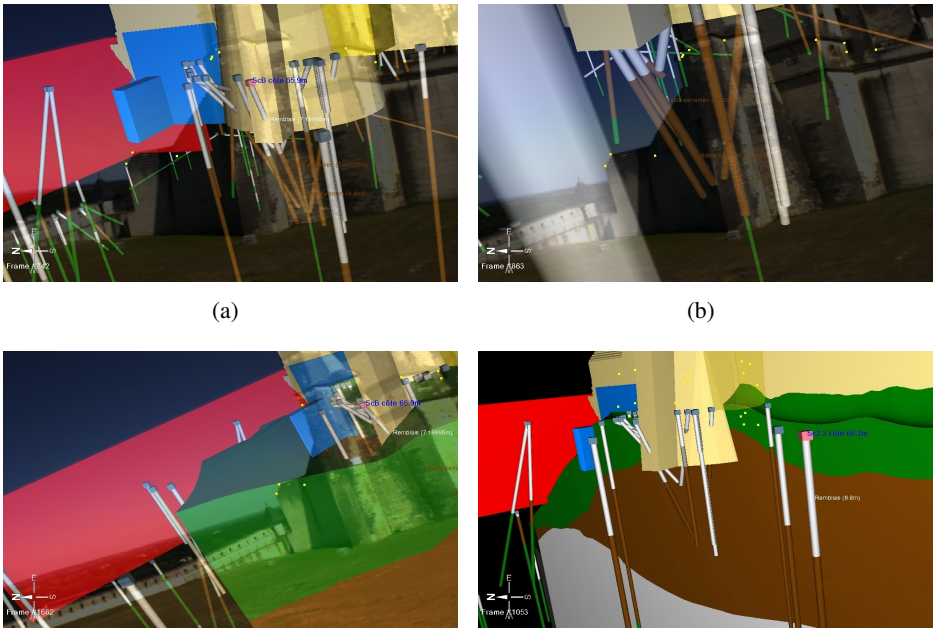
(a) (b)

(c) (d)

FIG. 5.28: Exemple de coupe dans le sol

Autre option proposée à l'utilisateur consiste à avoir le choix de ne pas recaler le modèle

pour pouvoir le manipuler avec aisance. Ceci n'empêche pas le système de localisation de fonctionner entre temps dans le but que lorsque l'utilisateur est fini d'utiliser ce mode d'utilisation de revenir en un clique à une vue augmentée avec un recalage correcte. La figure 5.29 illustre quelques exemples de manipulation du modèle sous ce mode.



(a) (b)

(c) (d)

FIG. 5.29: Exemple d'utilisation en mode sans recalage visuel

5.6 Conclusion

Nous avons présenté en détail notre système de localisation qui se base sur une approche de suppléance multi-capteurs. Ce système combine un sous-système de vision utilisant une approche basée points d'intérêts avec un sous-système d'assistance qui utilise un récepteur GPS et une centrale inertielle pour remplacer la vision. Ces deux sous-systèmes fonctionnent de manière complémentaire afin de disposer en continu d'une estimation de la pose de la caméra. Le système de localisation bascule entre ces deux sous-systèmes selon la qualité des mesures fournies par le sous-système principal à savoir la vision. En effet, la vision est le composant principal qu'utilise notre système pour fournir une bonne estimation de la pose. Cependant, lorsque l'estimation fournie par la vision n'est pas précise et ne respecte pas les critères de fiabilité, le système de localisation bascule alors vers le sous-système palliatif appelé assistance à la localisation. Entre temps, le système tente de reprendre le suivi visuel pour redonner la main au sous-système de vision. Pour cela, nous avons mis au point une approche de réinitialisation qui ne requière pas l'intervention de l'utilisateur. Les basculements d'un sous-système à un autre se basent sur la philosophie des automates à états fini. L'utilisation de ce type de structure a été facilitée et guidée par l'architecture logicielle utilisée ARCS.

Les expérimentations menées, sur les composants pris séparément et sur le système en sa globalité, ont démontré qu'une telle approche de suppléance était faisable en milieu extérieur. Comparé à d'autres types de système multi-capteurs, notre système de localisa-

tion fournit des performances similaires. Cependant, il tire son avantage du fait que nous sommes face à une solution logicielle qui offre une solution de remplacement pour une approche de vision qui peut échouer à tout moment. Notre système a la particularité de s'adapter à des situations complexes que la vision seule ne peut pas gérer. Par ailleurs, l'utilisation d'un système palliatif permet de reprendre ces approches classiques de vision qui ont prouvées leur efficacité et de les faire fonctionner de manière complémentaire. Dans ce cas, il suffit uniquement d'adapter ces approches en définissant des critères de validité qui soient en adéquation avec l'approche utilisée.

Conclusion générale et perspectives

Dans le cadre de cette thèse, nous nous sommes focalisés sur les aspects de la localisation en milieu extérieur. Cette problématique représente un enjeu important pour de nombreux domaines tels que la réalité augmentée qui connait depuis quelques temps un intérêt grandissant dans divers domaines d'applications. La localisation en milieu extérieur comporte plusieurs verrous scientifiques. Nous nous sommes particulièrement intéressés à ceux entourant la mise en œuvre de systèmes multi-capteurs notamment sur la stratégie à adopter pour la combinaison des capteurs, les approches de calibration du capteur hybride ainsi que la prédiction des erreurs des données issues des capteurs.

Dans un premier temps, nous nous sommes intéressés aux approches de localisation basées vision utilisées en réalité augmentée. Nous nous sommes orientés vers une approche sans marqueurs utilisant des points d'intérêts. Brièvement, celles-ci consistent à identifier la projection des points 3D sur le plan image pour calculer les paramètres de la pose en minimisant l'erreur de reprojection. Notre tâche s'est concentrée sur la phase d'appariement entre points 3D issus du modèle et points 2D extraits en ligne. Ainsi, nous avons mis au point une approche d'initialisation qui requière l'intervention de l'utilisateur. L'approche proposée utilise des descripteurs SURF associés aux points 3D. Cette approche est suppléée par un suivi 2D/2D afin d'identifier les projections des points 3D dans le flux d'images et ainsi maintenir la localisation en temps réel.

Cependant cette approche a des limites. En effet, sa précision dépend, d'une part, du nombre d'appariements utilisés dans le processus d'estimation, et d'autre part de la précision du suivi visuel qui dépend des conditions de travail (occultation, mouvement brusque et variations de luminosité). De ce fait, la vision a besoin d'être suppléée par d'autres capteurs afin d'améliorer la précision et la robustesse de la localisation.

Suite à l'étude réalisée sur différents systèmes multi-capteurs, nous avons dégagé une taxonomie qui se base sur la stratégie de combinaison. Nous avons recensé deux types de stratégies qui sont la fusion de données et la suppléance de données. L'étude comparative effectuée entre ces deux classes, nous a conduit à plusieurs constatations. D'un côté, nous avons remarqué que les approches de fusion se basent sur des modèles cinématiques qui ne prennent pas en compte certains types de mouvements (mouvement brusque). D'un autre côté, les approches de vision présentent souvent des performances satisfaisantes lorsque les conditions d'utilisation de la caméra sont favorables (mouvement lisse, éclairage contrôlé, etc.). Ainsi, l'utilisation de la suppléance qui consiste à remplacer la vision par d'autres capteurs tant qu'elle est dans l'incapacité de fournir une estimation correcte de la localisation, nous parait intéressante. Notre choix s'est porté donc sur ce type d'approche pour,

d'une part, proposer un système palliatif à la vision et d'autre part pouvoir tester sa faisabilité en environnement extérieur. Le fait d'utiliser une approche de suppléance, nous a permis de concevoir une solution "logicielle" pour se localiser en milieu extérieur. Ainsi, notre système de localisation est subdivisé en deux sous-systèmes : un sous-système de vision (principal) et un sous-système d'assistance à la localisation (palliatif).

La mise en œuvre de notre système multi-capteurs repose sur la résolution de plusieurs problématiques. La première concerne la procédure de calibration qui permet d'unifier les données issues des différents capteurs et de les réexprimer dans le même référentiel. Etant donné que notre capteur hybride comprend une caméra, une centrale inertielle et un récepteur GPS, nous avons proposé deux processus de calibration qui se basent sur les modèles obtenus du couplage inertiel/Caméra et GPS/Caméra. Nos approches ont l'avantage d'être simples à mettre en œuvre et ne requièrent pas de lourdes hypothèses. De plus, elles sont génériques et peuvent fonctionner dans différentes configurations. Par exemple l'approche de calibration Inertiel/Caméra peut être utilisée aussi bien pour déduire les orientations absolues que relatives.

La deuxième problématique abordée concerne le fonctionnement et les interactions de nos deux sous-systèmes. Pour cela, nous nous sommes inspirés d'une des caractéristiques de l'architecture logicielle utilisée pour le développement de notre système. En effet, le système ARCS se base sur un automate à états fini pour décrire le fonctionnement d'une application. Nous avons ainsi utilisé le concept d'automate pour modéliser notre système de localisation. Les états sont définis selon les traitements principaux effectués par le système. Le système passe d'un état à un autre selon les critères associés à chaque traitement, comme par exemple l'échec du suivi visuel fait passer le système de l'état associé à l'exploitation de la vision à l'état associé à l'assistance.

Par ailleurs, nous avons aussi mis au point une approche de réinitialisation automatique qui permet de retrouver les appariements des points 3D dans l'image courante après un échec du suivi visuel. Cette approche a prouvé son efficacité et sa précision.

Les contributions faites dans cette thèse ont permis de mettre au point un système qui a la faculté de s'adapter aux conditions extérieures en changeant son état interne. Nous avons décelé plusieurs problématiques auxquelles nous avons apporté des solutions que ce soit au niveau de la vision (approche d'initialisation et de réinitialisation), ou du système de suppléance. Les résultats obtenus à l'issue des différentes expérimentations nous ont démontré que l'utilisation d'une approche de suppléance pouvait être intéressante. En effet, l'ajout d'un sous-système de suppléance permet de pallier les problèmes de défaillance de la vision qui sont fréquents dans ce type de milieu.

A partir des résultats que nous avons obtenus, nous pouvons recenser plusieurs améliorations à apporter à notre système. Certes l'approche basée vision que nous avons mise au point est simple, elle fournit des résultats satisfaisants mais elle a des limites. Son problème principal réside dans le fait que la méthode ne peut suivre que les points identifiés lors de la phase d'initialisation. Une amélioration serait d'avoir la possibilité d'identifier de nouveaux points du modèle 3D et de les inclure en temps réel dans la phase de suivi.

De plus, il serait fort intéressant d'utiliser d'autres approches basées vision telle que les contours ou les segments. Dans cette perspective, nous avons pensé à utiliser une approche basée segments dans un environnement urbain et en utilisant des données extrait des SIG

(Systèmes d'information géographiques) ou bien des contours appariés avec des données extraits du Modèle Numérique de terrain (MNT) pour un système fonctionnant dans un environnement panoramique représenté par une chaine de montagne. Le lecteur intéressé peut entrevoir les pistes dégagées pour ces scénario dans l'annexe C et l'annexe B.

A cela s'ajoute le fait que l'utilisateur est toujours restreint à évoluer dans la partie modélisée de l'environnement. Pour pallier ce problème, il serait fort intéressant de se pencher sur des approches de type SLAM qui offrent la possibilité d'estimer simultanément les paramètres de la pose et de la structure de la scène. Ceci permettra de compléter les connaissances *a priori* dont nous disposons de l'environnement ou bien d'apporter cette connaissance pour des environnements inconnus.

Concernant le sous-système d'assistance, nous pouvons envisager des améliorations à plusieurs niveaux. Par exemple, au niveau de la prédiction, il serait intéressant de l'appliquer sur l'erreur de recalage étant donné que l'un des objectifs principaux dans les systèmes de réalité augmentée et le recalage réel/virtuel. En effet, au lieu d'apporter la correction au niveau de la localisation celle-ci sera appliquée directement sur le recalage afin de le faire coïncider avec ce que pourrait donner la vision.

Enfin, afin que le système de localisation fonctionne quelles que soient les conditions, il faut prendre en considération la défaillance des capteurs composant le système AL. Par exemple si le récepteur GPS se trouve dans une zone où il n'y a pas de couverture satellitaire, il fraudera trouver un moyen de prédire cette position. Pour cela, nous pourrions utiliser de nouvelles technologies telles que les réseaux de téléphonie mobile ou wifi afin de développer de nouvelles méthodes de localisation.

Bibliographie

[Ababsa, 2009] F. Ababsa (2009). Advanced 3d localization by fusing measurements from gps, inertial and vision sensors. Dans *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, pages 871 –875. [cité page(s). 66]

[Ababsa et al., 2008] F. Ababsa, J-Y. Didier, I.M. Zendjebil, et M. Mallem (2008). Markerless vision-based tracking of partially known 3d scenes for outdoor augmented reality applications. Dans *ISVC (1)*, pages 498–507. [cité page(s). 43]

[Ababsa et Mallem, 2004] F. Ababsa et M. Mallem (2004). Robust camera pose estimation using 2d fiducials tracking for real-time augmented reality systems. Dans *International conference on Virtual Reality continuum and its applications in industry*, pages 431–435, New York, NY, USA. ACM Press. [cité page(s). 35, 48]

[Ababsa et Mallem, 2006] F. Ababsa et M. Mallem (2006). Robust camera tracking for augmented reality combining orthogonal iteration and ransac algorithms. Dans *International Conference on Systems, Signals and Image Processing*. [cité page(s). 35]

[Ababsa et Mallem, 2007] F. Ababsa et M. Mallem (2007). Hybrid 3d camera pose estimation using particle filter sensor fusion. Dans *Advanced Robotics, the International Journal of the Robotics Society of Japan (RSJ)*, pages 165–181. [cité page(s). vi, 66, 71, 72, 83]

[Alves et al., 2004] J. Alves, J. Lobo, et J. Diaz (2004). Camera-inertial sensor modeling and alignment for visual navigation. *Journal of Robotic Systems*, 21 :6–12. [cité page(s). vi, 66, 91]

[Armstrong et Zisserman, 1995] M. Armstrong et A. Zisserman (1995). Robust object tracking. Dans *Asian Conference on Computer Vision*, pages 58–61. [cité page(s). 36]

[Aron et al., 2007] M. Aron, G. Simon, et M. Berger (2007). Use of inertial sensors to support video tracking : Research articles. *Comput. Animat. Virtual Worlds*, 18(1) :57–68. [cité page(s). vi, 77, 78, 80, 84, 85, 93, 94, 99, 143]

[Auer, 2000] T. Auer (Mars 2000). *Hybrid Tracking for Augmented Reality*. thèse de doctorat, Graz University. [cité page(s). 12]

[Azuma, 1993] R. Azuma (1993). Tracking requirements for augmented reality. *Commun. ACM*, 36(7) :50–51. [cité page(s). 65]

[Azuma, 1997] R.T. Azuma (1997). A survey of augmented reality. *Teleoperators and Virtual Environments*, pages 355 – 385. [cité page(s). v, 8, 18]

[Azuma et al., 1999] R. Azuma, J.W. Lee, B. Jiang, J. Park, S. You, et U.Neumann (1999). Tracking in unprepared environments for augmented reality systems. *Computers and Graphics*, 23(6) :787–793. [cité page(s). 66]

[Bailey et Durrant-Whyte., 2006] T. Bailey et H. Durrant-Whyte. (2006). Simultaneous localisation and mapping (slam) : Part ii - state of the art. *Robotics and Automation Magazine*. [cité page(s). 69]

[Baillot et al., 2003] Y. Baillot, S.J. Julier, D. Brown, et M.A. Livingston (2003). A tracker alignment framework for augmented reality. Dans *International Symposium on Mixed and Augmented Reality*, page 142. IEEE Computer Society. [cité page(s). 94, 102]

[Bay et al., 2008] H. Bay, A. Ess, T. Tuytelaars, et L. Van Goo (2008). Surf : Speeded up robust features. *Computer Vision and Image Understanding (CVIU)*, 110(3) :346–359. [cité page(s). 38, 51, 127, 128]

[Behzadan, 2008] A.H Behzadan (2008). *ARVISCOPE : Georeferenced Visualization of Dynamic Construction Processes in Three-Dimensional Outdoor Augmented Reality*. thèse de doctorat, Department of Civil and Environmental Engineering, University of Michigan. [cité page(s). 23, 24]

[Behzadan et Kama, 2005] Amir H. Behzadan et V.R. Kama (2005). Visualization of construction graphics in outdoor augmented reality. Dans *Winter simulation*, page 1914Ű1920, Orlando, Florida. Winter Simulation Conference. [cité page(s). 23, 24]

[Benhimane et Malis, 2004] S. Benhimane et E. Malis (2004). Real-time image-based tracking of planes using efficient second-order minimization. Dans *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 943–948. [cité page(s). 39]

[Bleser, 2009] G. Bleser (2009). *Towards Visual-Inertial SLAM for Mobile Augmented Reality*. thèse de doctorat, Technical University Kaiserslautern. [cité page(s). vi, vii, 43, 50, 69, 70, 83, 85, 97, 99, 102]

[Bleser et Stricker, 2008a] G. Bleser et D. Stricker (2008a). Advanced tracking through efficient image processing and visual-inertial sensor fusion. Dans *VR*, pages 137–144. [cité page(s). 66]

[Bleser et Stricker, 2008b] G. Bleser et D. Stricker (2008b). Using the marginalised particle filter for real-time visual-inertial sensor fusion. *Mixed and Augmented Reality, IEEE / ACM International Symposium on*, 0 :3–12. [cité page(s). 66, 83]

[Bleser et al., 2006] G. Bleser, H. Wuest, et D. Stricker (2006). Online camera pose estimation in partially known and dynamic scenes. Dans *International Symposium on Mixed and Augmented Reality*, pages 56–65, Santa Barbara, USA. [cité page(s). 39]

[Borenstein et Feng, 1996] J. Borenstein et L. Feng (1996). Gyrodometry : A new method for combining data from gyros and odom etry in mobile robots. Dans *International Conference onRobotics and Automation*, pages 423–428. [cité page(s). 77]

[Bérard, 1999] F. Bérard (1999). *Vision par ordinateur pour l'interaction homme-machine fortement couplée*. thèse de doctorat, Université Joseph Fourier, Grenoble. [cité page(s). 9]

[Caarls et al., 2008] J. Caarls, P. Jonker, et S. Persa (2008). Sensor fusion for augmented reality. [cité page(s). 66]

[Chai et al., 2002] L. Chai, W.A. Hoff, et T. Vincent (2002). Three-dimensional motion and structure estimation using inertial sensors and computer vision for augmented reality. *Presence : Teleoper. Virtual Environ.*, 11(5) :474–492. [cité page(s). 66]

[Claus et Fitzgibbon, 2004] D. Claus et A. Fitzgibbon (2004). Reliable fiducial detection in natural scenes. Dans *European Conference on Computer Vision*, volume 3024, pages 469–480, Prague, Czech Republic. Springer-Verlag. [cité page(s). 34]

[Comport et al., 2003] A.I. Comport, F. Marchand, et F. Chaumette (2003). A real-time tracker for markerless augmented reality. Dans *Symp. on Mixed and Augmented Reality*, pages 36–45, Tokyo, Japan. [cité page(s). 36]

[Comport et al., 2006] A.I. Comport, M. Pressigout, E. Marchand, et F. Chaumette (2006). Real-time markerless tracking for augmented reality : The virtual visual servoing framework. *Transactions on Visualization and Computer Graphics*, 12(4) :615–628. [cité page(s). v, 36, 37]

[Didier et al., 2006] J.Y. Didier, S. Otmane, et M. Mallem (2006). A component model for augmented/mixed reality applications with reconfigurable data-flow. Dans *International Conference on Virtual Reality*, pages 243–252, Laval (France). [cité page(s). 35]

[Didier et al., 2008] J-Y. Didier, F. Ababsa, et M. Mallem (2008). Hybrid camera pose estimation combining square fiducials localization technique and orthogonal iteration algorithm. *International Journal of Image and Graphics (IJIG)*, 8(1) :169–188. [cité page(s). 48, 49]

[Didier et al., 2009] J-Y. Didier, S. Otmane, et M. Mallem (2009). Arcs : Une architecture logicielle reconfigurable pour la conception des applications de réalité augmentée. *Technique et Science Informatiques (TSI), Innovations en Réalité Virtuelle et Réalité Augmentée*. Numéro spécial. [cité page(s). v, 9, 10, 131]

[Didier et al., 2005] J.-Y. Didier, D. Roussel, M. Mallem, S. Otmane, S. Naudet, Q.-C. Pham, S. Bourgeois, C. Mégard, C. Leroux, et A. Hocquard (2005). Amra : Augmented reality assistance in train maintenance tasks. Dans *Workshop on Industrial Augmented Reality*, Vienna (Austria). [cité page(s). 3]

[Drummond et Cipolla, 1999] T. Drummond et R. Cipolla (1999). Real-time tracking of complex structures for visual servoing. Dans *Workshop on Vision Algorithms*, pages 69–84. [cité page(s). 36]

[Faugeras et Toscani, 1987] O.D. Faugeras et G. Toscani (1987). Camera calibration for 3d computer vision. Dans *Proc. Int'l Workshop Industrial Applications of Machine Vision and Machine Intelligence*, pages 240–247. [cité page(s). 54]

[Fischler et Bolles, 1981] M.A. Fischler et R.C. Bolles (1981). Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6) :381–395. [cité page(s). 41, 52, 78, 128, 169]

[Foxlin et Naimark, 2003] E. Foxlin et L. Naimark (2003). Vis-tracker : A wearable vision-inertial self-tracker. Dans *Proceedings of the IEEE Virtual Reality 2003*, page 199, Washington, DC, USA. IEEE Computer Society. [cité page(s). 66]

[Fuchs et Moreau, 2003] P. Fuchs et G. Moreau (2003). *Le traité de la réalité virtuelle*, volume 1 : Fondements et interfaces comportementales. Presses de l'Ecole des Mines de Paris, deuxième édition édition. [cité page(s). 9]

[Gagnières, 2006] P. Gagnières (2006). Markerless tracking initialisé avec une cible. Master's thesis, Université d'Evry - Master RVSI. [cité page(s). 52]

[Gennery, 1992] D.B. Gennery (1992). Visual tracking of known three-dimensional objects. *Int. J. Comput. Vision*, 7(3) :243–270. [cité page(s). 36]

[Gleue et Dähne, 2001] T. Gleue et P. Dähne (2001). Design and implementation of a mobile device for outdoor augmented reality in the archeoguide project. Dans *Conference on Virtual reality, archeology, and cultural heritage*, pages 161–168, New York, NY, USA. ACM. [cité page(s). v, 22, 40]

[Greenberg et Fitchett, 2001] S. Greenberg et C. Fitchett (2001). Phidgets : easy development of physical interfaces through physical widgets. Dans *Symposium on User interface software and technology*, pages 209–218, New York, NY, USA. ACM. [cité page(s). 26, 144]

[Harris, 1993] C. Harris (1993). Tracking with rigid models. *Active vision*, pages 59–73. [cité page(s). 51, 54]

[Herbst et al., 2008] I. Herbst, A-K. Braun, R. McCall, et W. Broll (2008). Timewarp : Interactive time travel with a mobile mixed reality game. Dans *International Conference on Human Computer interaction with Mobile Devices and Services*, Amsterdam, The Netherlands. ACM. [cité page(s). 25]

[Hoff et al., 1996] W. Hoff, T. Lyon, et K. Nguyen (1996). Computer vision-based registration techniques for augmented reality. volume 2904, pages 538–548. Intelligent Robots and Computer Vision, In Intelligent Systems et Advanced Manufacturing. [cité page(s). 34]

[Hol et al., 2008] J.D. Hol, T.B. Schön, et F. Gustafsson (2008). Relative pose calibration of a spherical camera and an imu. Dans *International Symposium on Mixed and Augmented Reality*, Cambridge, United Kingdom. [cité page(s). 92, 93]

[Hol et al., 2006] J.D. Hol, T.B. Schon, F. Gustafsson, et P.J Slycke (2006). Sensor fusion for augmented reality. Dans *International Conference on Information Fusion*, pages 1–6, Florence. IEEE. [cité page(s). vi, 39, 66, 68]

[Hollerer et al., 1999] T. Hollerer, S. Feiner, T. Terauchi, G. Rashid, et D. Hallaway (1999). Exploring mars : Developing indoor and outdoor user interfaces to a mobile augmented reality system. *Computers and Graphics*, 23(6) :779–785. [cité page(s). v, 3, 19, 66]

[Holweg et Schneider, 2004] D. Holweg et O. Schneider (2004). Geist. mobile outdoor arinformationssystem for historical education with digital storytelling. [cité page(s). v, 22, 23]

[Horaud et Monga, 1995] R. Horaud et O. Monga (1995). *Vision par ordinateur*. Editions Hermès, deuxième édition revue et augmentée édition. [cité page(s). viii, 105, 173, 175]

[Horn, 1987] B.K.P. Horn (1987). Closed-forms solution of absolute orientation using unit quaternion. *Journal of the Optical Society of America*, 4(4) :629–642. [cité page(s). 92]

[Julier et al., 2000] S. Julier, Y. Baillot, M. Lanzagorta, D. Brown, et L. Rosenblum (2000). Bars : Battlefield augmented reality system. Dans *NATO Symposium on Information Processing Techniques for Military Systems*, Istanbul, Turkey. [cité page(s). v, 20, 66]

[Kalman, 1960] R.E. Kalman (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D) :35–45. [cité page(s). 66]

[Kato et Billinghurst, 1999] H. Kato et M. Billinghurst (1999). Marker tracking and hmd calibration for a video-based augmented reality conferencing system. Dans *International Workshop on Augmented Reality*, page 85, Washington, DC, USA. IEEE Computer Society. [cité page(s). v, 35]

[Kato et al., 2000] H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, et K. Tachibana (2000). Virtual object manipulation on a table-top ar environment. Dans *Proc. ISAR2000*, pages 111–119. [cité page(s). 35]

[King et al., 2005] G.R. King, W. Piekarski, et B.H Thomas (2005). Arvino - outdoor augmented reality visualisation of viticulture gis data. Dans *International Symposium on Mixed and Augmented Reality*. [cité page(s). 21]

[Klein et Drummond, 2003] G Klein et T. Drummond (2003). Robust visual tracking for non-instrumented augmented reality. Dans *International Symposium on Mixed and Augmented Reality*, page 113, Washington, DC, USA. IEEE Computer Society. [cité page(s). 36, 47, 73]

[Klinker, 1999] G. Klinker (1999). Augmented reality : A problem in need of many computer vision-based solutions. Dans *NATO Advanced Research Workshop, Confluence of Computer Vision and Computer Graphics*, Ljubljana, Slowenia. [cité page(s). 9]

[Koller et al., 1997] D. Koller, G. Klinker, E. Rose, D. Breen, R. Whitaker, et M. Tuceryan (1997). Real-time vision-based camera tracking for augmented reality applications. Dans Daniel Thalmann, editor, *Symposium on Virtual Reality Software and Technology*, New York, NY. ACM Press. [cité page(s). 34]

[Ladikos et al., 2008] A. Ladikos, S. Benhimane, et N. Navab (2008). High performance model-based object detection and tracking. Dans *Computer Vision and Computer Graphics. Theory and Applications*, volume 21 de *Communications in Computer and Information Science*. Springer. [cité page(s). 39]

[Lafarge et al., 2005] F. Lafarge, X. Descombes, J. Zerubia, et M. Pierrot-Deseilligny (2005). Modèle paramétrique pour la reconstruction automatique en 3d de zones urbaines denses à partir d'images satellitaires haute résolution. *Revue Française de Photogrammétrie et de Télédétection*, 180 :4–12. [cité page(s). 11]

[Lang et Pinz, 2005] P. Lang et A. Pinz (2005). Calibration of hybrid vision/inertial tracking systmes. Dans *Workshop on Integration of Vision and Inertial Sensors*, Barcelona. [cité page(s). vi, 92, 99]

[Lepetit et al., 2003] V. Lepetit, L. Vacchetti, et P. Fua D. Thalmann (2003). Fully automated and stable registration for augmented reality applications. Dans *International Symposium on Mixed and Augmented Reality*, page 93, Washington, DC, USA. IEEE Computer Society. [cité page(s). v, 38]

[Livingston et al., 2006] Mark A. Livingston, Dennis Brown, Simon J. Julier, et Greg S. Schmidt (2006). Military applications of augmented reality. NATO Human Factors and Medicine Panel Workshop on Virtual Media for Military Applications. [cité page(s). 20]

[L.Naimark et E.Foxlin, 2002] L.Naimark et E.Foxlin (2002). circular data matrix fiducial system and robust image processing for a wearable vision-interial self-tracker. Dans *International Symposium on Mixed and Augmented Reality*, Darmstadt,Germany. [cité page(s). 34]

[Lobo et Dias, 2007] J. Lobo et J. Dias (2007). Relative pose calibration between visual and inertial sensors. *Int. J. Rob. Res.*, 26(6) :561–575. [cité page(s). 91]

[Lowe, 1992] D.G. Lowe (1992). Robust model-based motion tracking through the integration of search and estimation. *International Journal of Computer Vision*, 8 :113–122. [cité page(s). 36]

[Lowe, 2004] D.G. Lowe (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2) :91–110. [cité page(s). 37, 38, 43, 50]

[Lu et al., 2000] C-P. Lu, G.D. Hager, et E. Mjolsness (2000). Fast and globally convergent pose estimation from video images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(6) :610–622. [cité page(s). 49, 168]

[Lucas et Kanade, 1981] B.D. Lucas et T. Kanade (1981). An iterative image registration technique with an application to stereo vision. Dans *IJCAI81*, pages 674–679. [cité page(s). 39, 53, 69]

[Maidi, 2007] M. Maidi (2007). *Suivi Hybride en présence d'Occultations pour la Réalité Augmentée*. thèse de doctorat, Université d'Évry Val d'Essonne, Évry. [cité page(s). 35, 85]

[Maidi et al., 2005] M. Maidi, F. Ababsa, et M. Mallem (2005). Vision-inertial system calibration for tracking in augmented reality. Dans *2nd International Conference on Informatics in Control, Automation and Robotics*, pages 156–162. [cité page(s). vi, 95, 99, 143]

[Maidi et al., 2009] M. Maidi, F. Ababsa, et M. Mallem (2009). Vision-inertial tracking system for robust fiducials registration in augmented reality. Dans *Symposium on Computational Intelligence for Multimedia Signal and Vision Processing*, Nashville (USA). [cité page(s). vi, vii, 79, 80, 84, 96, 99]

[Marchand et Bouthemy, 2001] E. Marchand et P. Bouthemy (2001). A 2d-3d model-based approach to real-time visual tracking. *IVC*, 19 :941–955. [cité page(s). 36]

[Markus et Dieter, 2007] S. Markus et S. Dieter (2007). Urban sketcher : Mixed reality on site for urban planning and architecture. Dans *6th International Symposium on Mixed and Augmented Reality*, pages 27–30, Nara, Japan. IEEE, ACM. [cité page(s). v, 24, 25]

[Marvie, 2004] J-E. Marvie (2004). *Visualisation Interactive d'Environnements Virtuels Complexes à travers des Réseaux et sur des Machines à Performances Variables*. thèse de doctorat, INSA, de Rennes, France. [cité page(s). 145]

[Mei et Rives, 2007] C. Mei et P. Rives (2007). Cartographie et localisation simultanée avec un capteur de vision. Dans *Journées Nationales de la Recherche en Robotique*. [cité page(s). 42]

[Milgram et Kishino, 1994] P. Milgram et F. Kishino (1994). A taxonomy of mixed reality visual displays. *IEICE Transactions on Information Systems*, E77-D(12). [cité page(s). v, 8, 16]

[Müller et al., 2006] P. Müller, P. Wonka, S. Haegler, A. Ulmer, et L. Van Gool (2006). Procedural modeling of buildings. 25(3) :614–623. [cité page(s). v, 11, 12]

[Mouragnon et al., 2006] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, et P. Sayd (2006). Monocular vision based slam for mobile robots. Dans *International Conference on Pattern Recognition*, pages 1027–1031, Washington, DC, USA. IEEE Computer Society. [cité page(s). 42]

[Nistér, 2003] D. Nistér (2003). Preemptive ransac for live structure and motion estimation. Dans *International Conference on Computer Vision*, page 199, Washington, DC, USA. IEEE Computer Society. [cité page(s). 42]

[Park et Martin, 1994] F. Park et B. Martin (1994). Robot sensor calibration : solving ax=xb on the euclidean group. *Transactions on Robotics and Automation*, 5 :717–721. [cité page(s). 94]

[Piekarski et Thomas, 2001] W. Piekarski et B.H. Thomas (2001). Tinmith-evo5 - an architecture for supporting mobile augmented reality environments. Dans *International Symposium on Augmented Reality*, pages 177–178, New York. [cité page(s). v, 20, 21]

[Piekarski et Thomas, 2002] W. Piekarski et B. Thomas (2002). Arquake : the outdoor augmented reality gaming system. *Communications of the ACM*, 45(1) :36–38. [cité page(s). v, 20, 21, 35]

[Pressigout et Marchand, 2005] M. Pressigout et E. Marchand (2005). Real-time planar structure tracking for visual servoing : a contour and texture approach. Dans *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'05*, volume 2, pages 1701–1706, Edmonton, Canada. [cité page(s). 42]

[Pressigout et Marchand, 2006] M. Pressigout et E. Marchand (2006). Real-time 3d model-based tracking : Combining edge and texture information. Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'06*, pages 2726–2731, Orlando, Florida. [cité page(s). vi, 42]

[Reitmayr et Drummond, 2006] Gerhard Reitmayr et Tom Drummond (2006). Going out : Robust model-based tracking for outdoor augmented reality. Dans *IEEE ISMAR*, Santa Barbara, California, USA. [cité page(s). v, vi, 36, 47, 59, 66, 73, 74, 84, 85, 99, 102]

[Reitmayr et Drummond, 2007] G. Reitmayr et T. Drummond (2007). Initialisation for visual tracking in urban environments. Dans *ISMAR*, Nara, Japan. [cité page(s). vi, 74, 84, 115, 125, 135, 143]

[Reitmayr et Schmalstieg, 2003] G. Reitmayr et D. Schmalstieg (2003). Collaborative augmented reality for outdoor navigation and information browsing. Technical Report TR-188-2-2003-28, Interactive Media Systems Group, Vienna University of Technology. [cité page(s). v, 21, 94]

[Rekimoto, 1998] J. Rekimoto (1998). Matrix : A realtime object identification and registration method for augmented reality. Dans *Asian Pacific Computer and Human Interaction*, page 63, Washington, DC, USA. IEEE Computer Society. [cité page(s). 35]

[Rekimoto et Nagao, 1995] J. Rekimoto et K. Nagao (1995). The world through the computer : Computer augmented interaction with real world environments. Dans *symposium User Interface Software and Technology*. [cité page(s). v, 17]

[Ribo et al., 2002] M. Ribo, P. Lang, H. Ganster, M. Brandner, C. Stock, et A. Pinz (2002). Hybrid tracking for outdoor augmented reality applications. *IEEE Comput. Graph. Appl.*, 22(6) :54–63. [cité page(s). 66]

[Rolland et al., 2000] J.P. Rolland, Y. Baillot, et A.A. Goon (2000). A survey of tracking technology for virtual environments. [cité page(s). 12]

[Schall et al., 2008] G. Schall, E. Mendez, E. Kruijff, E. Veas, S. Junghanns, B. Reitinger, et D. Schmalstieg (2008). Handheld augmented reality for underground infrastructure visualization. *Personal and Ubiquitous Computing*. [cité page(s). 24]

[Schall et al., 2007] G. Schall, E. Mendez, B. Reitinger, S. Junghanns, et D. Schmalstieg (2007). Handheld geospatial augmented reaily using urban 3d models. MSI Workshop, CHI. [cité page(s). v, 24]

[Schall et al., 2009] G Schall, D. Wagner, G. Reitmayr, E. Taichmann, M. Wieser, D. Schmalstieg, et B. Hofmann Wellenhof (2009). Global pose estimation using multi-sensor fusion for outdoor augmented reality. Dans *Int. Symposium on Mixed and Augmented Reality 2009*, Orlando, Florida, USA. [cité page(s). vi, 75, 76, 84, 85]

[Schmalstieg et al., 2002] D. Schmalstieg, A. Fuhrmann, G. Hesina, Z. Szalavari, L. M. Encarnaçäo, M. Gervautz, et et W. Purgathofer (2002). The studierstube augmented reality project. Dans *Teleoperators and Virtual Environments*, page 33Ű54. [cité page(s). 21]

[Schnadelbach et al., 2002] H. Schnadelbach, B. Kolvea, M. Flintham, M. Fraser, P. Chandler, M. Foster, S. Benford, C. Greenhalgh, S. Izadi, et T. Rodden (2002). The augurscope : A mixed reality interface for outdoors. Dans *ACM Conference on Computer-Human Interaction*, pages 9–16. ACM Press. [cité page(s). v, 22, 23]

[Simon et Berger, 2002] Gilles Simon et Marie-Odile Berger (2002). Pose estimation for planar structures. *IEEE Comput. Graph. Appl.*, 22(6) :46–53. [cité page(s). vi, 40, 41, 78]

[Simon et al., 2000] G. Simon, A. Fitzgibbon, et A. Zisserman (2000). Markerless tracking using planar structures in the scene. Dans *Proc. International Symposium on Augmented Reality*, pages 120–128. [cité page(s). 47, 143]

[State et al., 1996] A. State, M.A. Livingston, G. Hirota, W.F. Garrett, M.C. Whitton, H. Fuchs, et E.D. Pisano (1996). Techniques for augmented-reality systems : Realizing ultrasound-guided needle biopsies. pages 439–446, New Orleans. Proceedings of SIGGRAPH. [cité page(s). 3]

[Strauss et Carey, 1992] P.S. Strauss et R. Carey (1992). An object-oriented 3d graphics toolkit. *SIGGRAPH Comput. Graph.*, 26(2) :341–349. [cité page(s). 21]

[Stricker et Kettenbach, 2001] D. Stricker et T. Kettenbach (2001). Real-time and markerless vision-based tracking for outdoor augmented reality applications. Dans *ISAR '01 : Proceedings of the IEEE and ACM International Symposium on Augmented Reality (ISAR'01)*, page 189, Washington, DC, USA. IEEE Computer Society. [cité page(s). vi, 22, 40, 47]

[Sutherland, 1998] I.E. Sutherland (1998). A head-mounted three dimensional display. pages 295–302. [cité page(s). v, 7, 8]

[Technologies, 2007] Xsens Technologies (2007). Mti and mtx user manual and technical. Documentation. [cité page(s). 100]

[Thomas et al., 1999] B. Thomas, W. Piekarski, et B. Gunther (1999). Using augmented reality to visualize architecture designs in an outdoor environment. Dans *In Design Computing on the Net*, Sydney, NSW. [cité page(s). 20, 21]

[Vacchetti et al., 2004] L. Vacchetti, V. Lepetit, et P. Fua (2004). Combining edge and texture information for real-time accurate 3d camera tracking. Dans *ISMAR '04 : Proceedings of the Third IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'04)*, pages 48–57, Washington, DC, USA. IEEE Computer Society. [cité page(s). v, vi, 39, 41, 42]

[Vairon et al., 2007a] J. Vairon, L. Frauciel, F. Ababsa, J.Y. Didier, I.M. Zendjebil, R. Delmont, P. Guitton, M. Hachet, et B. Rodiere (2007a). Réalité augmentée en extérieur : Etat de l'art. Délivrable. [cité page(s). 12]

[Vairon et al., 2007b] J. Vairon, L. Frauciel, F. Ababsa, J.Y. Didier, I.M. Zendjebil, R. Delmont, P. Guitton, M. Hachet, et B. Rodiere (2007b). Raxenv : a new ar research project to serve outdoor environmental activities. 4e conférence du réseau Intuition. [cité page(s). 25]

[Vallino, 1998] J. Vallino (1998). *Interactive Augmented Reality*. thèse de doctorat, University of Rochester, New York. [cité page(s). 8]

[Viéville et al., 1993] T. Viéville, F. Romann, B. Hotz, Hervé Mathieu, Michel Buffa, Luc Robert, P. Facao, Olivier Faugeras, et J.T. Audren (1993). Autonomous navigation of a mobile robot using inertial and visual cues. Dans *International Conference on Intelligent Robots and Systems*. [cité page(s). 65]

[Williams, 1997] C. Williams (1997). Prediction with gaussian processes : From linear regression to linear prediction and beyond. Technical report, Neural Computing Research Grou. [cité page(s). 74, 125, 126]

[Wuest et al., 2005] H. Wuest, F. Vial, et D. Stricker (2005). Adaptive line tracking with multiple hypotheses for augmented reality. Dans *ISMAR '05 : Proceedings of the Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 62–69, Washington, DC, USA. IEEE Computer Society. [cité page(s). 37]

[You et al., 1999] S. You, U. Neumann, et R. Azuma (1999). Orientation tracking for outdoor augmented reality registration. *IEEE Computer Graphics and Applications*, 19(6) :36–42. [cité page(s). vi, 66, 67, 68, 90, 91, 99]

[Zendjebil et al., 2008a] I.M Zendjebil, F. Ababsa, J. Didier, J. Vairon, L. Frauciel, M Hachet, P. Guitton, et R. Delmont (2008a). Outdoor augmented reality : State of the art and issues. Dans *Virtual Reality International Conference*, pages 177–187. [cité page(s). 4]

[Zendjebil et al., 2009] I.M. Zendjebil, F. Ababsa, J-Y. Didier, E. Lalagüe, F. Decle, R. Delmont, L. Frauciel, et J. Vairon (2009). Réalité augmentée en extérieur : Etat de l'art. *Technique et Science Informatiques, Réalité Virtuelle - Réalité Augmentée*, 28(6-7/2009) :857–890. Numéro spécial. [cité page(s). 4, 31]

[Zendjebil et al., 2008b] I.M. Zendjebil, F. Ababsa, J-Y. Didier, et M. Mallem (2008b). Hybrid localization system for mobile outdoor augmented reality applications. Dans *International Workshops on Image Processing Theory, Tools and Applications*, Sousse, Tunisia. IEEE. [cité page(s). 5]

[Zendjebil et al., 2010] I.M. Zendjebil, F. Ababsa, J-Y. Didier, et M. Mallem (2010). A gps-imu-camera modelization and calibration for 3d localization dedicated to outdoor mobile applications. Dans *International Conference On Control, Automation and system*. a paraitre. [cité page(s). 5]

[Zendjebil et al., 2008c] I.M. Zendjebil, F. Ababsa, J-Y. Didier, et M.Mallem (2008c). Toward an inertial/vision sensor calibration for outdoor augmented reality applications. Dans Springer, editor, *International Workshop on Mobile Geospatial Augmented Reality (REGARD)*, Québec, Canada. Springer, Springer. [cité page(s). 5]

[Zendjebil et al., 2008d] I. M. Zendjebil, F. Ababsa, J-Y. Didier, et M. Mallem (2008d). On the hybrid aid-localization for outdoor augmented reality applications. Dans *Symposium on Virtual reality software and technology*, pages 249–250, New York, NY, USA. ACM. [cité page(s). 5]

[Zollner et al., 2008] M. Zollner, A. Pagani, Y. Pastarmov, H. Wuest, et D. Stricker (2008). Reality filtering : A visual time machine in augmented reality. Dans *VAST*, pages 71–77. European Association for Computer Graphics (Eurographics). [cité page(s). 39, 50]

Annexe A

Outils Mathématiques

Dans cette annexe, nous allons présenter quelques concept et outils mathématiques utilisés en vision par ordinateur.

A.1 Géométrie de la caméra

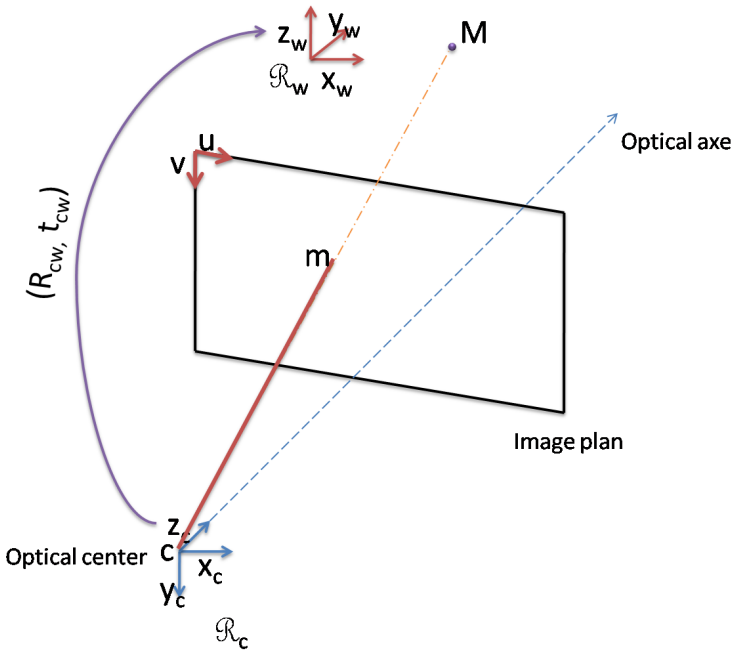


FIG. A.1: Modèle de projection perspective : modèle sténopé

Le modèle de caméra présenté dans la figure A.1 est le modèle sténopé dit modèle de projection perspective. La caméra est représentée par un plan rétinien, en l'occurrence le plan image ($\mathscr{P}$), et un centre optique qui correspond au centre de projection $\mathscr{C}$ (ce point n'appartient pas au plan image).
La projection orthogonale de $\mathscr{C}$ sur le plan $\mathscr{P}$, le point c, est appelée point principal. La droite ($\mathscr{C}c$) est nommée axe optique. La distance $f = \mathscr{C}c$ est la distance focale.

Un point M se projette en un point m sur le plan $\mathscr{P}$ selon une projection perspective du centre $\mathscr{C}$. Autrement dit le point m correspond à l'intersection de la droite $(\mathscr{C}M)$ avec le plan image $\mathscr{P}$. La projection monde/image se décompose donc en :

- Une projection du point M dans le repère associé à la caméra selon les paramètres extrinsèques de la caméra.
- Le point résultant de la transformation monde-caméra se projette sur le plan image en fonction des paramètres intrinsèques de la caméra.

A.1.1 Paramètres extrinsèques

Soit le point M de coordonnées $(\mathrm{X},\mathrm{Y},\mathrm{Z})^T$ exprimées dans le repère monde. Ses coordonnées dans le repère caméra $(\mathrm{X}_c,\mathrm{Y}_c,\mathrm{Z}_c)^T$ sont obtenues à partir de la relation suivante :

$$\begin{pmatrix}\mathrm{X}_c\\ \mathrm{Y}_c\\ \mathrm{Z}_c\end{pmatrix} = R\begin{pmatrix}\mathrm{X}\\ \mathrm{Y}\\ \mathrm{Z}\end{pmatrix} + t = [R|t]\begin{pmatrix}\mathrm{X}\\ \mathrm{Y}\\ \mathrm{Z}\\ 1\end{pmatrix} \tag{A.1}$$

Où $(\mathrm{X},\mathrm{Y},\mathrm{Z},1)^T$ sont les coordonnées homogènes du point P dans le repère monde. R et t représentent le déplacement rigide entre les deux repères, R étant la matrice de rotation et t le vecteur de translation. Ces paramètres définissent la position et l'orientation de la caméra par rapport au repère monde. Ils représentent les paramètres extrinsèques de la caméra appelés pose de la caméra.

A.1.2 Paramètres intrinsèques

La projection du point M, de coordonnées $(\mathrm{X}_c,\mathrm{Y}_c,\mathrm{Z}_c)^T$ définies dans le repère caméra, est le point m de coordonnées $(x_c,y_c,z_c)^T$ définies dans le repère caméra, telle que :

$$\begin{pmatrix}x_c\\ y_c\\ z_c\end{pmatrix} = f\begin{pmatrix}\mathrm{X}_c\\ \mathrm{Y}_c\\ \mathrm{Z}_c\\ 1\end{pmatrix} \tag{A.2}$$

Dans le repère image $2D$, le point m est exprimé en coordonnées pixel $(u,v)^T$ obtenues grâce à la relation suivante :

$$\begin{pmatrix}u\\ v\end{pmatrix} = \begin{pmatrix}k_u & -k_u\cos\theta & u_0\\ 0 & k_v\sin\theta & v_0\end{pmatrix}\begin{pmatrix}x_c\\ y_c\\ 1\end{pmatrix} \tag{A.3}$$

Où k_u et k_v représentent les distances focales exprimées en pixels suivant l'axe u et l'axe v respectivement ; $(u_0,v_0)^T$ sont les coordonnées pixel du point centre c ; θ est l'angle entre les deux axes (u et v) dit aussi angle de distorsion. Les paramètres k_u, k_v,f, u_0,v_0, θ sont les paramètres intrinsèques de la caméra.

De l'équation (A.2) et de l'équation (A.3), nous obtenons la relation suivante :

$$\begin{pmatrix}u\\ v\\ 1\end{pmatrix} = \begin{pmatrix}\alpha_u & 0 & u_0\\ 0 & \alpha_v & v_0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}\mathrm{X}_c\\ \mathrm{Y}_c\\ \mathrm{Z}_c\\ 1\end{pmatrix} \tag{A.4}$$

Avec $\alpha_u = fK_u$ et $\alpha_v = fk_v$ et θ généralement de l'ordre de $\frac{\pi}{2}$. Posons K la matrice des paramètres intrinsèques. Au final, la relation qui relie le point M de coordonnées homogènes $(\mathrm{X},\mathrm{Y},\mathrm{Z},1)^T$ dans le repère monde et le point m de coordonnées pixel homogènes $(u,v,1)^T$ est donnée par :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = K[R|t] \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ 1 \end{pmatrix} \tag{A.5}$$

La matrice $K[R|t]$ est appelée matrice de projection perspective. De nombreuses méthodes ont été développées par la communauté de la vision par ordinateur pour le calcul de ces paramètres (méthodes de calibration de caméra). Les paramètres intrinsèques sont généralement supposés connus dans le problème d'estimation de pose.

A.2 Faugeras-Toscani : calibration de caméra

Soit M un point 3D dont les coordonnées homogènes $(X,Y,Z,1)^T$ sont définit dans le repère monde. Ce point se projette sur le plan image en le point m de coordonnées homogènes $(u,v,1)^T$. Soit P la matrice de projection, telle que :

$$\begin{pmatrix} su \\ sv \\ s \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{A.6}$$

Avec s un facteur d'échelle. Cette équation se réécrit :

$$u = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}} v = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}} \tag{A.7}$$

A partir d'un ensemble de points (u_i, v_i) et un ensemble de correspondant de points 3D (X_i, Y_i, Z_i), l'approche estime la matrice de projection P (matrice 3x4) en minimisant le critère suivant

$$Q = |BX9 + CX3|^2 \tag{A.8}$$

Où

$$B = \begin{pmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -u_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -v_i \end{pmatrix} \tag{A.9}$$

$$B = \begin{pmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -u_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -v_i \end{pmatrix} \tag{A.10}$$

$$C = \begin{pmatrix} -u_iX_i & -u_iY_i & -u_iZ_i \\ -v_iX_i - v_iY_i - v_iZ_i & & \end{pmatrix} \tag{A.11}$$

$$X9 = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} & p_{21} & p_{22} & p_{23} & p_{24} & p_{34} \end{pmatrix}^T \tag{A.12}$$

$$X3 = \begin{pmatrix} p_{31} & p_{32} & p_{33} \end{pmatrix}^T \tag{A.13}$$

Une fois la matrice de projection estimée, les paramètres intrinsèques et extrinsèques sont extraits telque :

$$\begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{pmatrix} = \begin{pmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (A.14)$$

En posant :

$$P_i = [p_{i1} p_{i2} p_{i3}] \quad (A.15)$$

et :

$$R_i = [r_{i1} r_{i2} r_{i3}] \quad (A.16)$$

Les paramètres sont obtenus à partir des relations suivantes :

$$R_3 = P_3 \quad (A.17)$$

$$u_0 = P1.P_3 \quad (A.18)$$

$$v_0 = M_2.M_3 \quad (A.19)$$

$$\alpha_u = -|P_1 \hat{} P_3| \quad (A.20)$$

$$\alpha_v = |P_2 \hat{} P_3| \quad (A.21)$$

$$R1 = 1/\alpha_u * (P_1 - u_0 * p_3) \quad (A.22)$$

$$R_2 = 1/\alpha_v * (P_2 - v_0 * P_3) \quad (A.23)$$

$$t_x = 1/\alpha_u * (p_{14} - u_0 * p_{34}) \quad (A.24)$$

$$t_y = 1/\alpha_v * (p_{24} - v_0 * p_{34}) \quad (A.25)$$

$$t_z = p_{34} \quad (A.26)$$

A.3 Itération orthogonale

L'algorithme d'itération orthogonale [Lu et al., 2000] permet de déterminer dynamiquement les paramètres externes de la caméra en utilisant les mises en correspondances 2D/3D établies entre les points du modèle 3D et les points extraits de l'image courante. L'algorithme d'itération orthogonale calcule d'abord le vecteur d'erreur de colinéarité de l'espace d'objet :

$$e_i = (I - \hat{V}_i)(RP_i + t) \quad (A.27)$$

Où $\hat{V}_i$ est la ligne de vue observée à travers la matrice de projection définie par :

$$\hat{V}_i = \frac{\hat{p}_i \hat{p}_i^T}{\hat{p}_i \hat{p}_i} \quad (A.28)$$

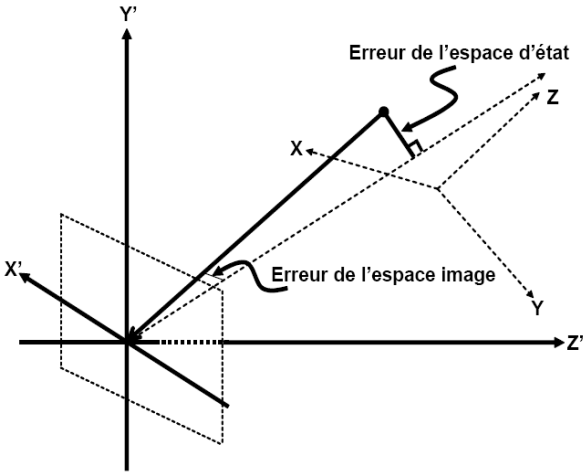


FIG. A.2: Schéma représentant le principe de l'algorithme Itération orthogonale

Avec $\hat{p}_i$ est la projection de P_i dans l'image. Les paramètres de la pose sont calculés avec une minimisation de l'erreur quadratique définie par :

$$E(R,t) = \sum_{i=1}^{n} \|e_i\|^2 = \sum_{i=1}^{n} \|(I - \hat{V}_i)(RP_i + t)\|^2 \tag{A.29}$$

L'algorithme d'itération orthogonale converge à un optimum pour un quelconque ensemble dz points observés et n'importe quel point de départ. Cependant, afin d'assurer la convergence de l'algorithme de calcul de la pose en un temps minimal, une bonne initialisation de paramètres de pose est requise.

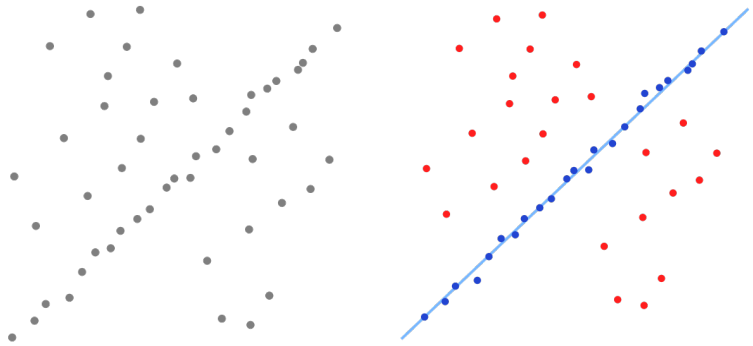
A.4 Algorithme de RANSAC

RANSAC (RANdom SAmple Consensus) [Fischler et Bolles, 1981] est une méthode itérative pour estimer les paramètres d'un modèle mathématique à partir d'un ensemble de données observées qui contient des valeurs aberrantes ("outliers"). Il s'agit d'un algorithme non-déterministe dans le sens où il produit un résultat correct seulement avec une certaine probabilité, cette probabilité augmentant à mesure que le nombre d'itérations est grand.

L'hypothèse de base est que les données sont constituées de *inliers*, à savoir les données dont la distribution peut être expliquées par un ensemble de paramètres d'un modèle, et de "outliers" qui sont des données qui ne correspondent pas au modèle choisi. De plus, les données peuvent être soumises au bruit. Les valeurs aberrantes (outliers) peuvent venir, par exemple, des valeurs extrêmes du bruit, de mesures erronées ou d'hypothèses fausses quant à l'interprétation des données. RANSAC suppose également que, étant donné un ensemble (généralement petit) d'*inliers*, il existe une procédure qui permet d'estimer les paramètres

d'un modèle de telle façon à expliquer de manière optimale ces données.

Un exemple simple est l'ajustement d'une ligne 2D à une série d'observations. On suppose que cet ensemble contient à la fois des *inliers*, c'est-à-dire, les points qui peuvent être approximativement ajustés à une ligne, et les outliers (valeurs aberrantes), les points qui sont éloignés de ce modèle de ligne. Un simple traitement par une méthode des moindres carrés donnera une ligne qui est mal ajustée aux *inliers*. En effet, la droite s'ajustera de manière optimale à tous les points, y compris les valeurs aberrantes (outliers). RANSAC, par contre, peut générer un modèle qui ne tiendra compte que des *inliers*, à condition que la probabilité de choisir seulement que des *inliers* dans le choix des données soit suffisamment élevée. Cependant, il n'y a aucune garantie d'obtenir cette situation, et il existe un certain nombre de paramètres de l'algorithme qui doivent être soigneusement choisis pour maintenir ce niveau de probabilité suffisamment élevé.



(a) (b)

FIG. A.3: (a) Ensemble de données (b) Ajustement de modèle de ligne avec détermination de ligne droite

Les données d'entrée de l'algorithme RANSAC sont un ensemble de valeurs des données observées, un modèle paramétré qui peut expliquer ou être ajusté aux observations, et des paramètres d'intervalle de confiance. RANSAC atteint son objectif en sélectionnant itérativement un sous-ensemble aléatoire des données d'origine. Ces données sont d'hypothétiques *inliers* et cette hypothèse est ensuite testée comme suit :

1. Un modèle est ajusté aux *inliers* hypothétiques, c'est-à-dire que tous les paramètres libres du modèle sont estimés à partir de cet ensemble de données.
2. Toutes les autres données sont ensuite testées sur le modèle précédemment estimé. Si un point correspond bien au modèle estimé alors il est considéré comme un *inlier* candidat.
3. Le modèle estimé est considéré comme correct si suffisamment de points ont été classés comme *inliers* candidats.
4. Le modèle est re-estimé à partir de cet ensemble des *inliers* candidats.
5. Finalement, le modèle est évalué par une estimation de l'erreur des *inliers* par rapport au modèle.

Cette procédure est répétée un nombre fixe de fois, chaque fois produisant soit un modèle qui est rejeté parce que trop peu de points sont classés comme *inliers*, soit un modèle réajusté et une mesure d'erreur correspondante. Dans ce dernier cas, on conserve le modèle réévalué si son erreur est plus faible que le modèle précédent.

A.5 Homographie

L'homographie est une transformation linéaire entre deux plans projectifs. Ceci se traduit par le fait qu'un ensemble de points 2D projectifs p_i définit un plan π_1 peuvent être projetées sur un deuxième plan π_2 en des points p_j données par :

$$p_j = H p_i \tag{A.30}$$

avec :

$$H = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \tag{A.31}$$

L'homographie est :

- de 8 degrés de liberté, car $H \equiv \alpha H$;
- de rang 3 (inversible) ;
- pour deux relations $H_{01} : \pi_0 \rightarrow \pi_1$ et $H_{12} : \pi_1 \rightarrow \pi_2$, on peut construire $H_{02} = H_{01} H_{12} : \pi_0 \rightarrow \pi_2$.

Sachant que $p_i = (x_i, y_i)^T$ et $p_j = (x_j, y_j)^T$, nous obtenons la relation suivante :

$$(h_{31} x_i + h32 y_i + h_{33}) x_j = h_{11} x_i + h12 y_i + h_{13} \tag{A.32}$$

$$(h_{31} x_i + h32 y_i + h_{33}) y_j = h_{21} x_i + h22 y_i + h_{23} \tag{A.33}$$

Ce qui nous donne :

$$h_{11} x_i + h12 y_i + h_{13} - h_{31} x_i x_j - h32 y_i x_j - h_{33} x_{i+1} = 0 \tag{A.34}$$

$$h_{21} x_i + h22 y_i + h_{23} - h_{31} x_i y_j - h32 y_i y_j - h_{33} y_{i+1} = 0 \tag{A.35}$$

Mise sous forme matricielle, cela revient à résoudre l'équation $Ax = b$ en posant $h = (h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33})^T$, nous abtenons le système quit suit :

$$\begin{pmatrix} x_i & y_i & 1 & 0 & 0 & 0 & -x_i x_j & -y_i x_j \\ 0 & 0 & 0 & x_i & y_i & 1 & -x_i x_j & -y_i x_j \end{pmatrix} h = \begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} \tag{A.36}$$

En utilisant la pseudo-inverse, l'homographie peut être retrouvé de manière analytique.

Annexe B

Estimation de pose basée segments

Parmi les approches basées vision que nous avons explorés, les méthodes basées contours constituent un potentiel. En effet, ce type d'information est très présent dans des environnements extérieurs essentiellement dans les environnements urbains où nous trouvons principalement des segments (empreintes de bâtiments, routes, bâtiments, etc.). Ce type de données est généralement répertorié dans des bases de données géo-référencées.

L'approche que nous allons décrire permet d'estimer la pose de la caméra à partir de données naturelles extraites de la scène et mises en correspondance avec des données d'un modèle 3D représentant une connaissance partielle de l'environnement. Cette approche se distingue par sa robustesse face aux semi-occultations et aux variations de luminosité. L'intérêt de l'approche que nous allons décrire et que nous n'avons pas besoin d'une connaissance exacte de l'environnement. En effet, il suffit de connaitre les coordonnées d'un point 3D du segment et du vecteur directeur pour retrouver les paramètres de la pose. Voici un bref descriptif de l'approche décrite dans le livre de [Horaud et Monga, 1995].

B.1 Vue globale

L'idée est de chercher les correspondants des segments 3D dans l'image et de calculer la pose à partir de ces correspondances. L'appariement consiste à chercher pour chaque point du segment projeté avec un point de vue prédéfini, le maxima du gradient le long de la normal au point. Pour le calcul de pose, il existe plusieurs approches. Nous optons pour l'approche décrite dans [Horaud et Monga, 1995] qui a comme avantage de ne pas requérir un appariement point par point du contour, au contraire il suffit uniquement de retrouver une partie du segment projeté comme correspondant du segment 3D. Ceci permet de gérer les occultations d'une part et d'autre part, ceci peut représenter une solution face à l'imprécision du modèle 3D utilisé.

La figure B.1 illustre le flux de données de l'approche que nous allons décrire. La méthode est décomposée en deux phases : une phase d'appariement et une phase d'estimation de pose. Dans ce qui va suivre nous allons décrire chacune de ces étapes.

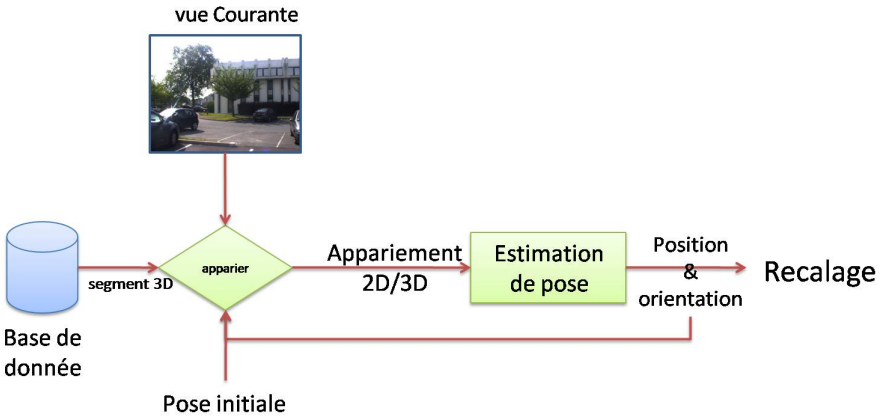


FIG. B.1: Principe général de fonctionnement en utilisant une approche basée segment.

Le principe consiste à projeter l'ensemble des segments visibles d'un modèle filaire représentant l'environnement dans le repère de l'image courante avec un point de vue prédit (soit une pose prédéfinie ou la pose précédente) puis de chercher le segment correspondant dans cette image.

Nous définissons par $M_i^1 = (X_i^1, Y_i^1, Z_i^1)$ et $M_i^2 = (X_i^2, Y_i^2, Z_i^2)$ les deux points d'extrémité du segment i. Le segment projeté est échantillonné en un ensemble de n points. Le nombre d'échantillon par segment est fixé au préalable. A chaque point échantillonné, l'approche détermine la direction de la normal. Etant donné que nous utilisant des segments, la direction de la normale n'est autre que la perpendiculaire au segment calculé en chaque échantillon. Puis, pour chaque point, nous cherchons le long de cette direction de normale, le point qui a correspond au maxima du gradient qui représente un point contour. La figure B.2 présente une illustration du principe de l'appariement.

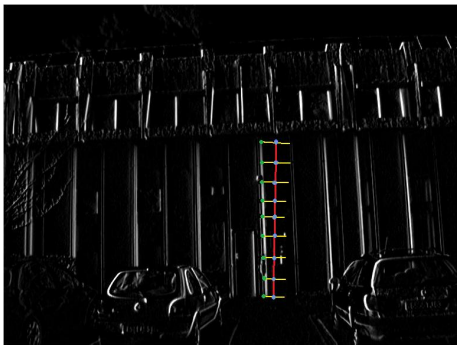


FIG. B.2: Schéma descriptif de la phase d'appariement 2D/3D en utilisant les segments

B.2 Appariement 2D/3D basé segments

Cependant, il se peut que ce point contour ne soit pas le véritable correspondant. Ceci peut être le cas, lorsque nous avons un ensemble de contours voisins. Pour pallier à ceci et éliminer ces faux appariements, nous utilisons un RANSAC (cf. annexe A). Le RANSAC utilisé suppose que les correspondants obtenu d'un segment doivent former un segment de droite. De ce fait, nous cherchons la meilleure équation du segment de droite, telle que l'ensemble des points appariés appartient à cette droite. Ceci nous permet au final de ne garder que les bons appariements formants une droite. Pour le reste de la méthode, nous n'aurons besoin que de deux points du segment apparié. Ainsi, nous choisissons les extrémités du segment, noté $m_1^i = (u_1^i, v_1^i)$ et $m_2^i = (u_2^i, v_2^i)$, obtenues après appariements.

B.3 Estimation de la pose basée segments

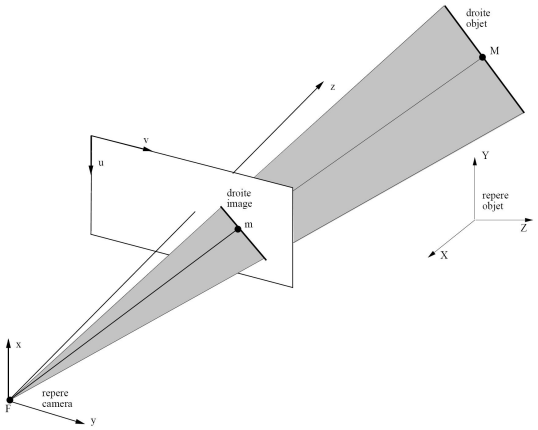


FIG. B.3: Schéma descriptif [Horaud et Monga, 1995]

Une fois les corresponds obtenus, la pose peut être estimée. Celle-ci peut être estimée en calculant le mouvement qui permet d'aligner le contour projeté avec le contour image. Ce mouvement permet de mettre à jour la pose prédite. Cependant, ce modèle de solution requière une bonne connaissance a priori de l'environnement. La solution que nous allons définir maintenant se base sur le calcul des paramètres de la pose de manière à minimiser une erreur. Cette erreur se base sur la définition de la droite 3D à partir de la normale au plan.

A partir du modèle de projection et sachant que (u_0, v_0) sont les coordonnées pixel du point centre et (α_u, α_v) les distances focales en pixel, un point M de coordonnées (X, Y, Z) se projette sur le plan de la caméra en le point M^c de coordonnées (X^c, Y^c) telle que :

$$\begin{pmatrix} X^c \\ Y^c \end{pmatrix} = \begin{pmatrix} \frac{X}{Z} \\ \frac{Y}{Z} \end{pmatrix} \tag{B.1}$$

Ce point correspond au point image m est défini par ces coordonnées (u, v) dans le repère associé à l'image telle que :

$$\begin{pmatrix} X^c \\ Y^c \end{pmatrix} = \begin{pmatrix} \frac{u-u_0}{\alpha_u} \\ \frac{v-v_0}{\alpha_v} \end{pmatrix} \tag{B.2}$$

Ce point m appartient à une droite définie dans l'image définie dans le repère caméra par l'équation :

$$aX^c + bY^c + c = 0 \Leftrightarrow aX + bY + cZ = 0 \Leftrightarrow \overrightarrow{n}.\overrightarrow{OM} = 0$$

Cette équation représente le plan formé par le centre de projection O, et la droite comprenant M. $\overrightarrow{n}$ représente la normal au plan. Sachant que :

$$M^c = RM + t \tag{B.3}$$

Telle que (R,t) sont les paramètres de pose de la caméra. La droite définie dans le repère caméra appartient aussi à ce plan là. De ce fait, ceci nous donne :

$$\overrightarrow{n}.\overrightarrow{OM^c} = 0 \tag{B.4}$$

En remplaçant M^c par leur expressions dans les équations B.3, nous obtenons :

$$n.(RM + t) = 0 \tag{B.5}$$

Ce qui nous donne au final, une contrainte par segment de droite. L'estimation des paramètres de la pose dépend de la manière de les représenter. Si par exemple nous optons pour la représentation en angle d'Euler de la rotation, ceci nous donne 3 paramètres pour la rotation et 3 paramètres pour la translation. De ce fait, en utilisant 3 segments de droite au minimum, la pose peut être estimée. Cependant, ce formalisme présente une singularité. De ce fait, nous optons pour la représentation en quaternions qui est plus robuste.

A partir de N correspondances de segments 3D et segments 2D, la pose est obtenu en minimisant la fonction de coût qui décrit l'ensemble des contraintes obtenues à partir de chaque couple de segments 3D/2D, ce qui nous donne :

$$f(R,t) = \sum_{i=1}^{N} (n_i.(Rm_i + t)) \tag{B.6}$$

En représentant notre rotation R par un quaternion noté r telle que $R = W(r)^t Q(r)$, les contraintes définies dans les équations B.5 s'écrivent de la manière suivant :

$$n.(Rm+t) = n^T(W(r)^T Q(r)m + t) = n^T W(r)^T Q(r)m + n^T t = r^T Q(n)^T W(m)r + n^T t \tag{B.7}$$

En posant $B = Q(n)^T W(m)$, la fonction de cout décrite dans B.6, s'écrit de la manière suivante :

$$f(R,t) = \sum_{i=1}^{N} (r^T B_i r + n_i^T t)^2 + \lambda(r^T r - 1) \tag{B.8}$$

L'ajout de la contrainte $(r^T r - 1)$ correspond au fait que la norme du quaternion doit être égale à 1. Cependant, nous pouvons estimer seulement trois paramètre du quaternion r (à savoir r_x,r_y et r_z) et déduire la quatrième composante qui permet d'assurer cette contrainte. Les matrices W et Q font parti des caractéristiques des quaternions.
Les paramètres de la pose $w = (r_x, r_y, r_z, t_x, t_y, t_z)$ sont obtenue en minimisant la fonction de cout de l'équation B.8. Un levenberg-Maquardt ou un Newton-Gauss peuvent être utilisés. Dans notre cas nous allons utiliser la méthode de levenberg-Maquardt.

Annexe C

Estimation de pose basée crêtes de montagnes (contours)

Un des scénarios mis au point dans le cadre du projet RAXENV consiste à déployé la plate-forme dans un site panoramique représentant une chaine de montagne. En gardant le même système multi-capteurs décrit dans cette thèse, nous apportons une modification au niveau du sous-système de vision. En effet, au lieu d'utiliser une méthode basée points, nous nous orientons vers une approche utilisant des contours.

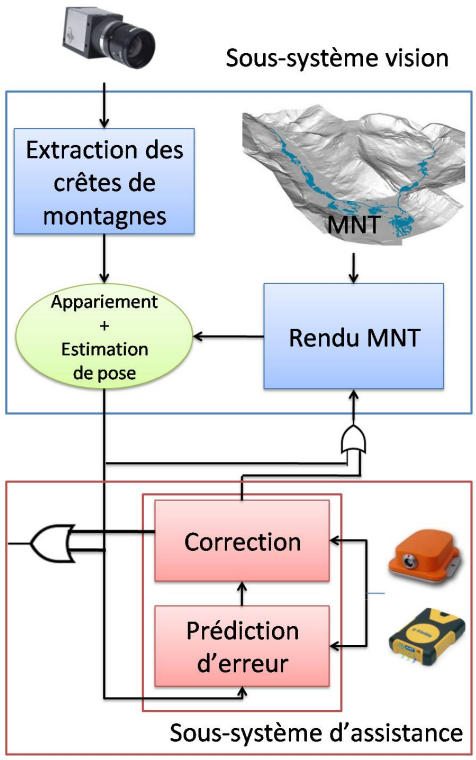
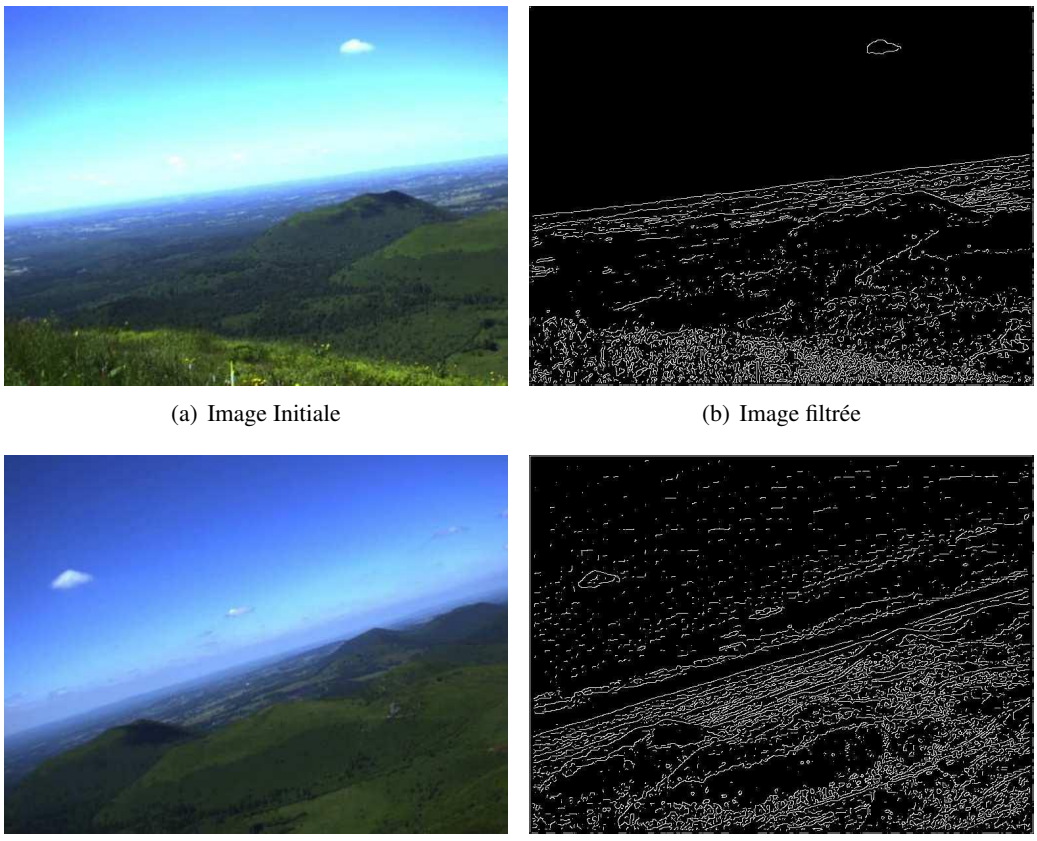


FIG. C.1: Système multi-capteur pour la localisation en environnement panoramique

La figure C.1 illustre l'adaptation du système multi-capteurs au scénario panormaique.

Cette version utilise un modèle numérique de terrain (MNT) qui représente l'élévation du sol. Le principe consiste à calculer les paramètres de la pose en alignant les contours obtenus avec le rendu du MNT et ceux de l'image. Cependant, pour cette approche nous optons pour la seconde classe de méthode où les contours sont extraits de manière explicite (cf. 2 page 33). Ce choix se justifie par le fait que la scène est très texturées et engendre beaucoup de mauvais appariements. L'extraction explicite de ces crêtes de montagnes permet de réduire considérablement ces mauvais appariements. Toutefois, pour l'extraction des crêtes, nous n'allons pas utiliser un détecteur de contours conventionnel en utilisant des filtres pour calculer la norme du gradient. En effet, ce type de filtre ne fournit pas le résultat souhaité sur ce type d'environnement (cf. fig. C.2). Pour cela, nous avons mis au point une approche qui se base sur une segmentation couleur pour extraire les contours qui nous intéresse.



(a) Image Initiale (b) Image filtrée

(c) Image Initiale (d) Image filtrée

FIG. C.2: Exemples de résultats d'extraction de contours avec le filtre de Deriche sur les images panormiques

C.1 Extraction des crêtes de montagnes

La segmentation consiste à regrouper les pixels d'une image en régions homogènes suivant des critères prédéfinis. Il existe plusieurs approches de segmentation qui sont basées sur des régions, sur des contours sur des classifications ou seuillage ou bien une combinaison de plusieurs critères. Pour extraire les sommets des crêtes de montagnes, nous proposons la chaine de traitements suivante :

1. Segmentation de l'image en région en utilisant l'espace de couleur HSV (Hue, Saturation, Value).
2. Filtrage de l'image partitionnée en région pour absorber les petites régions parasites.
3. Extraction des contours des régions.

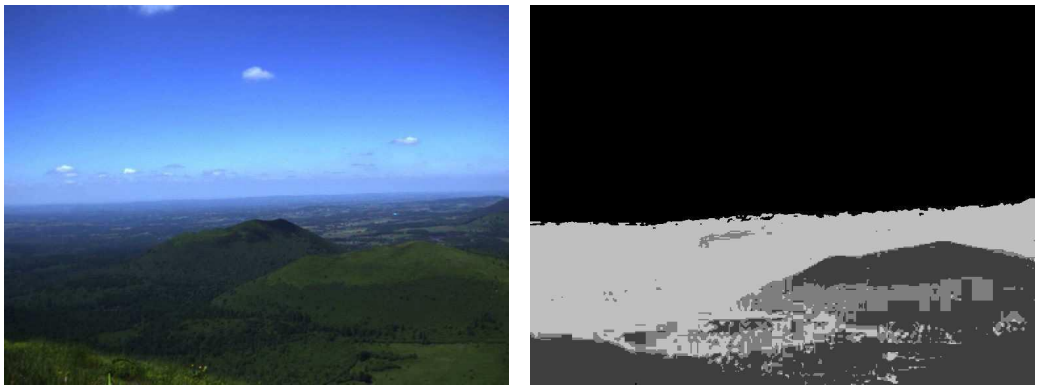
Voici un descriptif détaillé de chaque étape.

C.1.1 Segmentation basée HSV

Les images qui nous intéressent se composent essentiellement de chaine de montagne verdoyante sur un fond de ciel. Ceci nous amène à utiliser l'espace de couleur comme critère de segmentation. Il existe plusieurs types d'espace de couleurs tel que l'espace RGB ou l'espace HSV qui sont les plus connus. L'espace RGB décompose la couleur en trois canaux (Rouge, Vert et Bleu). Chaque couleur est une combinaison de ses trois couleurs. Cependant, utiliser cet espace de couleur pour segmenter nos images ne fournit pas un bon résultat car il est très difficile de trouver des seuils qui permettent de distinguer entre les montagnes et le ciel (la teinte verte contient du bleu et vice versa). Pour cette raison, on se retourne vers l'espace de couleur HSV. L'espace HSV lui aussi décompose une couleur en trois composantes qui sont :

- Teinte : qui est codé selon un angle défini dans un cercle de couleur ;
- Saturation : qui est l'intensité de la couleur ;
- Valeur : qui représente la brillance de la couleur.

Cet espace de couleur est très utilisé pour la détection des visages car l'espace HSV est proche de la manière dont l'être humain distinct les couleurs. Nous allons utiliser cette espace pour essayer de segmenter notre image.



(a) Image Initiale (b) Image segmentée

FIG. C.3: Exemple de segmentation en région

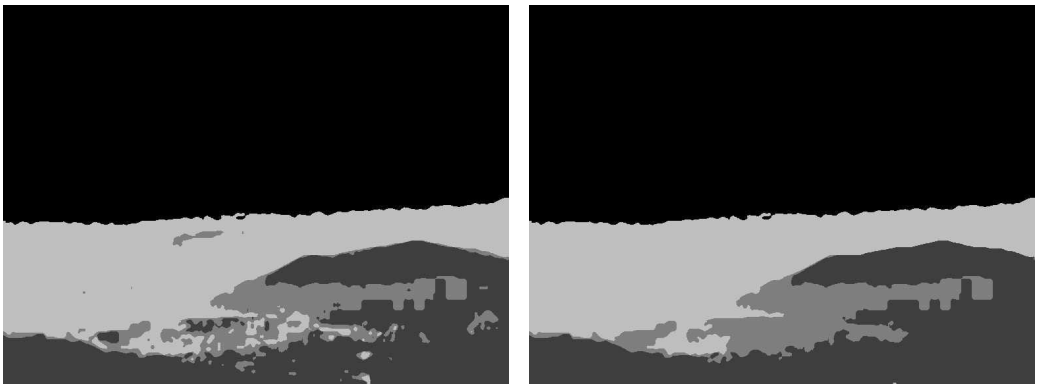
Notre segmentation n'utilise que deux des trois composantes à savoir la valeur et la saturation. Une première étape consiste à distinguer entre deux régions en utilisant la valeur. En effet, en utilisant la brillance, nous pouvons distinguer entre le ciel et la terre. Ceci a été constaté à partir des histogrammes sur différentes images. En observant ces histogrammes, nous avons constaté deux pics de valeurs répétitives dans chaque histogramme. En faisant le rapprochement avec les images, ceci correspondait parfaitement entre le ciel et les chaines de montagnes. Cette première phase de segmentation, nous permet aussi de délimiter la ligne d'horizon qui correspond à la frontière entre la région ciel et région terre. La seconde

étape consiste à segmenter la région terre afin de distinguer entre les différentes montagnes en utilisant la composante teinte. A partir des histogrammes de teinte réalisés sur différentes images, nous distinguons entre trois plages de valeurs qui correspondent à trois régions de couleur vertes distinctes. A la fin de cette phase, nous obtenons une image où nous distinguons entre différente régions montagneuses (cf. fig. C.3).

C.1.2 Filtrage

La phase de segmentation, nous permet de partager l'image en plusieurs régions. Cependant, à partir de cette segmentation, les contours obtenus comprennent des contours parasites qui ne correspondent pas à des crêtes de montagnes. Pour ceci, nous avons besoin de filtrer l'image pour ainsi absorber les petites régions parasites et ainsi ne garder que des régions bien distinctes. Un premier traitement consiste à utiliser un filtre médian. Le filtre médian permet d'absorber les petites régions parasites sans altérer la forme des contours (cf. fig. C.4-a). En effet, nous aurions pu utiliser des opérateurs morphologiques tels qu'une dilatation et érosion. Mais ce type de filtre altère énormément la forme des contours ce qui n'est pas souhaité.

Après un filtre médian, il reste toujours des régions qui sont assez grandes pour ne pas être résorbées par le filtre médian mais assez petit pour ne pas correspondre à des crêtes de montagnes. Pour cela, nous fusionnons les petites régions avec les plus grandes régions en se basant sur le cardinal. En effet, les régions dont le cardinal (nombre de pixels) est inférieur à un certain seuil sont fusionnées avec les régions voisines principales. Par région principale, nous désignons la région ayant la frontière la plus importante. Le filtre médian en prétraitement, nous permet d'éviter d'itérer plusieurs fois pour absorber les petites régions (cf. fig. C.4-b).

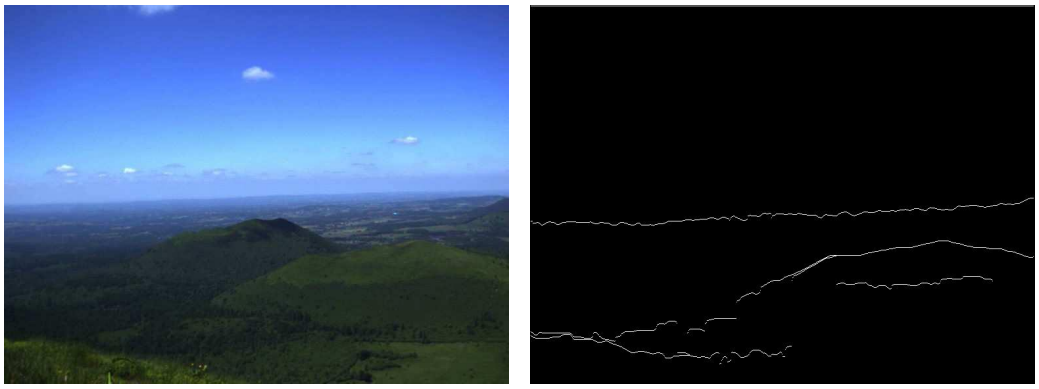


(a) Application du filtre Median (b) Fusion des petites régions

FIG. C.4: Exemple d'application du filtrage pour l'élimination des régions parasites

C.1.3 Extraction des contours

Une fois les régions définies, nous extrairions les contours correspondants aux frontières des régions. Afin de n'avoir que les contours des sommets, nous gardons que les premières intersections de chaque région avec la ligne verticale en partant d'en haut. Nous observons dans la figure C.5 un exemple de résultat obtenu.

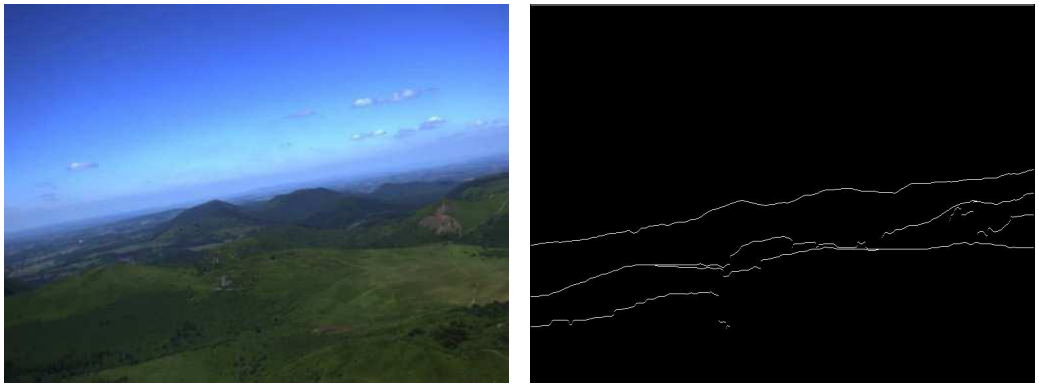


(a) Image initiale (b) Extraction des contours

FIG. C.5: Exemple d'extraction des contours à partir des régions obtenues

C.2 Résultats

Voici quelques résultats (cf. fig. C.6) obtenus sur quelques images panormaiques représentants des montagnes de la chaine du Puy du Dôme.



(a) Image initiale (b) Extraction des contours

(c) Image initiale (d) Extraction des contours

FIG. C.6: Exemple d'extraction des contours à partir des régions obtenues

Annexe D

GPS : fiabilité

La plupart des récepteurs sont capables d'affiner leurs calculs en utilisant plus de 4 satellites (ce qui rend les calculs plus robustes) tout en ôtant les sources qui semblent peu fiables, ou trop proche l'une de l'autre pour fournir une mesure correcte. On parle dans ce dernier cas de dilution de précision, mesurée par le facteur Global dilution of précision (GDOP).

Le GPS n'est pas utilisable dans toutes les situations : le signal émis par les satellites NAVSTAR étant assez faible, la traversée des couches de l'atmosphère est un facteur qui perturbe la précision de la localisation ; de même, les simples feuilles des arbres peuvent absorber le signal et rendre la localisation hasardeuse. De la même façon, l'effet canyon, particulièrement sensible en milieu urbain, consiste en l'occultation d'un satellite par le relief (un bâtiment par exemple) ou pire encore, en un écho du signal contre une surface qui n'empêchera pas la localisation mais fournira une localisation fausse (problèmes des multi-trajets des signaux GPS).

En l'absence d'obstacles, il reste quand même un facteur de perturbation important : la traversée des couches basses de l'atmosphère. La présence d'humidité et les modifications de pression de la troposphère modifient l'indice de réfraction n et donc la vitesse (et la direction) de propagation du signal radio. Si le terme hydrostatique est actuellement bien connu, les perturbations dues à l'humidité nécessitent, pour être corrigées, la mesure du profil exact de vapeur d'eau en fonction de l'altitude, une information difficile à collecter, sauf par des moyens extrêmement onéreux comme les lidars, qui ne donnent que des résultats parcellaires. Les récepteurs courants intègrent un modèle de correction.

Il existe un autre facteur de perturbation atmosphérique : la traversée de l'ionosphère. Cette couche ionisée par le rayonnement solaire modifie la vitesse de propagation du signal. La plupart des récepteurs intègrent un algorithme de correction mais en période de forte activité solaire, cette correction n'est plus assez précise. Pour corriger plus finement cet effet, certains récepteurs (bi-fréquences) utilisent le fait que les deux fréquences du signal GPS (L1 et L2) ne sont pas affectées de la même façon et recalculent ainsi la perturbation réelle.

Annexe E

Composants développés

Voici un aperçu des composants développés pour le système de localisation. Nous trouvons dans :

1. La figure E.1 : l'automate qui régit l'application associée au système de localisation ;
2. La figure E.2 : la feuille décrivant l'initialisation des capteurs, et le chargement des paramètres ;
3. La figure E.3 : la feuille qui comprend la phase d'initialisation semi-automatique ;
4. La figure E.4 : la feuille qui est associée au sous-système de vision ;
5. La figure E.5 : la feuille associée au sous-système AL ;
6. La figure E.6 : la feuille qui décrit la phase de réinitialisation automatique.

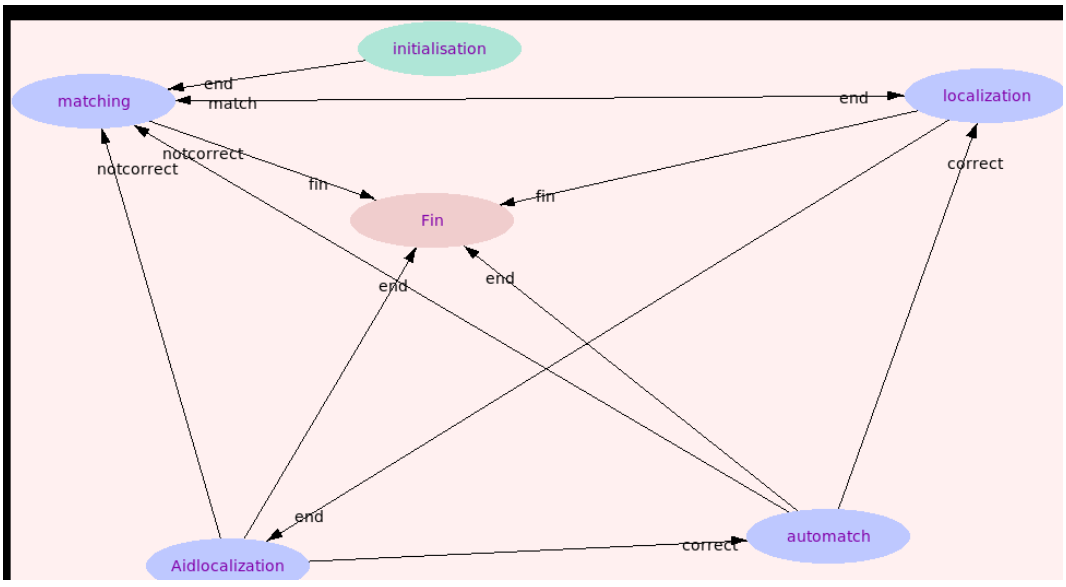


FIG. E.1: L'automate à états finis représentant le système de localisation

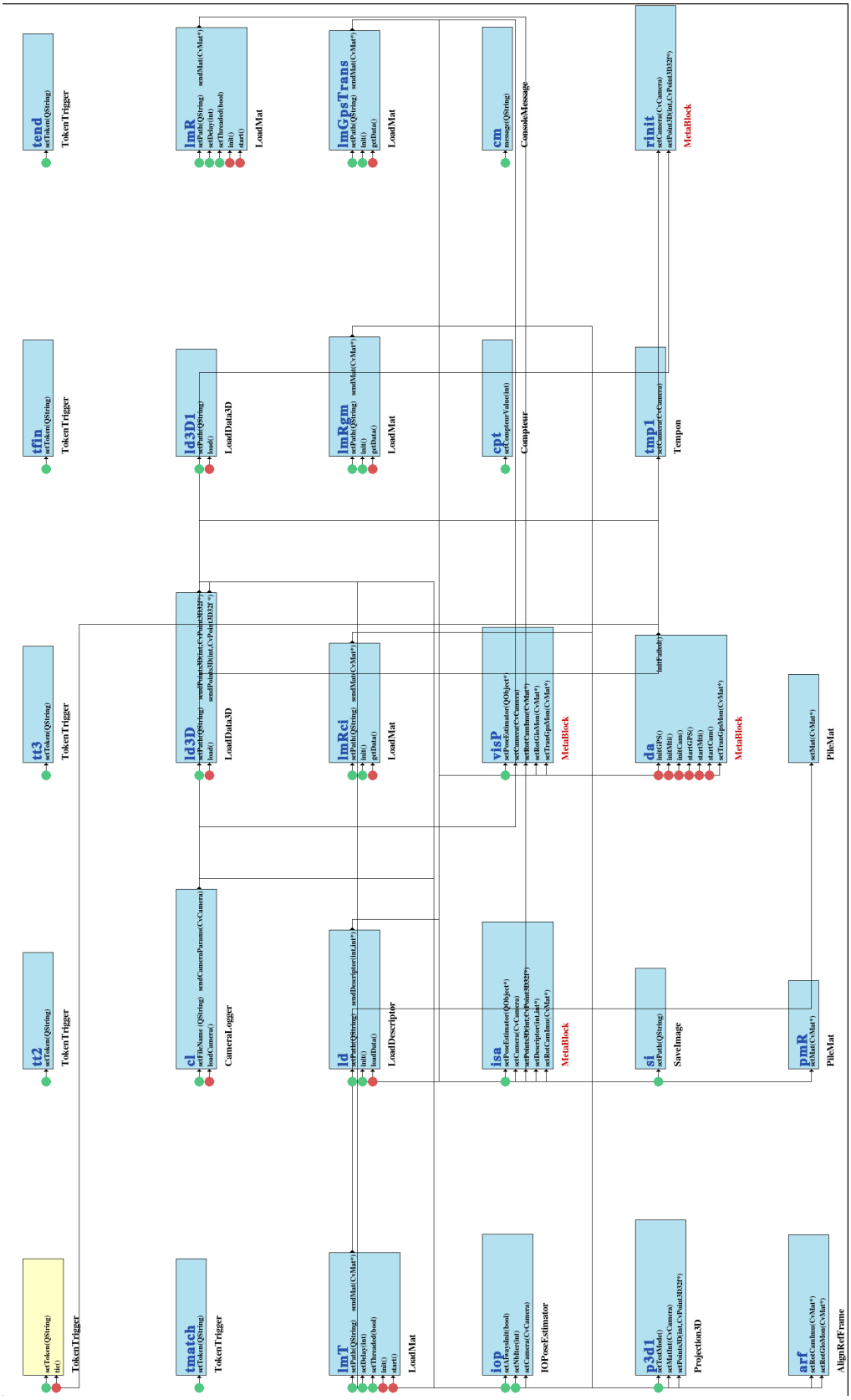


FIG. E.2: Les composants utilisés dans la feuille *initialisation* : chargement des paramètres, et initialisation des capteurs

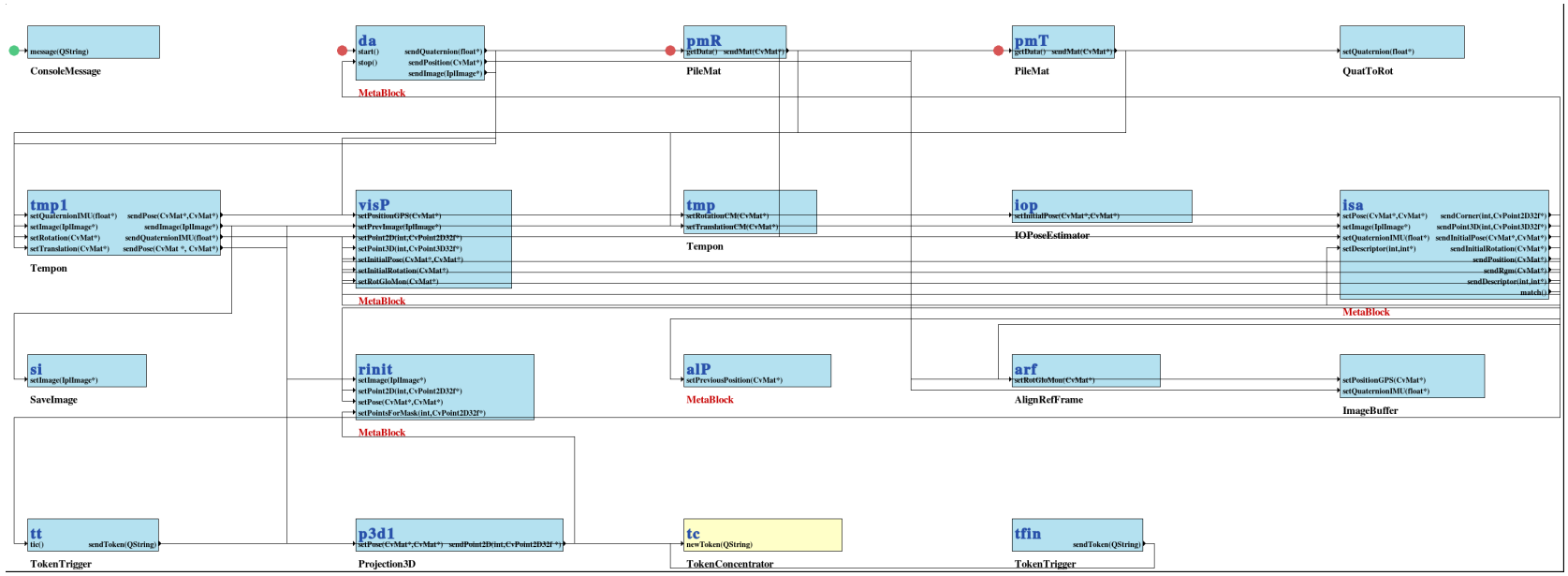


FIG. E.3: Les composants utilisés dans la feuille *matching* : l'initialisation semi-automatique

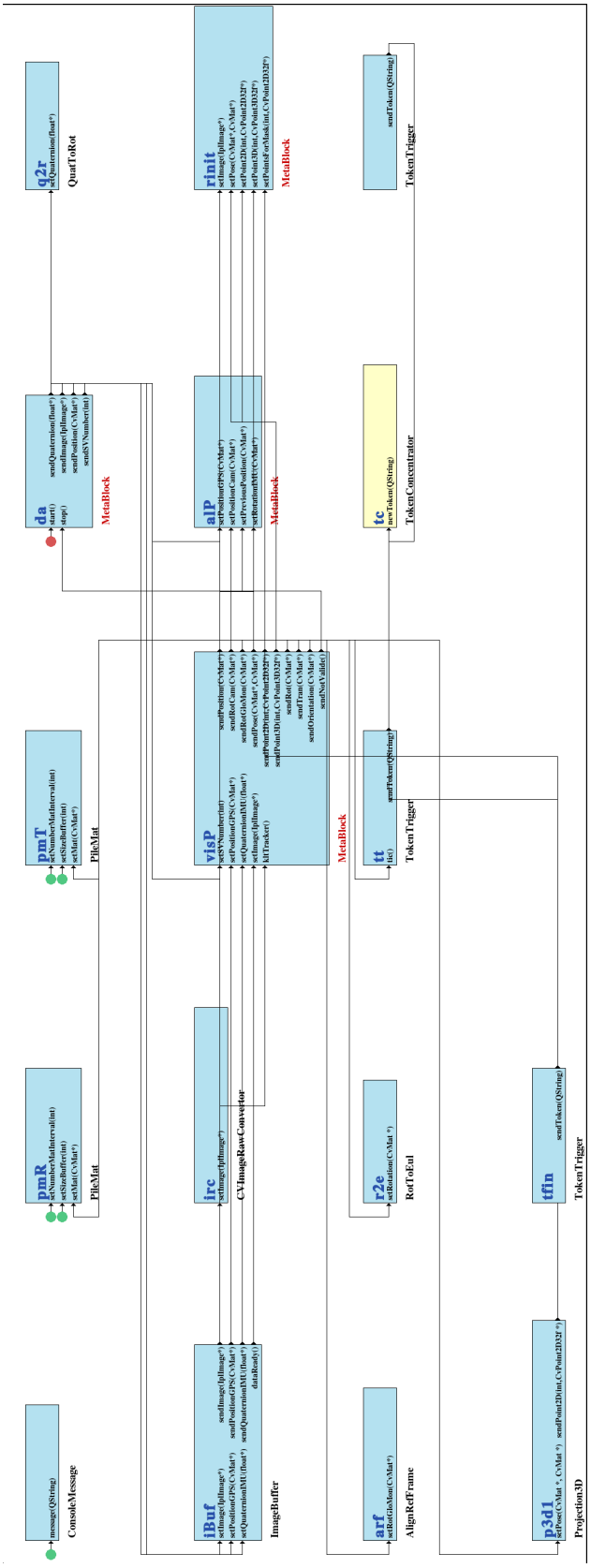


FIG. E.4: Les composants utilisés dans la feuille *localization* représentant le fonctionnement du sous-système de vision

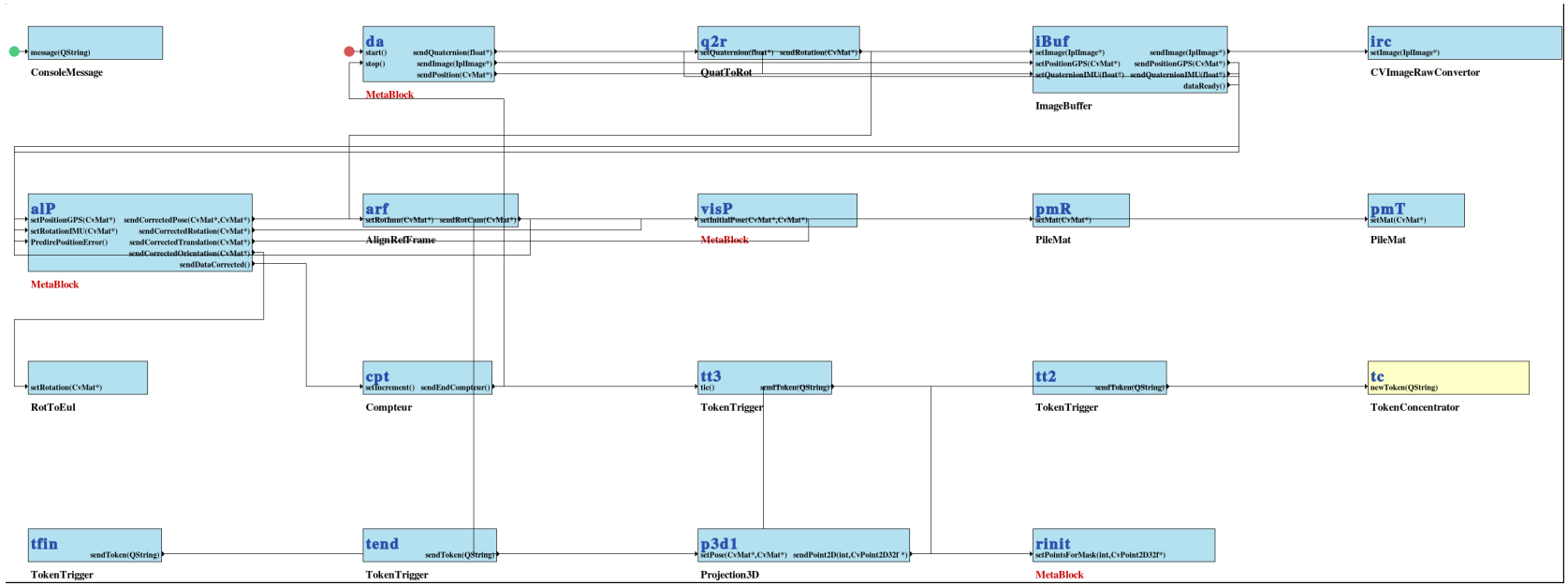


FIG. E.5: Les composants utilisés dans la feuille *Aidlocalization* représentant le fonctionnement du sous-système d'assistance à la localisation

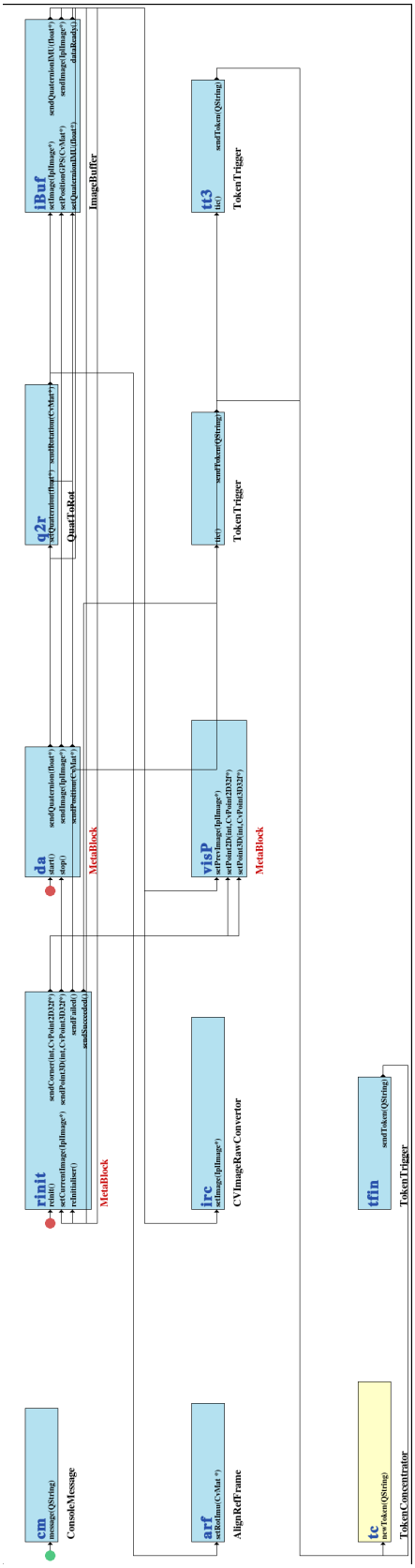


FIG. E.6: Les composants utilisés dans la feuille *automatch* représentant la phase de réinitialisation

Résumé

La démocratisation des terminaux mobiles telle que les téléphones cellulaires, les PDAs et les tablettes PC a rendu possible le déploiement de la réalité augmentée dans des environnements en extérieur à grande échelle. Cependant, afin de mettre en œuvre de tels systèmes, différentes problématiques doivent êtres traitées. Parmi elle, la localisation représente l'une des plus importantes. En effet, l'estimation de la position et de l'orientation (appelée pose) du point de vue (de la caméra ou de l'utilisateur) permet de recaler les objets virtuels sur les parties observées de la scène réelle. Dans nos travaux de thèse, nous présentons un système de localisation original destiné à des environnements à grande échelle qui utilise une approche basée vision sans marqueur pour l'estimation de la pose de la caméra. Cette approche se base sur des points caractéristiques naturels extraits des images. Etant donné que ce type d'approche est sensible aux variations de luminosité, aux occultations et aux mouvements brusques de la caméra, qui sont susceptibles de survenir dans l'environnement extérieur, nous utilisons deux autres types de capteurs afin d'assister le processus de vision. Dans nos travaux, nous voulons démontrer la faisabilité d'un schéma de suppléance dans des environnements extérieurs à large échelle. Le but est de fournir un système palliatif à la vision en cas de défaillance permettant également de réinitialiser le système de vision en cas de besoin. Le système de localisation vise à être autonome et adaptable aux différentes situations rencontrées.

Mots clés :
Réalité augmentée mobile, application en extérieur, système multi-capteurs, suppléance de données, estimation de pose sans marqueurs, appariement 2D/3D, prédiction d'erreur

Abstract

The democratization of mobile devices such as smartphones, PDAs or tablet-PCs makes it possible to use Augmented Reality systems in large scale environments. However, in order to implement such systems, many issues must be adressed. Among them, 3D localization is one of the most important. Indeed, the estimation of the position and orientation (also called pose) of the viewpoint (of the camera or the user) allows to register the virtual objects over the visible part of the real world. In this paper, we present an original localization system for large scale environments which uses a markerless vision-based approach to estimate the camera pose. It relies on natural feature points extracted from images. Since this type of method is sensitive to brightness changes, occlusions and sudden motion which are likely to occur in outdoor environment, we use two more sensors to assist the vision process. In our work, we would like to demonstrate the feasibility of an assistance scheme in large scale outdoor environment. The intent is to provide a fallback system for the vision in case of failure as well as to reinitialize the vision system when needed. The complete localization system aims to be autonomous and adaptable to different situations. We present here an overview of our system, its performance and some results obtained from experiments performed in an outdoor environment under real conditions.

Key words:
Mobile Augmented Reality, Outdoor application, hybrid sensor, fallback system, markerless pose estimation, 2D/3D matching, error prediction.