Reconnaissance 3D de gestes pour l'interaction homme-système

Hajar Hiyadi

NNT : 2016SACLE052

Thèse de doctorat
de
l'Université Mohammed V Rabat
Et de
l'Université Paris-Saclay
préparée à l'Université d'Evry Val d'Essonne

École doctorale n°580
Sciences et Technologies de l'Information et de la Communication
Spécialité de doctorat : Robotique

par

Mlle. Hajar HIYADI

Reconnaissance 3D de gestes pour l'interaction homme-système

Thèse présentée et soutenue à "Rabat", le 08 Décembre 2016 :

Composition du Jury :

M. E. H. BOUYAKHF, Professeur, Université Mohammed V Rabat, Président du Jury
M. D. MERAD, Maître de Conférences, Université Aix Marseille, Rapporteur
M. A. TAMTAOUI, Professeur, INPT Rabat, Rapporteur
M. M. JEDRA, Professeur, Université Mohammed V Rabat, Examinateur
M. C. MONTAGNE, MCF, Université d'Evry, Encadrant de thèse
Mme. F. REGRAGUI, Professeure, Université Mohammed V Rabat, Co-Directrice de thèse
M. F. ABABSA, Maître de Conférences - HDR, Université d'Evry, Directeur de thèse

Avant Propos

Les travaux présentés dans ce rapport ont été effectués au sein du laboratoire Informatique, Mathématiques Appliquées, Intelligence Artificielle et de Reconnaissance de Forme (LIMIARF) à la faculté des sciences de Rabat.

La thèse a été préparée en collaboration avec le laboratoire Informatique, Biologie Intégrative et Systèmes Complexes (IBISC) à l'université Evry Val d'Essonne en France.

Je tiens à exprimer mes plus vifs remerciements à Madame Fakhita REGRAGUI qui fut pour moi une directrice de thèse attentive et disponible malgré ses nombreuses charges. J'adresse aussi toute ma gratitude à mon co-directeur de thèse Monsieur El houssine BOUYAKHF pour son temps qu'il m'avait accordé durant ma thèse et son effort fourni pour qu'elle se passe dans les meilleurs conditions. J'exprime tous mes remerciements à mon encadrant de thèse Monsieur Fakhreddine ABABSA pour son suivi permanent. Sa compétence, sa rigueur scientifique et sa clairvoyance m'ont beaucoup appris. Je tiens aussi à remercier mon co-encadrent Monsieur Christophe MONTAGNE pour son aide, son écoute, sa disponibilité et surtout ses judicieux conseils, qui ont contribué à alimenter ma réflexion.

Je remercie Professeur BOUYAKHF une deuxième fois d'avoir accepté d'être le président du jury de mon travail. Je voudrais exprimer ma reconnaissance envers tout les membres du jury d'avoir accepté d'assister à la soutenance de mon travail. Merci au Professeur Mohamed JEDRA d'avoir accepté d'être parmi les membres du jury. Un grand merci au Professeur Ahmed TAMTAOUI et au Professeur habilité Monsieur Djamel MERAD d'avoir accepté de rapporter mon travail.

Merci

Résumé

Cette thèse porte sur la reconnaissance de gestes pour l'interaction naturelle homme-système basée sur les gestes. L'objectif des travaux présentés dans ce manuscrit est de proposer des approches de reconnaissance de différents types de geste dynamiques, simples (un seul geste dans la séquence) et composés (deux gestes simples ou plus), et aussi de mettre en oeuvre une démarche comparative entre les approches proposées. Le but des applications visées par l'interaction homme-système est de parvenir à une interaction naturelle qui simule l'interaction homme-homme. Comme dans la communication homme-homme, les gestes sont aussi très utilisés dans la communication homme-système. On distingue deux ensembles d'approches dans ce domaine : les méthodes basées sur les dispositifs de contact (gants, marqueurs) et celles basées sur la vision. Les méthodes basées sur les dispositifs de contact sont contraignantes. La vision par ordinateur est la seule technologie permettant la reconnaissance de gestes, sans interférence entre la personne et le système, qui est à un faible coût. Un geste dynamique correspond à une variation temporelle de la forme et de la position du membre. Dans cette thèse, nous proposons deux approches pour la reconnaissance de gestes dynamiques : a) une approche de reconnaissance des gestes simples, b) une approche de reconnaissance des gestes composés. La première partie de la thèse se concentre sur la description des gestes à reconnaître. Cette description est basée sur des informations pertinentes qui caractérisent chaque geste et le rendent distinctif, telle que la profondeur, la combinaison des informations 3D des articulations et leurs angles. Ces informations sont extraites en temps réel à partir d'un flux provenant d'un capteur kinect en utilisant l'algorithme Skeleton fourni par le constructeur. L'idée est d'extraire les coordonnées 3D des articulations qui se trouvent sur la partie supérieure du corps humain et ensuite de calculer les angles correspondants. Nous avons mis en place un protocole expérimental ainsi qu'une base de données qui nous a servi pour les entraînements, les tests et les évaluations. Dans un second temps, nous décrivons en détails les approches proposées. La première est basée sur les Modèles de Markov Cachés. Nous avons proposé un modèle MMC pour chaque geste.

La variation des angles entre les articulations est utilisée comme entrée des Modèles de Markov Cachés. La deuxième approche traite le cas des gestes composés et successifs dans une même séquence. Cette approche combine la méthode de la Déformation Temporelle Dynamique (Dynamic Time Warping) avec une fenêtre glissante adaptative d'où le nom de l'approche : Adaptive Dynamic Time Warping. Afin de reconnaître tous les gestes d'une séquence dans le bon ordre, nous utilisons une fenêtre adaptative pour parcourir la séquence du geste composé en l'alimentant à chaque fois avec de nouvelles données. Ensuite, nous utilisons DTW pour comparer les gestes de référence avec les séquences définies par la fenêtre adaptative. La reconnaissance des gestes composés à base de ADTW donne de très bons résultats. Cependant, le taux moyen de reconnaissance de gestes composés avec transition est inférieur à celui des gestes composés sans transition.

Table des matières

Résumé **5**

Table des figures **11**

Liste des tableaux **13**

1 Introduction générale **15**

1.1 Contexte de l'étude 16

1.2 Motivations 18

1.3 Contributions et Publications 20

1.4 Organisation du manuscrit 21

2 État de l'art **23**

2.1 Introduction 23

2.1.1 Définition et nature du geste 24

2.1.2 Les technologies 26

2.2 L'extraction des caractéristiques 29

2.2.1 La profondeur 30

2.2.2 La couleur 30

2.2.3 Modèle de frontière elliptique 30

2.2.4 Combinaison des indices 32

2.2.5 Fusion profondeur/couleur 32

2.2.6 Les points d'interêt 33

2.2.7 Le sac des caractéristiques 34

2.3 La détection 35

2.3.1 K-means Expectation Maximization 35

2.3.2 Mean Shift 6D 36

2.3.3 Modèle 2D/3D de la tête . . . 37
2.4 Le suivi (Tracking) . . . 37
2.4.1 Les coordonnées 3D . . . 38
2.4.2 Fusion profondeur/silhouette . . . 39
2.4.3 Unscented Kalman Filter . . . 39
2.4.4 Classification de la couleur . . . 41
2.4.5 Filtrage particulaire . . . 41
2.5 La reconnaissance de gestes . . . 42
2.5.1 L' apprentissage automatique . . . 42
2.5.2 La classification du geste . . . 43
2.5.3 Les modèles 2D/3D . . . 46
2.5.4 Le Template Matching . . . 46
2.6 Synthèse et discussion . . . 47

3 Modélisation du geste 49
3.1 Introduction . . . 49
3.1.1 Les objectifs . . . 49
3.1.2 Les contraintes . . . 49
3.1.3 L'approche proposée . . . 49
3.2 Représentation du flux de données . . . 50
3.2.1 Les capteurs . . . 50
3.3 L'extraction de données . . . 53
3.4 Description des gestes . . . 55
3.5 Protocole expérimental . . . 61
3.5.1 La base de données . . . 61
3.5.2 Protocole expérimental . . . 62
3.5.3 Critères d'évaluation . . . 65
3.6 Conclusion . . . 66

4 Reconnaissance de gestes simples 69
4.1 Introduction . . . 69
4.2 La méthode basée MMC . . . 70
4.2.1 Formalisme . . . 70
4.2.2 Résultats expérimentaux . . . 77

4.3 La méthode basée sur MMC et DTW 80
4.3.1 Formalisme 80
4.3.2 Résultats 85
4.4 Conclusion 86

5 Reconnaissance de gestes composés **87**
5.1 Introduction 87
5.2 La méthode ADTW 88
5.2.1 DTW pour la reconnaissance de gestes 88
5.2.2 La fenêtre adaptative 88
5.2.3 ADTW pour la reconnaissance de gestes 89
5.2.4 Résultats expérimentaux 94
5.3 Conclusion 95

Bibliographie **103**

Table des figures

2.1 Une taxonomie des catégories du geste. 26
2.2 L'Animazoo IGS-190 contient dix-huit gyros pour la reconnaissance de geste en plus des marqueurs visuels en jaune. 28

3.1 Une caméra temps de vol (TOF). 51
3.2 Une caméra stéréoscopique. 52
3.3 Un capteur Microsoft Kinect. 53
3.4 Système de coordonnées de la Kinect. 54
3.5 (a) image RGB, (b) image de profondeur, (c) suivi du Skeleton. 55
3.6 Les articulations suivies par la Kinect 56
3.7 Skeleton actif détecté par la Kinect 56
3.8 Les cinq gestes à reconnaître. 57
3.9 Les angles α, β et γ. 58
3.10 Les variations des angles pour le geste *Viens*. 58
3.11 Les variations des angles pour le geste *Recule*. 59
3.12 Les variations des angles pour le geste *Pointage à droite*. 59
3.13 Les variations des angles pour le geste *Pointage à gauche*. 59
3.14 Les variations des angles pour le geste *Stop*. 60
3.15 Les cas d'échec de la détection par la Kinect ; première image : la distance est supérieure à $3m$, deuxième et troisième image : le sujet n'est pas face à la Kinect. 63

4.1 Les principales étapes de notre système de reconnaissance 70
4.2 Les topologies des MMCs 72
4.3 Le taux de reconnaissance moyen du système de reconnaissance en variant le nombre d'états des cinq MMCs de 3 à 14 états. 78
4.4 Alignement de deux séquences avec DTW. 81

4.5 Matrice de distance DTW entre deux séquences. 81
4.6 Combinaison MMC et DTW pour la reconnaissance de gestes. 84

5.1 Le nombre des données nécessaires pour une bonne reconnaissance. 89
5.2 La méthode *Foward Déformation Temporelle Dynamique Adaptative.* 91
5.3 La méthode *Backward Déformation Temporelle Dynamique Adaptative.* 93
5.4 Le taux de reconnaissance du geste *viens* sans et avec début ambigu 95
5.5 Le taux de reconnaissance du geste *recule* sans et avec début ambigu 96
5.6 Le taux de reconnaissance du geste *pointage à droite* sans et avec début ambigu . 96
5.7 Le taux de reconnaissance du geste *pointage à gauche* sans et avec début ambigu 97
5.8 Le taux de reconnaissance du geste *stop* sans et avec début ambigu 97

Liste des tableaux

2.1 Comparaison entre les dispositifs de contact et les dispositif de la vision. 29

3.1 Les variations de l'angle principal dans chaque geste. 61

4.1 La matrice de confusion et la précision des différents gestes avec entraînement. . 78
4.2 La comparaison des performances de notre méthode avec la méthode de l'article [Gu et al., 2012] (N. de P. : Nombre de personnes). 79
4.3 Les seuils des classes 85
4.4 La matrice de confusion de la méthode MMC/DTW 86

5.1 Le taux de reconnaissance des gestes composés sans transition. 94
5.2 Le taux de reconnaissance des gestes composés avec transition. 94

Chapitre 1

Introduction générale

La reconnaissance de gestes est un domaine de recherche en vogue depuis les trois dernières décennies. Le geste est une forme importante dans la communication et l'interaction humaine. Et, comme dans de nombreux autres domaines, la recherche scientifique a copié l'homme réutilisant le mécanisme de geste et de reconnaissance du geste dans des interactions avec la machine ou entre les machines. Les mains sont d'habitude utilisées pour interagir avec des objets (prendre, déplacer) et notre corps gesticule pour communiquer avec d'autres personnes (non, oui, arrêtez-vous). Ainsi, plusieurs applications de reconnaissance de gestes ont été développées, jusqu'à présent, grâce aux succès obtenus dans les sous domaines de l'intelligence artificielle (apprentissage automatique, vision cognitive, contrôle multi-modal). Par exemple, l'homme peut interagir avec les machines via un dispositif de reconnaissance de gestes (on cite Wii-mote dans [Schmidt et al., 2008], CyberGlove dans [Kevin et al., 2004] et Multi-touch screen dans [Webel et al., 2008]). Cependant, les méthodes basées sur un dispositif de contact sont intrusives et nécessitent une utilisation correcte de l'outil. Par contre, les méthodes basées vision proposent de surmonter ces limites et permettent une reconnaissance de gestes sans contact avec ou sans une coopération de l'utilisateur (ex : les marqueurs du corps ou -body markers-). Les méthodes basées sur la vision ont cependant leurs propres limites telles les variations d'illumination ou divers problèmes liés à l'arrière-plan qui perturbent les traitements d'images. De plus, ces méthodes impliquent l'usage de caméras qui sont des dispositifs plus fragiles et des capteurs moins précis que les appareils de contact.

Dans cette thèse, nous visons à développer un système basé vision pour la reconnaissance de gestes. Après avoir présenté les motivations dans la section 1.1, le contexte de l'étude est décrit dans la section 1.2. La section 1.3 expose les objectifs, les hypothèses et les contributions de ce

travail et la section 1.4 conclut cette introduction en exposant la structure du manuscrit.

1.1 Contexte de l'étude

Une interaction homme-système naturelle et efficace nécessite des approches non contraignantes pour les utilisateurs, basées sur des gestes naturels. Parmi ces gestes, celui de la main représente une modalité intéressante. L'objectif des recherches dans ce domaine est de développer des méthodes robustes qui identifient puis utilisent les gestes humains pour contrôler une application. Les gestes de la main sont une collection de mouvements de la main et du bras qui peuvent varier d'une posture statique, comme le pointage sur un objet, à des gestes dynamiques utilisés pour la communications entre personnes. La reconnaissance de ces gestes nécessite leur modélisation dans les domaines spatial et temporel. La posture de la main est la structure statique de la main tandis que son mouvement dynamique est dit geste de la main, et les deux formes sont particulièrement cruciales pour l'interaction homme-système.

Les informations liées aux gestes de la main ont une structure spatiale aussi bien qu'une structure temporelle. Les méthodes de reconnaissance de gestes sont principalement divisées en deux catégories : les méthodes basées sur les dispositifs attachés aux corps tels que les gants et les marqueurs, et les méthodes basées vision. Les approches basées sur les dispositifs attachés au corps utilisent des capteurs mécaniques ou optiques connectés au dispositif qui convertissent les flexions des articulations en des signaux électriques pour reconnaître la posture de la main. Cette méthode freine l'interaction naturelle parce qu'elle exige que l'utilisateur porte des dispositifs sans fil ou liés à l'ordinateur via des câbles. Cependant, ce genre de dispositif est relativement cher et inconfortable. Au contraire, les techniques basées vision n'ont besoin que d'une seule caméra réalisant, par conséquent, une interaction naturelle homme système sans le besoin de dispositifs supplémentaires. Cependant, ces techniques exigent que l'utilisateur et la caméra soient indépendants et que les méthodes développées soient invariables vis à vis des changements de prise de vue (ex : l'arrière plan, des transformations et des conditions de luminosité) et qu'elles tournent en "temps réel". Par ailleurs, puisque la main est un objet déformable et articulé, ceci augmente la difficulté du processus de segmentation et de reconnaissance de forme.

Les approches basées vision utilisées dans la détection et la reconnaissance du geste permettent une interaction naturelle homme-système. Cependant, c'est aussi la plus difficile à cause des inconvénients du matériel de la vision d'aujourd'hui. Les approches basées vision

nécessitent une ou plusieurs caméras. Dans certains environnements visuels, les informations sur l'utilisateur sont obtenues (la couleur, la silhouette, etc.) et le geste est extrait. Cependant, il existe plusieurs défis : La détection du corps se déplaçant dans un environnement encombré, l'analyse de mouvements, le suivi des positions du corps par rapport à l'arrière plan, ainsi que la reconnaissance des postures et des gestes.

Les méthodes basées vision diffèrent entre elles par :

- Le nombre de caméras.
- Le temps des traitements.
- La nature de l'environnement.
- Les exigences de l'utilisateur. Est ce que la personne doit porter un objet/vêtement particulier ; comme des marqueurs, des gants, des longues manches, etc. ?
- Les caractéristiques visuelles. Quelles sont les caractéristiques de bas niveau extraites de l'image et utilisées ensuite pour la détection et la reconnaissance ? La couleur ? La forme ? Les bords ? Les régions ? Les silhouettes ? Les histogrammes ?
- La eprésentation du geste : en 2D ou en 3D ?
- La représentation du temps et de l'espace : Comment les aspects temporels et spatiaux du geste sont-ils représentés et utilisés dans la reconnaissance ?

Il y a, généralement, une perte d'informations lors de la projection d'un objet 3D dans un plan 2D. En plus, l'élaboration des modèles 3D inclut des espaces de paramètres de grandes dimensions. De même, le suivi doit être robuste face aux changements de taille et de forme, aux autres objets qui bougent dans l'arrière plan, et aux bruits. Un nouveau système basé vision est présenté dans ce manuscrit, il permet une interaction homme-système par l'intermédiaire de gestes dynamiques. Ce système se veut robuste vis à vis des différentes conditions de luminosité ainsi que des arrières plan encombrés. Nous proposons d'étudier les différentes étapes nécessaires à la reconnaissance de gestes à savoir la détection, le suivi de la partie supérieure du corps, l'extraction des informations articulaires à partir des images de profondeur et enfin la classification du geste et sa reconnaissance.
En générale, le problème de reconnaissance de gestes est traité comme étant un problème de reconnaissance de formes où un ensemble de caractéristiques est extrait à partir des images ou vidéos et comparé ensuite à une représentation prédéfinie.

1.2 Motivations

La détection et le suivi en temps réel de personnes à partir des images ou des vidéos est un défi majeur pour les applications d'interaction homme-système. Ces dernières années, beaucoup de travaux ont été consacrés à l'amélioration de la robustesse des approches de reconnaissance de gestes. Pour atteindre cet objectif, ces travaux proposaient d'améliorer soit le dispositif d'acquisition des mouvements et des gestes, soit l'approche de suivi et de reconnaissance de gestes. La reconnaissance de gestes est utilisée dans plusieurs domaines d'application. Allant de l'interprétation du langage des signes jusqu'aux environnement virtuels avec des interfaces intelligents homme-machine. Aujourd'hui, le nombre des applications ne cesse d'augmenter, et les solutions proposées sont de plus en plus efficaces. Ci-dessous, quelques exemples d'applications de la reconnaissance de gestes à partir de vidéos.

- Le langage des signes. Tandis que dans la reconnaissance de la parole le but est de transcrire le discours pour en extraire un message, le but de la reconnaissance, dans le langage des signes, est de transcrire les gestes du langage des signes pour en extraire un message. Les études dans ce type d'applications se concentrent principalement, sur la reconnaissance des gestes de la main et de la tête. Un geste dans le langage des signes peut être soit statique ou dynamique. Par exemple, [Swee et al., 2007] propose un système pour reconnaître le langage des signes avec un ensemble de capteurs pour mesurer le mouvement des épaules, des coudes, poignets, paumes et doigts. Le système est capable de reconnaître 25 mots à partir des signes en se basant sur les Modèles de Markov Cachés.

- La réalité virtuelle. Les environnements virtuels permettent à l'utilisateur d'interagir avec un environnement simulé (le monde simulé peut être, soit un modèle d'un monde réel ou bien imaginaire). Elle inclut les jeux immersifs, les simulateurs de vol. Ici, la reconnaissance de gestes et utilisée comme un moyen de communication avec le monde virtuel.

- Les interfaces hommes-machines. Actuellement, l'homme communique beaucoup avec les machines et à l'aide des machines. C'est pourquoi, on essaye de créer des dispositifs plus cognitifs et intelligents qui peuvent remplacer les souris, les claviers et tous les outils placés entre l'homme et la machine.

- L'interaction homme-robot. Le domaine de la robotique a beaucoup évolué. L'interaction homme-robot est basée sur la reconnaissance de gestes humains par le robot. Ceci per-

met au robot de communiquer avec la personne dans le but de l'aider dans ses tâches quotidiennes (ex. personne en situation d'handicap).

- La biométrie. Il s'agit, d'une part, de créer des applications qui consistent à compter le nombre de personnes et décrire leur flux dans les lieux publics présentant un grand nombre de visiteurs, comme les aéroports ou les stations de métros. D'autre part, de pouvoir détecter des comportements suspects (vol, crime, acte terroriste,...). [Kage et al., 2007] proposent une approche pour détecter les actes de violence dans un ascenseur basée sur le flux optique. Les visages des personnes violentes sont ensuite reconnus.

- Les jeux. Utiliser les mouvements ou la posture d'une personne pour agir sur l'environnement ou l'interface du jeu...

Ces applications nécessitent d'effectuer un suivi d'un ou de plusieurs membres du corps humain. Le suivi permet d'effectuer la mise en correspondance, image par image, des caractéristiques précédemment détectées. Les problèmes qui rendent cette tâche difficile sont liés à la variabilité de l'environnement et de la personne suivie. En effet, l'environnement dans lequel évolue la personne est généralement imprévisible : changement de la luminosité, complexité des arrières plans, ombre des personnes et des objets. Pour le suivi de personnes, il est généralement difficile d'identifier les différentes parties du corps, l'habillement, la rapidité du mouvement, la présence de plusieurs personnes dans la scène, etc. Le suivi du mouvement dans l'espace 3D permet de définir l'orientation du geste, le pointage et la manipulation des objets. La notion de 3D est très importante dans la reconnaissance du geste car elle permet de définir le geste d'une façon très précise.

Les capteurs de vision comprennent les capteurs 2D et 3D. Cependant, il existe certaines limites dans la reconnaissance de gestes qui utilise les images 2D. En effet, les images ne peuvent pas être acquises sous le même niveau d'éclairage. De plus, les éléments de l'arrière plan peuvent rendre la reconnaissance très difficile. Avec l'émergence du capteur de profondeur (ex : Kinect) [Beleboni, 2015], l'information de profondeur obtenue rend le suivi robuste vis à vis des problèmes que présentent les caméras 2D.
Dans le cadre de cette thèse, nous proposons d'utiliser uniquement l'information 3D pour construire un système robuste de reconnaissance gestuelle pour l'interaction homme-système. La détection et le suivi du corps sont réalisés avec l'algorithme Skeleton fourni par le Kinect SDK. Durant le suivi, des informations sont extraites en temps réel et combinées pour former

des descripteurs. Ces descripteurs seront l'entrée du système de reconnaissance. Ce dernier est réalisé en se basant sur une méthode d'apprentissage automatique. Nous en avons testé quelques unes. Dans un premier temps, nous avons travaillé avec les méthodes de Markov Cachés (HMM). Ensuite, nous avons travaillé avec la Déformation Temporelle Dynamique (DTW). Nous avons combinés les deux approches. Le système est robuste aux contraintes précédemment citées liées à l'utilisateur et l'environnement.
Nous nous concentrons sur l'aspect dynamique des gestes et notamment sur la variabilité de leur forme et de leur vitesse d'exécution.

1.3 Contributions et Publications

Dans ces travaux, nous nous intéressons aux gestes déictiques notamment les gestes de contrôle et de pointage. Nos contributions majeures sont comme suit :

1) Proposition d'une technique de reconnaissance de gestes dynamiques basée sur l'information de profondeur fournie par la Kinect [Hiyadi et al., 2015a].

2) Identification d'un descripteur de gestes à partir des variations des angles liés aux articulations des membres actifs de la partie supérieure du corps. Nous avons obtenu un taux élevé de reconnaissance de gestes en utilisant les Modèles de Markov Cachés [Hiyadi et al., 2015a].

3) Développement d'une approche qui parvient à rejeter les gestes "faux positifs" au lieu de les mal classer, en combinant les Modèles de Markov Cachés avec la méthode de Déformation Temporelle Dynamique [Hiyadi et al., 2016b].

4) Proposition d'une méthode qui permet de reconnaître les gestes continus dans une même séquence en combinant une fenêtre adaptative avec l'algorithme Déformation Temporelle Dynamique [Hiyadi et al., 2016a].

Les publications résultantes des recherches menées dans la thèse sont listées ci-dessous :

1. [Hiyadi et al., 2015b] Hajar Hiyadi, Fakhr-Eddine Ababsa, El Houssine Bouyakhf, Fakhita Regragui, Christophe Montagne. Reconnaissance 3D des Gestes pour l'Interaction Naturelle Homme Robot. Journées francophones des jeunes chercheurs en vision par ordinateur, Juin 2015, Amiens, France.

2. [Hiyadi et al., 2015a] Hajar Hiyadi, Fakhreddine Ababsa, Christophe Montagne, El Houssine Bouyakhf, Fakhita Regragui, "A Depth-based Approach for 3D Dynamic Gesture

Recognition", Informatics in Control, Automation and Robotics 12th International Conference, ICINCO 2015 Colmar, France, July 21-23, 2015.

3. [Hiyadi et al., 2016b] Hajar Hiyadi, Fakhreddine Ababsa, Christophe Montagne, El Houssine Bouyakhf, Fakhita Regragui, "Combination of HMM and DTW for 3D Dynamic Gesture Recognition Using Depth Only", Informatics in Control, Automation and Robotics 12th International Conference, ICINCO 2015 Colmar, France, July 21-23, 2015 Revised Selected Papers. Springer International Publishing, 2016.

4. [Hiyadi et al., 2016a] Hajar Hiyadi, Fakhreddine Ababsa, Christophe Montagne, El Houssine Bouyakhf, Fakhita Regragui, "Adaptive Dynamic Time Warping for Recognition of Natural Gestures", International Conference on Image Processing Theory, Tools and Applications IPTA, Oulu, Finland, December 12-15, 2016.

5. [Hiyadi et al., 2016c] Hajar Hiyadi, Fakhreddine Ababsa, Christophe Montagne, El Houssine Bouyakhf, Fakhita Regragui, "Dynamic Gesture Recognition for Natural human system interaction", Journal of Theoretical and Applied Information Technology (JATIT), Vol.91. No.2 (374-383), 30th September 2016.

1.4 Organisation du manuscrit

Cette thèse inclut 6 chapitres :

1. Le premier chapitre introduit le contexte de l'étude, les différentes étapes de traitement de gestes en utilisant la vision, et les motivations de ce travail. Les objectifs et les contributions sont aussi présentés.

2. Le deuxième chapitre fournit une étude de l'état de l'art divisée en trois catégories : La détection et l'extraction des données, le suivi et enfin la reconnaissance des gestes. Une comparaison de toutes les méthodes est donnée à la fin du chapitre.

3. Le troisième chapitre présente notre modélisation des gestes à reconnaître, le capteur et les données utilisées pour décrire les gestes. Cette description est utilisée dans les méthodes présentées dans les chapitre 4 et 5. Le protocole expérimental est ensuite détaillé.

4. Le quatrième chapitre propose deux systèmes de reconnaissance de gestes. Le premier est basé sur les Modèles de Markov Cachés. Le deuxième utilise la Déformation Temporelle

Dynamique. Les deux sont utilisés pour reconnaître les gestes simples. Une comparaison des performances des deux méthodes est donnée.

5. Le cinquième chapitre présente deux approches de reconnaissance de gestes. La première approche combine les MMCs avec la DTW dans le but d'éliminer des gestes "faux positifs". La deuxième approche combine la DTW avec une fenêtre adaptative afin de reconnaître les gestes composés.

6. Le sixième chapitre fournit des conclusions et décrit le travail futur.

Chapitre 2

État de l'art

2.1 Introduction

La reconnaissance du geste humain consiste à identifier et interpréter automatiquement les gestes fournis par des capteurs (caméras) ou grâce à d'autres dispositifs (gants). Dans ce chapitre nous présentons un état de l'art sur la reconnaissance du geste humain qui inclut l'extraction et la représentation du geste, les techniques de suivi et de reconnaissance. Afin d'étudier la reconnaissance du geste, il est important de comprendre d'abord la définition et la nature du geste vues par la littérature. Quand nous essayons de définir le mot « geste »nous nous trouvons face à plusieurs questions : Quelles sont les différentes catégories de geste ? Pourquoi utilisons-nous les gestes ? Quelle genre d'information est transmise par l'intermédiaire des gestes ? Nous répondrons à ces questions dans la sous-section 2.1.1. Ensuite, nous introduirons, dans la sous-section 2.1.2, les deux principales catégories d'approches de reconnaissance de gestes en fonction du type de capteurs utilisés :

- Les approches basées contact : utilisant des dispositifs physiques de localisation et suivi (gants, marqueurs, manettes, etc.).
- Les approches basées vision : utilisant une ou plusieurs caméras.

Dans notre étude, nous nous concentrons principalement sur les approches basées vision. Par conséquent, les sections qui suivent sont reliées à cette catégorie. Dans la section 2.2, nous décrivons quelques méthodes de l'état de l'art pour l'extraction des caractéristiques à partir des images ou vidéos. Les différentes techniques de détection et segmentation du corps de la personne sont listées dans la section 2.3, celles du suivi sont listées dans la section 2.4, tandis que celles de la reconnaissance de gestes sont présentées dans la section 2.5. Nous finissons le

chapitre avec la section 2.5 qui présente une synthèse des approches basées vision et discute les défis de la reconnaissance des gestes humains.

2.1.1 Définition et nature du geste

En générale, on peut définir un geste comme un mouvement du corps. Un geste est une communication non verbale, qui peut être combiné avec une communication verbale ou la remplacer afin d'exprimer quelque chose. Il peut être articulé par les mains, les bras ou le corps, il peut aussi être un mouvement de la tête, du visage et des yeux, tels que cligner les yeux, hocher la tête ou rouler les yeux. Les gestes constituent un moyen majeur et important dans la communication humaine. En effet, [Pei, 1984] a énuméré sept cent mille signaux de communication non verbale incluant cinquante mille deux cents expressions faciales [Birdwhistell, 1963] et cinq mille gestes manuels [Krout, 1935]. Cependant, la signification d'un geste diffère d'une culture à une autre : il n'existe pas de signification universelle et invariable d'un geste. Par exemple, pointer avec un doit étendu est un geste ordinaire dans l'Europe et les Etats Unies mais considéré comme un geste offensif en Asie. Cela implique que l'interprétation sémantique d'un geste dépend de la culture. En plus, un geste peut dépendre d'un état individuel : par exemple, les gestes de la main sont synchronisés et co-expressifs avec la parole et les expressions faciales qui reflètent l'humeur individuel. Selon [Hall, 1973], quand deux personnes engagent une discussion, 35% de leur communication est verbale et 65% non verbale. On peut catégoriser la communication non verbale en sept classes [Hall, 1973] :

1. La langue du corps : les expressions faciales, les postures, le regard (la durée du regard, la fréquence du clignement, le contact visuel), les gestes, l'attitude.

2. L'apparence : les vêtements, les effets personnels (les accessoires, les lunettes).

3. La voix : le ton, l'intonation, le volume, le silence, le rire.

4. L'espace et la distance : les proxémités et catégories de comportement proxémique (utilisation de l'espace par les personnes dans leurs comportements, et des significations qui s'en dégagent).

5. Les couleurs : des couleurs froides ou chaudes, l'interprétation des couleurs.

6. Les chronemiques (lié au temps) : la ponctualité, la précipitation du discours, la volonté de parler, la volonté d'écouter.

Par ailleurs, les gestes qui entreraient dans la « langue du corps »peuvent aussi être catégorisés. Par exemple, [Ottenheimer, 2012] distingue cinq types de gestes :

1. Les emblèmes : un emblème (ou le geste emblématique) est un geste qui peut être traduit directement en une communication verbale courte, comme « au revoir », afin de remplacer les mots. Ces gestes sont très liés à la culture.

2. Les illustrateurs : un illustrateur est un geste qui décrit ce que le communicateur dit verbalement (par exemple illustrer une action de lancement en prononçant l'expression « il a jeté »). Ces gestes sont inhérents aux pensées du communicateur et à son discours. Ces gestes sont aussi appelés des gesticulations, ils peuvent être classifiés en cinq sous-catégories comme proposé par [McNeill, 1992] :

   - Les battements : rythmes et raccourcis souvent répétitifs (courts et rapides) de la main ou des doigts.
   - Les gestes déictiques : les gestes de pointage qui peuvent être soit concrets (pointer sur un emplacement réel, un objet ou une personne) ou abstraits (pointer sur un emplacement abstrait, une période de temps).
   - Les gestes iconiques : il consiste en des mouvements manuels qui décrivent une représentation figurative ou une action (lever la main vers le haut en tortillant les doigts pour décrire grimper un arbre).
   - Les gestes métaphoriques : des gestes représentant des abstractions.
   - Les gestes cohérents : ils sont des gestes thématiquement connexes, mais temporellement séparés généralement, en raison d'une interruption de la communication en cours par une autre personne.

3. Les présentateurs des sentiments : un présentateur de sentiments est un geste qui transmet l'émotion ou les intentions du communicateur (par exemple si le communicateur est embarrassé). Ce type de geste dépend moins de la culture.

4. Les régulateurs : un régulateur est un geste qui contrôle l'interaction (par exemple pour contrôler la prise de tour, dans une conversation).

5. Les adaptateurs : un adaptateur est un geste qui permet la libération de la tension du corps (par exemple trembler la tête, déplacer rapidement une jambe). Ces gestes ne sont pas utilisés intentionnellement pendant une communication ou une interaction : ils ont

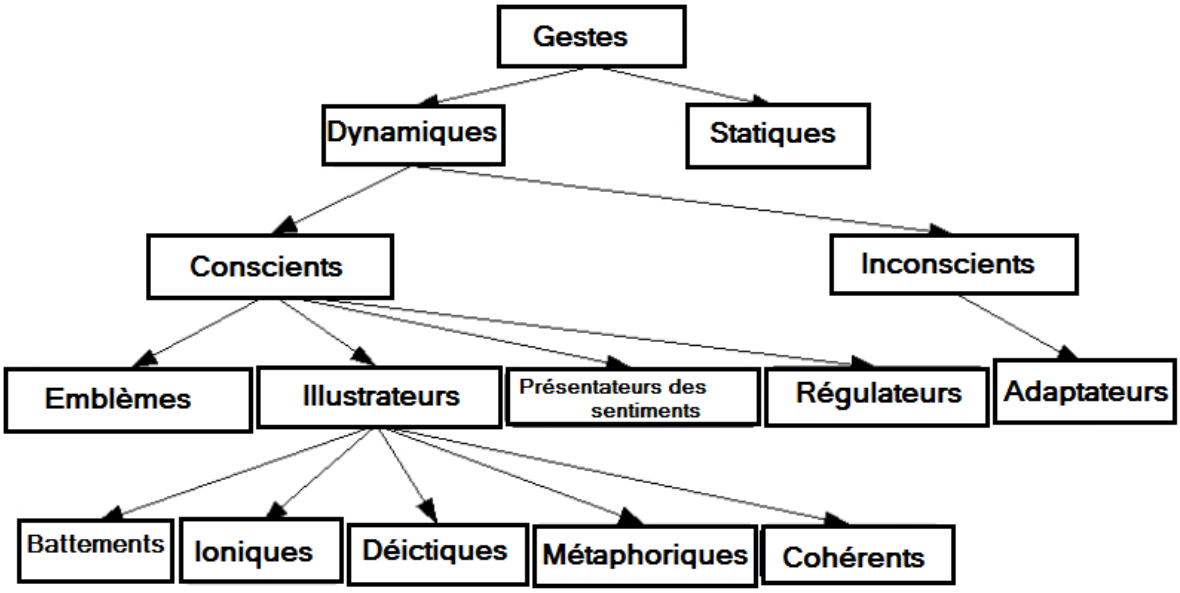

FIGURE 2.1 – Une taxonomie des catégories du geste.

été, à un moment donné, utilisés pour le confort personnel et se sont transformés en une habitude.

Un geste peut être conscient (intentionnel) ou bien inconscient (réflexe). En plus, un geste peut être dynamique ou statique. Ce dernier devient alors une posture. Enfin, on peut classifier les gestes en fonction des parties du corps impliquées dans le geste : (1) gestes manuels, (2) gestes faciaux et (3) gestes corporels. La figure 2.1 illustre une taxonomie des catégories de geste qui reprend tout les critères précédemment cités sauf le dernier. Ceci n'est pas la façon unique de classifier les gestes. En effet, on peut en trouver beaucoup d'autres dans la littérature. Cependant, nous pensons que la taxonomie présentée ici représente presque tous les aspects du geste. Dans notre travail, nous nous concentrons sur les gestes dynamiques de la partie supérieure du corps, et on en vise deux catégories : les gestes simples et composés.

2.1.2 Les technologies

Dans cette sous section, nous étudions les technologies les plus utilisées dans la reconnaissance de gestes. Comme vu précédemment, il y a deux types d'outils : (1) les outils basés sur le contact et (2) les outils basés sur la vision. Ci-après, nous discutons les deux types de dispositif.

La technologie basée sur le contact

Les outils basés sur le contact sont divers : les accéléromètres, l'écran tactile, les gants... Quelques dispositifs, comme l'IPhone (Apple), incluent plusieurs détecteurs : par exemple, un écran tactile et un accéléromètre. D'autres dispositifs utilisent un seul détecteur : par exemple, les accéléromètres de Nintendo (Wii-mote). Par conséquent, nous pouvons classer ces dispositifs en cinq sous-catégories :

- Mécanique : Immersion propose le "CyberGlove II", une paire de gants sans-fil utilisés pour la reconnaissance de gestes de la main. Animazoo propose un costume de corps appelé "IGS-190" pour capturer des gestes du corps (voir figure 2.2). Ce type de dispositif est d'habitude utilisée en collaboration avec d'autres dispositifs. Par exemple, [Kevin et al., 2004] présente une méthode pour la modélisation des trajectoires dans la reconnaissance de gestes qui utilise des CyberGloves avec des marqueurs magnétiques. De même, le costume de corps "IGS-190" est combiné avec dix-huit dispositifs inertiels (gyroscopes) qui permettent la détection de mouvement.

- Inertiel : ces dispositifs mesurent la variation du champ magnétique de la terre pour détecter le mouvement. Deux types de dispositifs sont disponibles : les accéléromètres (par exemple Wii-mote) et gyroscopes (par exemple, IGS-190). [Schlömer et al., 2008] proposent de reconnaître des gestes avec un wii-controler indépendamment du système cible en utilisant les Modèles de Markov Cachés (HMM). L'utilisateur peut apprendre des gestes personnalisés pour la navigation médiatique intuitive multimodale. [Noury et al., 2003] et [Bourke et al., 2007] proposent de détecter des chutes parmi des gestes normaux en utilisant des accéléromètres.

- Tactile : les écrans tactiles sont devenus communs dans notre vie (par exemple : la tablette, iPhone). [Webel et al., 2008] proposent de reconnaître les interactions gestuelles avec les écrans tactiles en utilisant les HMM.

- Magnétique : ces dispositifs mesurent les variations d'un champs magnétique artificiel pour la détection du mouvement. A l'inverse des dispositifs inertiels, les dispositifs magnétiques ont quelques influences sur la santé à cause du champs électromagnétique artificiel.

- Ultrasonique : les traqueurs de mouvement dans ce cas sont composés de trois sortes de dispositifs : (1) les émetteurs soniques qui émettent les ultrasons, (2) les disques soniques qui reflètent les ultrasons (connectés à l'utilisateur) et (3) des multiples capteurs qui

FIGURE 2.2 – L'Animazoo IGS-190 contient dix-huit gyros pour la reconnaissance de geste en plus des marqueurs visuels en jaune.

calculent le temps de retour de l'impulsion. La position est calculée en fonction du temps de réflexion et de la vitesse du son. L'orientation est ensuite triangulée. Ces dispositifs ne sont pas précis et ont une faible résolution mais sont utiles dans un environnement qui manque de lumière et présente des obstacles magnétiques ou du bruit.

La technologie basée sur la vision

Les systèmes de reconnaissance de geste basés sur la vision reposent sur une ou plusieurs caméras pour analyser et interpréter le mouvement à partir des séquences vidéos. De même que les dispositifs de contact, les dispositifs basés sur la vision sont divers. Par exemple, nous pouvons distinguer les capteurs suivants :

- Les caméras infrarouges : typiquement utilisées pour la vision nocturne, les caméras infrarouges donnent généralement une vue fragile de la silhouette humaine.
- Les caméras monoculaires traditionnelles : ce sont les plus communes en raison de leur faible coût. Une variante spécifique peut être utilisée comme les caméras panoramiques pour la vision grand angle ou les caméras time-of-flight pour connaître la profondeur.
- Les stéréo-cameras : la stéréo-vision livre directement des informations 3D par l'intégration du processus de triangulation.

Les avantages et les inconvénients des deux technologies

Les deux types de technologies citées auparavant ont des avantages et des inconvénients. Ainsi, les dispositifs de contact exigent la coopération de l'utilisateur et peuvent être inconfortables surtout si la durée d'utilisation est longue, cependant ils sont précis. Les dispositifs

basés sur la vision n'exigent pas la coopération de l'utilisateur mais ils sont plus difficiles à configurer et présentent des problèmes d'occultation. La table 2.1 résume les principaux avantages et inconvénients des deux technologies. Les dispositifs de contact sont plus précis sauf les ultrasoniques. Aussi, ils n'ont pas généralement de problèmes d'occultation sauf les capteurs magnétiques (obstacles métalliques) et les capteurs ultrasoniques (obstacles mécaniques). Par ailleurs, certains dispositifs de contact peuvent causer quelques problèmes de santé : par exemple une allergie au matériel du capteur mécanique, le risque d'un cancer pour des dispositifs magnétiques.

TABLE 2.1 – Comparaison entre les dispositifs de contact et les dispositif de la vision.

Critère	Dispositif de contact	Dispositif de la vision
Coopération de l'utilisateur	Oui	Non
Dérange l'utilisateur	Oui	Non
Précision	Oui/Non	Non/Oui
Flexibilité de la configuration	Oui	Non
Flexibilité de l'utilisation	Non	Oui
Problème de l'occultation	Non (Oui)	Oui
Problèmes de santé	Oui (Non)	Non

2.2 L'extraction des caractéristiques

Les techniques d'extraction des caractéristiques visent à collecter des données concernant la position du geste, son orientation, et sa progression temporelle. L'extraction des caractéristiques repose sur des traitements et analyses de données de bas niveau, qui sont nécessaires pour produire des données de haut niveau comme les contours. Théoriquement, l'extraction des caractéristiques vise à réduire la dimension des données par un codage des informations dans une représentation compressée et à supprimer des données moins discriminantes. Évidemment, ces informations liées ou distinctives diffèrent selon l'objet d'intérêt et les objectifs de l'application. L'extraction des caractéristiques est essentielle pour la reconnaissance de gestes. Par conséquent, le choix des caractéristiques et de la méthode d'extraction n'est pas anondin, il influence grandement les performances de la reconnaissance de gestes. Des méthodes à base d'apparence dépendent de l'enregistrement direct de gestes avec des caractéristiques d'image 2D. Les caractéristiques d'image les plus utilisées pour détecter les mains et reconnaître les gestes sont la couleur de la peau [Wu and Huang, 2002] [Mckenna and Morrison, 2004] [Yao and Cooperstock, 2002] [Bretzner et al., 2002], les formes [Ramamoorthy et al., 2003] [Ong and Bowden, 2004] [Chen et al., 2003], les caractéristiques locales de la main [Oka et al., 2002] [Zhang et al., 2001] [Malik et al., 2002]

le flux optique [Cutler and Turk, 1998] [Lu et al., 2003]. L'information de la couleur a été largement utilisée dans l'extraction des caractéristiques [Sigal et al., 2004] [Yao et al., 2004]. Cependant, la texture, la luminosité et le mouvement (définis par deux ou plusieurs images) ont été aussi abordés [Wu et al., 2000] [Cutler and Turk, 1998].

2.2.1 La profondeur

L'information de profondeur à partir de deux caméras calibrées [Rauschert et al., 2002], ou de capteurs qui fournissent directement la profondeur comme le LiDAR (Light Detection and Ranging) ou encore le Microsoft Kinect [Takahashi, 2009], est une bonne indication quand la personne se trouve en face de la caméra. Cependant, la profondeur fournie par une caméra stéréo est relativement bruitée et donc souvent combinée avec d'autre indices comme la couleur [Grange, 2007] [Nickel and Stiefelhagen, 2007].

2.2.2 La couleur

La couleur de la peau [Bretzner et al., 2002] [Mckenna and Morrison, 2004] [Imagawa et al., 2000] est une caractéristique importante qu'on peut extraire d'une image afin de détecter et suivre les mains d'une personne. Cependant, les méthodes basées sur la couleur sont généralement sensibles et ont du mal à différencier la main des autres objets dont la couleur est similaire à celle de la main, comme le visage ou bien le bras. Pour résoudre ce problème, les personnes sont obligées de porter des chemises à manches longues ce qui devient intrusif.
Les travaux de l'état de l'art utilisent différents seuils de la couleur de la peau dans différents espaces de couleur pour détecter les visages et les mains dans les images. Dans [Terrillon and Akamatsu, 1999], une comparaison de performance a été faite entre les espaces de couleurs, à savoir HSV/HSI, RGB, T-S, TSL et YCbCr. D'après leurs résultats, l'espace T-S normalisé donne une meilleure segmentation et une détection robuste de la tête.

2.2.3 Modèle de frontière elliptique

L'article [Xu et al., 2011] propose une méthode qui permet de localiser plusieurs visages et mains dans une scène en utilisant la détection de la couleur de la peau avec un modèle de frontière elliptique afin d'identifier les domaines possibles de la couleur de peau humaine. Le

modèle de contour elliptique est défini par :

$$\phi(x) = [x - \psi]^T \Lambda^{-1} [x - \psi] \tag{2.1}$$

Où x est le vecteur de couleur, les autres paramètres sont données par :

$$\psi = \frac{1}{n} \Sigma_{i=1}^{n} x_i \tag{2.2}$$

et

$$\Lambda = \frac{1}{N} \Sigma_{i=1}^{n} f_i (x_i - u)(x_i - u)^T \tag{2.3}$$

Avec $N = \Sigma_{i=1}^{n} f_i$: le nombre total des échantillons dans l'ensemble des données d'apprentissage.
$f_i = f(x_i)(i = 1, 2, ..., n)$: le nombre d' échantillons.
$u = \frac{1}{N} \Sigma_{i=1}^{n} f_i x_i$: la moyenne du vecteur de chrominance dans l'ensemble des données d'apprentissage.
Le modèle elliptique de couleur de peau obtient des taux élevés de détection de la couleur de peau. Afin de classer chaque pixel dans les images couleur, on définit une valeur de seuil Θ choisie par un compromis entre les vrais et les faux positifs. Si $\phi(x) < \Theta$, le pixel est identifié comme couleur de peau et vice versa. Avant d'utiliser le modèle de frontière elliptique, le corps humain est segmenté à partir de l'arrière plan. La procédure de la segmentation est donnée par les étapes suivantes : Une fois l'image en RGB et l'image en profondeur obtenues via la Kinect, l'image RGB est redimensionnée pour avoir une résolution similaire à celle de l'image de profondeur. Ensuite, l'algorithme « floor plan clip »de détection de la personne fourni par la Kinect est utilisé pour segmenter le corps humain en supprimant l'arrière plan. Lorsque le corps humain est segmenté, le modèle de contour elliptique est utilisé pour détecter la couleur de la peau. Ce dernier processus permet d'obtenir 3 blocs qui correspondent aux deux mains et au visage. Ensuite, l'algorithme des K-means est utilisé pour avoir le centre de chaque bloc afin de localiser le visage et les mains. Finalement, les mains et le visage peuvent être différenciés grâce à la valeur de profondeur et les coordonnées des centres des blocs. L' avantage de cette approche est la localisation du visage et des mains de plusieurs personnes dans une même scène même si l'environnement est complexe. L' inconvénient de cette méthode est dans l'utilisation de deux étapes de segmentation : la segmentation de la personne à partir de l'arrière plan et ensuite la segmentation des mains et du visage à partir des autres parties du corps. En effet, les auteurs utilisent l'espace de couleur $YCbCr$ qui ne fait pas la distinction des objets ayant

la même couleur que la peau, ce qui exige une première segmentation du corps.

2.2.4 Combinaison des indices

Ceci consiste à combiner plusieurs caractéristiques afin d'améliorer les performances. Plusieurs méthodes ont combiné plusieurs indices afin d'avoir plus d'informations. La détection des mains, du visage ou du corps en se basant sur les apparences, combinée avec la détection du mouvement basée sur les régions d'intérêt, peut augmenter la précision et diminuer la vitesse d'exécution. Dans [Rauschert et al., 2002], la détection de la couleur combinée avec le mouvement a été appliquée pour les interfaces tangibles utilisant les gestes de la main.
Dans [Yilmaz et al., 2004] les auteurs proposent une méthode générique de suivi d'objets appliquée au suivi des personnes. La méthode utilise une représentation de la personne par contour, couleur et texture. Les contours sont définis par des *levels set* et définissent un modèle de personne (le contour moyen et sa variance) selon son activité. La couleur est ici représentée par un modèle a priori dont la densité est estimée par un noyau multivarié. Le noyau choisi est celui d'Epanechnikov. Il a la propriété de fournir l'erreur minimum entre les données et l'estimation. La texture est modélisée par un mélange de deux gaussiennes des caractéristiques extraites par des filtres orientables (steerable filters). La fusion des informations de couleur et de texture fournit un modèle statistique semi-paramétrique. L'apport de cette méthode est de définir une énergie d'évolution du contour en fonction de la couleur et de la texture, du fond de scène et de la forme.

2.2.5 Fusion profondeur/couleur

L'article [Moreno et al., 2001] propose une méthode de localisation du visage qui fusionne des informations acquises par la caméra stéréo et l'analyse des images couleur. Tout d'abord, l'image de profondeur est déterminée. Ensuite, les régions de la peau sont extraites en utilisant un algorithme de classification des régions de la couleur de la peau. Cet algorithme est basé sur le calcul d'histogrammes de couleur et la recherche d'une correspondance avec une table de hachage constituée d'une version sous échantillonnée de cet histogramme. Afin de détecter la tête dans les images, les auteurs ont créé un modèle de la tête et du cou qui consiste en deux caractéristiques ; la forme de la tête dont la taille peut se modifier selon la distance de profondeur dans la scène, et la couleur de la peau. Une fois la carte de profondeur est calculée et la segmentation en région de la couleur de la peau est obtenue, la carte de profondeur originale est filtrée avec l'information de la couleur. La recherche du modèle proposé est effectuée avec

différentes valeurs de profondeur en commençant par les régions éloignées et en s'approchant de la caméra. En prenant en compte dans cette recherche la limitation de l'intervalle à analyser, à partir de la carte de profondeur réduite, à $400mm$ dans chaque intervalle de profondeur. Ceci est effectué par le filtrage de la carte réduite en utilisant la matrice de projection appropriée. Faire après, une re-projection du modèle de la tête sur cet intervalle de sorte que la forme de la tête et la taille de la fenêtre de recherche soient modifiés dynamiquement en passant d'un intervalle de profondeur à un autre. Pour chaque intervalle de recherche choisi un point de correspondance est calculé avec l'équation de corrélation normalisée suivante :

$$C_{u,v} = \frac{\Sigma IM - \Sigma I \Sigma M}{\sqrt{(\Sigma I^2) - (\Sigma I)^2)(\Sigma M^2 - (\Sigma M^2))}} \tag{2.4}$$

Avec I la carte de profondeur réduite et binarisée $I(u+i, v+j)$, M indique la position du modèle de projection $M(i,j)$, et toutes les sommations ont lieu dans l'intervalle (i,j) qui appartient à la fenêtre de recherche W modifiée dynamiquement. La région supposée représenter la tête correspond à l'intervalle qui contient le plus grand nombre de points de correspondance. L'avantage de cette approche est qu'elle permet de bien définir la région de la tête même si la carte de profondeur initiale contient de fausses correspondances. L'inconvénient est que la variabilité des modules de bas niveau, les formats de données ainsi que les niveaux de bruit qu'ils produisent rendent la fusion une tâche difficile.

2.2.6 Les points d'interêt

Récemment, des recherches se sont concentrées sur la détection de points d'interêt tels que la détection des coins, les jonctions en T où bien les points à forte variation. Ces points correspondent à des doubles discontinuités de la fonction d'intensité. Ils représentent une source d'information plus fiables que les contours car ils ont plus de contraintes sur la fonction d'intensité. Ils sont robustes aux occultations et sont présents dans une grande majorité d'images. Le détecteur des points d'intérêts le plus connu et SIFT (Scale Invariant Feature Transform), qui a été présenté dans [Lowe, 2004]. Il permet d'extraire les caractéristiques invariantes distinctives, à partir des images, qui peuvent être utilisées pour effectuer la correspondance entre des vues diverses d'un objet ou d'une scène. Un nouveau détecteur a été développé dans [Plagemann et al., 2010] pour un nouveau type de points d'intérêt nommés AJEX (Accumulative Geodesic EXtrema), qui sont calculés via la maximisation d'une manière progressive des

distances géodésiques sur une maille de surface. L'approche permet d'estimer l'orientation 3D d'un vecteur de points d'intérêt donné. Le but est d'extraire les informations sur les parties dur corps humain ; comme la localisation et l'orientation dans l'espace à partir des images de profondeur prises via la caméra temps de vol. Les auteurs ont identifié deux ensembles de points $AGEX$. Le premièr ensemble $AGEX_1(M)$ des points AGEX est initialisé avec le centre de gravité géodésique d'une maille. L'ensemble des points d'intérêt $AGEX_k(M)$ est calculé en utilisant l'algorithme de Djikstra qui a pour but de trouver le plus court chemin. Le sommet qui possède le plus long de ces court chemins trouvés est fixé afin de produire l'ensemble $AGEX_2(M)$. Les points d'intérêt fournis par cet algorithme sont considérés comme l'espace des hypothèses pour les emplacements possibles des parties du corps.

L'étape suivante consiste à estimer l'orientation des points d'intérêt. À chaque point d'intérêt k_i est assigné l'orientation qui donne le court chemin menant dans sa direction. L'identification des parties du corps est obtenue grâce à la classification des descripteurs des patch locaux. Des étiquettes désignant les parties du corps sont alors assignées aux points d'intérêt. Les auteurs ont pris comme descripteurs locaux des patchs d'image de profondeur de 41x41, entourant le point d'intérêt.

Le résultat donne un ensemble de patchs de l'image susceptibles d'être centrée sur les parties saillantes du corps humain. Afin d'attribuer des étiquettes désignant les parties du corps aux descripteurs de patchs, une approche d'apprentissage supervisée est utilisée. L'avantage de l'approche est qu'elle permet de bien définir les parties du corps. L'inconvénient est qu'elle nécessite un ensemble d'apprentissage des patchs centrés sur les pièces de points d'intérêt et aussi une grand quantité des patchs de l'arrière plan ou d'autres parties n'étant pas d'intérêt. De plus, le système d'acquisition utilisé est composé d'un capteur de mouvements basé sur les marqueurs.

2.2.7 Le sac des caractéristiques

La technique dite "Sac de caractéristiques" est une méthode qui représente les images comme étant des groupes désordonnés de caractéristiques locales. Récemment, les représentations ≪ Sac de caractéristiques ≫ont montré de bonnes performances pour la reconnaissance d' actions [Jhuang et al., 2007, Niebles et al., 2008, Schuldt et al., 2004]. Elles permettent la reconnaissance d'un grand nombre d'actions allant de mouvements basiques (courir) jusqu'aux interactions (serrer les mains) [Gilbert et al., 2009, Laptev et al., 2008, Marszalek et al., 2009, Yeffet and Wolf, 2009]. Cependant, les approches basées sur les ≪ Sac de caractéristiques ≫reposent sur des caractéristiques

locales des mouvements. Elles ne prennent pas en compte les relations entre le domaines spatial et temporel, malgré leur intérêt pour la reconnaissance.

2.3 La détection

La détection de la personne consiste à détecter une où plusieurs personnes et à connaître sa localisation précise dans une image numérique. C'est un sujet assez difficile dû à la variété d'apparences des personnes et de leurs vêtements, de la déformabilité du corps humain (corps articulé : bras, jambes, torse) et des phénomènes d'occultations. La segmentation est un prétraitement primordial pour la reconnaissance. Elle consiste à éliminer les régions dans l'image qui ne correspondent pas à l'objet d'intérêt. La plupart des méthodes proposées sont basées sur les caractéristiques visuels du corps humain. La couleur de la peau, la forme de la tête sont des indices très importants dans la détection de la présence d'une personne dans une scène.

2.3.1 K-means Expectation Maximization

S.E. Ghobadi et al. [Ghobadi et al., 2007] proposent une méthode de segmentation basée sur la fusion des images 2D/3D pour la reconnaissance des gestes. La technique de segmentation est basée sur la combinaison de deux approches de clustering : K-Means et Espérance-Maximisation. Le capteur utilisé est la caméra Time-of-Flight. K-means est une méthode de segmentation basée sur le clustering, une technique de classification non supervisée qui permet de segmenter les données en mettant celles ayant les mêmes caractéristiques dans le même groupe (appelé cluster).

Cependant, la fonction d'appartenance de K-means est très sensible, un petit décalage peut affecter un échantillon à un mauvais cluster. Une solution à ce problème a été proposée, elle consiste à remplacer l'étape clustering à base de K-means par l'affectation probabiliste de la technique EM étant donné que cette dernière n'impose pas une frontière stricte entre les clusters. Ainsi, un point de données sera affecté à chaque cluster avec une certaine probabilité. Cependant, EM peut générer de mauvais clustering si les paramètres ne sont pas proprement initialisés. Afin de palier ce problème, les auteurs ont développé l'algorithme KEM qui est une combinaison de ces deux techniques. Il s'agit de trouver d'abord les centres initiaux des clusters avec K-means. Ensuite, utiliser ces centres comme paramètres initiaux d'EM et itérer jusqu'à ce que le minimum local soit trouvé. Cette méthode a été utilisée avec succès pour la segmentation

de la main dans des images de scènes simples et complexes (encombrées).

La segmentation de la main a été effectuée avec succès dans des cas différents : La position du geste dans le premier plan dans une scène simple, la position du geste dans le premier plan dans une scène complexe, et une séquence du geste en allant du premier plan vers l'arrière plan.

2.3.2 Mean Shift 6D

Le but de la segmentation est de classifier les pixels d'une image sous forme d'objets distincts. Or, la segmentation basée sur la couleur ne marche pas dans le cas où les objets sont occultés, ou bien quand les objets partagent la même couleur avec l'arrière plan. Pour palier ce problème, Amit B. et al. [Bleiweiss and Werman, 2009] proposent une nouvelle version de l' algorithme Mean shift en ajoutant l'information de profondeur à la version native de l'algorithme. La procédure Mean Shift classique s'effectue dans l'espace des caractéristiques 5D : La couleur convertie dans l'espace des couleurs $L * u * v$ et les coordonnées du maillage en 2D. Les auteurs ont changé l'algorithme en ajoutant la profondeur comme dimension dans l'espace des caractéristiques ce qui donne un vecteur Mean Shift 6D.
Le calcul du vecteur Mean Shift 6D permet de mettre à l'échelle les poids initiaux et ajoute un facteur d'échelle supplémentaire σ afin de donner plus de poids aux données de profondeur : $w_6(x) = \frac{1}{W(x)+1} w_6(x)\sigma$ avec w_6 est le poids appliqué sur la composante de profondeur. Dans la pratique, une valeur de σ entre 2 et 5 donne de bons résultats. Ainsi, Mean Shift va s'appuyer plus sur l'information de la couleur dans les régions où la profondeur est bruitée.
L'histogramme utilisé est basé sur les données de couleur et de profondeur. Chaque canal est divisé par 16, ce qui donne un nombre total de 16^4 classes (bins). Le traitement se fait comme suit ; Si le nombre de pixels de profondeur dans la carte des poids dépasse un certain seuil, alors seulement 16^3 classes seront utilisés et les données de profondeur seront supprimées. Les auteurs traitent aussi les cas extrêmes : le cas de la sortie du cadre de la caméra, et le cas de l'absence de lumière. L'idée est de calculer à chaque itération les valeurs des pixels pour les données de couleurs et de profondeur par les équations suivantes :

$$S_{rgb} = \Sigma_{i=1}^{n_i}(R(x_i) + G(x_i) + B(x_i)) \tag{2.5}$$

et

$$S_{depth} = \Sigma_{i=1}^{n_1} D(x_i) \tag{2.6}$$

Dans le premier cas la valeur S_{depth} tend vers 0. Alors, seule la donnée RGB est utilisée dans l'histogramme. Cependant, dans le deuxième cas la valeur S_{rgb} tend vers 0 et on ne considère que la valeur de profondeur.

2.3.3 Modèle 2D/3D de la tête

Lu X. et al. [Xia et al., 2011] proposent une méthode de détection de la personne en utilisant la Kinect Xbox360. La première étape de la méthode consiste à utiliser l'information des contours. Pour ce faire, la technique « Canny edge Detector »est utilisée pour trouver les bords dans la carte de profondeur et calculer l'image binaire des contours. Les bords dont les tailles sont petites (inférieures à un seuil) sont éliminés. Un modèle binaire de la tête est utilisé pour réaliser la correspondance avec l'image des contours. Une carte de distance est alors calculée à partir de l'image de bord, où les pixels correspondent aux distances entre l'image des contours et le modèle. La correspondance consiste à positionner le modèle dans différents endroits de la carte de distances ; la mesure de correspondance est déterminée par les valeurs de pixel de l'image de distance. La région dont les valeurs sont plus petites indique la mise en correspondance entre le modèle et l'image. Si la valeur de la distance est inférieure à un certain seuil, alors l'objet cible sera considéré comme détecté, ce qui signifie qu'un objet similaire à la tête d'une personne se trouve à cet endroit.

Afin d'examiner toutes les régions détectées, les paramètres de la tête sont calculés. Si on considère que la forme de la tête est un cercle, alors ces paramètres seront la hauteur et le rayon de la tête. Les auteurs proposent un modèle 3D de la tête basé sur une hémisphère. Enfin, ils se basent sur une méthode de seuillage pour décider quelle région correspond vraiment à la tête parmi toutes les régions détectées. L'avantage de cette approche est qu'elle n'a pas besoin d'une base de données d'apprentissage.

2.4 Le suivi (Tracking)

Le suivi de personnes consiste à détecter puis mettre en correspondre, dans des images successives, des régions qui correspondent à la personne. C'est un domaine de recherche dynamique de la communauté de vision par ordinateur. Le suivi est une tâche très importante pour la reconnaissance de gestes. Il est généralement basé sur la cohérence du modèle de mouvement au cours du temps. Le mouvement peut être local (par exemple : mouvement des points caractéristiques, mouvement des parties du corps) ou bien global (par exemple : mouvement de

tout le corps). L'extraction et l'analyse des caractéristiques des mouvements permettent ensuite la reconnaissance des gestes. Une fois le mouvement du corps ou de ses parties est détecté, des traitements sont faits pour identifier le type du mouvement effectué, on parle alors de l'étape d'analyse du mouvement. Cette analyse peut être ensuite utilisée par différents algorithmes de suivi et de reconnaissance de gestes.

Dans [Jalal et al., 2015] les auteurs ont proposé une méthode de suivi basée sur les silhouettes en profondeur. Les images de profondeur sont traitées pour identifier des silhouettes humaines à partir d'un arrière plan bruité en utilisant la technique de soustraction de l'arrière plan. Cependant, un ensemble de techniques de représentation de caractéristiques, comme les informations temporelles des images, l'histoire de profondeur et les informations de différence de mouvement sont tirées pour augmenter les caractéristiques de profondeur spatiales et temporelles de silhouettes. A partir d'une séquence d'images en profondeur, les auteurs extraient la différence d'intensité de mouvement des silhouettes humaines d_f en considérant la différence entre deux images consécutives t et $t-1$ comme suit :

$$d_f = \left| f_t^i - f_t^i - 1 \right| \tag{2.7}$$

2.4.1 Les coordonnées 3D

Lu X. et al. [Xia et al., 2011] ont aussi développé un algorithme de suivi basé sur le mouvement des objets. Ils supposent que les coordonnées et la vitesse des mêmes objets dans des images voisines changent lentement. C'est à dire qu'il ne devrait pas y avoir de grands sauts dans les coordonnées ou la vitesse. L'algorithme commence par calculer le centre du blob détecté. Ensuite, il calcule les coordonnées 3D et la vitesse des personnes dans chaque image. Les coordonnées sont déterminées directement à partir de la matrice de profondeur. La vitesse est calculée à partir des coordonnées des points dans les images voisines. Un score d'énergie des variations des coordonnées dans l'espace des coordonnées et de la vitesse est alors défini par :

$$E = (c - c_0)^2 + (v - v_0)^2 \tag{2.8}$$

Avec, E : le score de l'énergie, c : les coordonnées de la personne dans l'image courante, c_0 : les coordonnées de la personne dans l'image précédente, v : la vitesse de la personne dans l'image courante et v_0 : la vitesse de la personne dans l'image précédente. Dans la première image, à chaque personne détectée est associé un label selon l'ordre de détection. Au cours des

images suivantes, toutes les possibilités de mise en correspondance entre les personnes détectées sont testées, celle qui minimise le score d'énergie correspond alors à la solution recherchée. L'inconvénient est que l'approche ne traite pas l'apparition/disparition d'une personne dans la scène.

2.4.2 Fusion profondeur/silhouette

Daniel G. et al. [Grest et al., 2007] proposent une approche de suivi des mouvements à partir d'une seule vue via la combinaison de la profondeur et la silhouette. Les correspondances entre le modèle de la silhouette projeté et l'image réelle sont établies d'une nouvelle manière permettant de gérer les arrières plans encombrés. La pose est estimée par la méthode des moindres carrés non linéaires. Le processus d'estimation est basé sur un modèle du corps défini dans le standard MPEG4. Ce modèle possède 180 degrés de liberté. Il est formé par une combinaison de chaînes cinématiques. Le mouvement d'un point (par exemple : le centre de la main) peut être estimé par une concaténation de rotations. Lorsque les axes de rotations sont connus, par exemple la flexion du coude, alors la rotation n'a qu'un seul degré de liberté c'est-à-dire l'angle autour de cet axe. En plus des angles des articulations, il y a 6 degrés de liberté pour la position et l'orientation de l'objet exprimées en coordonnées universelles globales. L'information de la silhouette est ajoutée à l'estimation. Pour ce faire, les auteurs calculent les correspondances 2D/3D de la silhouette du modèle et celle de la personne réelle. Le but est d'utiliser le modèle de silhouette prédit (predicted model silhouette) pour chercher les points de correspondance dans la silhouette réelle. Quand la pose initiale est connue, il est possible de calculer l'histogramme de couleur pour chaque segment du corps. L'information de la silhouette est intégrée dans la fonction de coût, telle que la distance de la projection du point 3D sur la ligne 2D soit minimale. Pour obtenir la pose initiale, l'utilisateur doit positionner le modèle manuellement dans un voisinage proche de la position dans l'image courante. Après quelques itérations de l'ICP (Iterative Closest Point) la pose initiale correcte est déterminée. L'image de profondeur est modifiée afin d'éliminer les valeurs de profondeur erronées entre l'avant et l'arrière-plan. Afin de réduire l'influence des points aberrants un filtre de variance est appliqué à l'image de profondeur, il calcule la variance à l'intérieur d'une fenêtre $3 * 3$ et met à zéro tous les pixels dont l'écart est supérieur à un seuil.

2.4.3 Unscented Kalman Filter

Le suivi d'un objet en utilisant le filtrage se fait en deux étapes : d'abord, la prédiction qui est l'utilisation de l'état estimé à l'instant précédent pour produire l'estimation à l'instant

courant. Ensuite, la mise à jour des informations en fonction des nouvelles mesures.
L' UKF utilise une méthode d'échantillonnage déterministe pour capturer les covariances moyennes et les estimations avec un ensemble minimal de points d'échantillonnage. L' UKF est une technique d'estimation non linéaire puissante, elle a été développée pour remplacer la technique Extended Kalman Filter dans différentes applications, y compris le suivi. A. Boesen et al. [Larsen et al., 2011] présentent un système de suivi articulé en utilisant les données de profondeur extraites d'une caméra stéréo et un UKF qui permet d'améliorer la qualité du suivi. Pour cela, deux modèles ont été proposés :

- Le modèle d'état : Les auteurs ont utilisés un modèle articulé du corps humain à partir d'un squelette cinématique composé de parties rigides connectées grâce à des articulations ayant jusqu'à 3 degrés de liberté selon le type d'articulation. L'ensemble des angles des articulations du squelette cinématique constitue le vecteur d'état. Le suivi est effectué sur la partie supérieure du corps en supposant que la personne est debout et ne fait bouger que sa partie supérieure. Les variations des angles sont limitées afin de simuler le comportement d'un squelette humain. L'état initial est initialisé manuellement. Ensuite il est propagé dans le temps en ajoutant un bruit gaussien de moyenne nulle à chaque angle d'articulation, tel que :

  $$p(x_t/x_{t-1}) = N(x_t/x_{t-1}, \sum) \tag{2.9}$$

  avec $\sum$: Matrice diagonale et x_t : l'état x à l'instant t.

- Le modèle d'observation : Pour chaque image, la caméra fournie un ensemble de points 3D. La segmentation est faite par la suppression des points qui sont à une distance supérieure à un seuil donné. Si les points restants contiennent des valeurs aberrantes, alors ils seront transformés et ramenés au membre le plus proche. L'ensemble des points ainsi extraits constitue le vecteur d'observation y. Enfin, une comparaison des deux modèles est effectuée pour estimer la position réelle du corps humain.
  Une comparaison des performances de deux filtre non linéaires, le filtre à particules et l'UFK, a été effectuée. Les deux filtres ont été appliqués sur les mêmes images. Les résultats montrent que l'UFK donne un suivi très précis et souple par rapport au filtre à particules. Cependant, lorsque le nombre d'échantillons est élevé le filtre à particules donne des résultats qui s'approchent de ceux de l' UFK. Dans les cas où les membres du

corps sont proches l'un de l'autre, l'UFK est plus robuste en terme de suivi que le filtre à particules.

2.4.4 Classification de la couleur

Le suivi de la main peut être effectué en se basant sur la couleur de la peau. Ceci peut être accomplit par l'utilisation d'une technique de classification de couleur dans un espace de couleur. Siddarth S. et al. [Rautaray and Agrawal, 2011], Miaolong Y. et al. [Yuan et al., 2008] ont développé une méthode de suivi de la main basée sur une technique de classification de couleur en utilisant l'espace $L * a * b*$. Cette technique consiste en deux étapes : l'apprentissage et le suivi. Dans l'étape de l'apprentissage, une région d'intérêt de la main est spécifiée pour obtenir les données d'apprentissage. Ces derniers seront catégorisées dans un nombre de classes de couleur utilisant la liste randomisée basée sur la couleur de la main. Dans l'étape du suivi, la segmentation de la main à partir de l'arrière plan est faite en temps réel en utilisant la liste randomisée apprise. L'approche de la segmentation de la main est basée sur une classification de la couleur de la peau. La distribution de la couleur de la main est classifiée dans plusieurs clusters de couleur en utilisant une liste aléatoire de données.

Si un signal de couleur d'entrée X est classé dans le cluster i (c'est-à-dire la distance correspondante est inférieure ou égale à un seuil λ) alors le conteur du modèle de ce cluster est incrémenté. Le centre de la région de la main, appelé point d'interaction, est extrait et utilisé comme un dispositif d'entrée dans le système. A partir de la deuxième image, le système acquiert une région limitée autour de la main suivie et répète les mêmes traitements pour la segmenter. Les avantages de cette approche : elle est robuste sous différentes conditions d'éclairage et la segmentation de la main est effectuée avec précision même si l'utilisateur fait des mouvements rapides avec la main.

2.4.5 Filtrage particulaire

Les filtres particulaires, ou bien les méthodes de Monte-Carlo séquentielles, sont des techniques d'estimation de modèles basées sur la simulation. Le but est d'estimer une séquence de paramètres cachés en se basant sur les mesures par une distribution ponctuelle exprimant la sélection d'une valeur dite *particule* avec la probabilité dite *poids*. Mathias et

al. [Fontmarty et al., 2007] présente une approche pour le suivi 3D de mouvements humains basée sur le filtrage particulaire. Le capteur utilisé est un capteur binoculaire. Le modèle de mesure utilisé fusionne les informations d'apparence (couleur, contour) et les information géométriques 3D. Le modèle d'état est constitué de cônes tronqués reliés par des articulations admettant un ou plusieurs degrés de liberté. Le modèle d'observation contient cinq modèles de mesures : les contours, les régions d'intérêt (ROIs), les blobs 3D, les parties de couleur peau et les parties intérieures des membres. Les auteurs ont montré que leur modèle de mesure, en fusionnant à la fois des informations d'apparence et géométriques (3D), s'intègre bien dans le cadre probabilistique des filtres particulaires utilisés. Ainsi, il reste suffisamment discriminant malgré les environnements variés. Le travail de [Migniot and Ababsa, 2013] propose un modèle hybride 2D-3D particulièrement adapté à la vue de dessus. La séparation du modèle proposé permet d'étudier chaque partie du corps dans l'espace où sa forme est le plus descriptive, et de réduire le temps de calcul. Les auteurs proposent une fonction de probabilité avec la distance de chanfrein 2D pour l'estimation de pose de la tête et des épaules. Ainsi qu'une fonction de probabilité avec la distance euclidienne 3D pour l'estimation de pose des bras. Afin de palier au problème des occultations entre les bras des personnes, ils ont réalisé un suivi multi-personnes avec un suiveur par cible.

2.5 La reconnaissance de gestes

Les approches de reconnaissance de gestes à partir de séquences vidéos se concentrent principalement sur le flux optique et l'analyse de l'historique du mouvement. Au cours des deux dernières décennies, de nombreuses techniques ont été developpées. Dans cette section, nous décrivons les méthodes les plus populaires.

2.5.1 L' apprentissage automatique

Dans le cadre de l'interaction homme machine, Xiaoyan W. et Ming X. [Wang et al., 2012] ont développé un système automatique de reconnaissance du geste dynamique de la main. La méthode utilisée pour modèliser et reconnaitre la trajectoire du geste dynamique est basée sur les Modèles de Markov Cachés (HMMs). Tout d'abord, la main est détectée avec l'algorithme de Adaboost puis son contour est suivi. Les auteurs ont utilisé les caractéristiques locales et globales pour représenter la trajectoire de la main. La trajectoire

est ensuite enregistrée et ses caractéristiques sont décrites par un vecteur qui sera utilisé comme une entrée du HMM. Le système proposé peut reconnaître en ligne sept gestes prédéfinis et rejeter les gestes atypiques.

Mahmoud et al. [Elmezain et al., 2009] ont développé un système automatique qui reconnaît le geste isolé et le geste significatif d'un mouvement continu de la main. Le but est de reconnaître les chiffres arabes (0-9) en temps réel basés sur la méthode des modèles de markov cachés (HMM). Afin de gérer les gestes isolés, les topologies Ergodique, Left-Right (LR) et Left-Right Banded (LRB) sont appliquées sur le vecteur de caractéristiques qui est extrait des séquences d'images couleurs stéréos. Les auteurs ont proposé un nouveau système pour reconnaître le geste continu qui relie deux gestes isolés afin de former un geste significatif (la transition entre deux gestes isolés pour former un nombre de deux chiffres).

2.5.2 La classification du geste

Généralement, le problème de classification via une base d'apprentissage est un problème de régression. Classifier c'est « deviner sortie (ou le label du geste) associée à la nouvelle entrée (c'est à dire, nouveau descripteur de gestes généré à partir d'une nouvelle vidéo).

Le classificateur des k plus proches voisins

Dans la littérature, l'algorithme des k plus proches voisins est le classificateur le plus courant. L'idée principale derrière cet algorithme est de sélectionner les k plus proches voisins d'une entrée à partir d'une base d'apprentissage et ensuite l'affecter à la sortie ayant la majorité de votes parmi celles associées aux entrées sélectionnées. L'avantage principale de cet algorithme est qu'il est un approximateur universel et peut bien modéliser n'importe quel mappage plusieurs-à-un. L'inconvénient de cet algorithme est le manque de robustesse dans le cas des espaces de grandes dimensions et la complexité du calcule lors de l'utilisation d'une grande base d'apprentissage. Afin de palier ce problème, les auteurs dans [Kaâniche, 2009] ont utilisé l'analyse en composantes principales ACP pour réduire la dimensionnalité et la méthode K-means pour réduire la taille de la base d'apprentissage. Cependant, un autre problème se pose ; Comment faire face au mappage plusieurs-à-plusieurs (C'est à dire quand un cluster correspond à plusieurs gestes et vice versa) ? Pour

ce cas, les auteurs on proposé un mécanisme de vote qui peut transformer le mappage plusieurs-à-plusieurs en un mappage plusieurs-à-un. Etant donnée la base de données finale, avec le cardinal N, suivante :

$$T = \left\{(c, g)/c \in C \& g \in G \& g \in \ Label^{-1}(c)\right\} \tag{2.10}$$

avec $Card(G) = m$ et $Card(C) = n$. La probabilité $L(c|g)$ d'un cluster c sachant le geste g est défini par l'équation suivante :

$$L(c|g) = P(G = g|C = c) \tag{2.11}$$

Avec C est l'ensemble des clusters, et G est l'ensemble des gestes.

La probabilité du geste g selon les k clusters observés c'_i, $i \in [1..k]$ est donné par :

$$L(g|c'_1, ..., c'_k) = \frac{\sum_{i=1}^{k} L(c'_i|g)}{\sum_h \sum_{i=1}^{k} L(c'_i|h)} \tag{2.12}$$

Cette probabilité satisfait l'équation : $\sum_g L(g|c'_1, ..., c'_k) = 1$ Durant le processus de classification, l'échantillon de test (video) génère plusieurs descripteurs locales de mouvement dlm_i, $i \in [1..M]$. Chaque descripteur lance des votes pour les k proches clustres. Le geste associé à léchantillon de test et sa probabilité de reconnaissance sont définis, respectivement, par les équations suivantes :

$$g_{reconnu} = \underbrace{argmax}_{g} \sum_{i=1}^{M} L(g|dlm_i) \tag{2.13}$$

$$PROBAreconnaissance(g_{reconnu}) = \frac{\sum_{i=1}^{M} L(g_{reconnu}|dlm_i)}{M} \tag{2.14}$$

Avec $L(g|dlm_i)$ représente la probabilité du geste g selon les k proches clusters à partir de dlm_i.

Les auteurs ont proposé une version modifiée de l'algorithme "les k plus proches voisins". Cette version suppose que chaque séquence de test contient un seul geste. L'algorithme est donné comme suit :

Algorithm 1 Algorithme des K plus proches voisins

T (La base d'entraînement)
dlm_i, $i \geq 1$ (Les descripteurs locaux du mouvement à partir de la séquence de test)
Calculer $g_{reconnu}$, $PROBAreconnaissance(g_{reconnu})$
$M \leftarrow 1$

- **while** un dlm_i est généré **do**
  trouver les k plus proches voisins pour dlm_i
  $g^M_{reconnu} \leftarrow \underbrace{argmax PROBA^M(g)}_{g \in G}$
  $PROBAreconnaissance(g_{reconnu}) \leftarrow \frac{\sum_{i=1}^{M} L(g_{reconnu}|dlm_i)}{M}$
  $M \leftarrow M + 1$

Les auteurs ont aussi proposé une autre version pour un traitement en ligne.

Correa et al. [Correa et al., 2009] proposent un système de reconnaissance de gestes de la main qui permet l'interaction avec un robot, dans un environnement dynamique et en temps réel. Le système détecte les mains et les gestes statiques à l'aide d'une cascade de classifieurs boostés, et reconnaît les gestes dynamiques en calculant les statistiques temporelles de la main qui sont la position et la vitesse. Il classe ensuite ces fonctions à l'aide d'un classifieur de Bayes. La principale nouveauté de l'approche proposée est l'utilisation des informations de contexte afin d'adapter en permanence le modèle de peau utilisé dans la détection des mains, pour limiter les régions de l'image qui doivent être analysées, et de réduire le nombre d'échelles qui doivent être pris en compte dans le processus de recherche manuelle et de reconnaissance.

Parmi les divers gestes humains, le pointage est un geste très utile pour l'interaction homme-robot (HRI). En effet, ce geste est très intuitif, il n'implique pas d' hypothèses "à priori" et ne peut pas être substitué par d'autres modes d'interaction. Un problème majeur dans la reconnaissance du geste de pointage est la difficulté d'estimer précisemment la direction de pointage. Cela est dû à la difficulté du suivi de la main et au manque de la fiabilité dans l'estimation de la direction. Chang-Beom et al. [Park and Lee, 2011] proposent un algorithme de reconnaissance du geste de pointage 3D en temps réel pour les robots mobiles, basé sur une cascade de modèles de Markov cachés (HMM) et un filtre à particules. Les auteurs ont utilisé une caméra stéréo comme capteur. Le filtre à particules est utilisé pour localiser et suivre le visage et les mains. La première étape du HMM

prend l'estimation de la position de la main et fait un mappage pour obtenir une position plus précise par la modélisation de la cinématique du doigt. Les coordonnées 3D obtenues sont utilisées comme entrée pour la deuxième étape du HMM qui discrimine le geste de pointage des autres types de geste. Dans le but de reconnaître le même geste « pointage », Stefelhagen et al.[Stiefelhagen et al., 2004] décomposent le geste en trois phases : *début*, *continue* et *fin*, ils modélisent chaque phase avec un Modèle de Markov Caché. Les coordonnées de la main sont transformées en un système de coordonnées cylindriques centré sur la tête afin d'être invariant par rapport à l'emplacement de la personne. Pour déterminer la direction du pointage 3D, la droite liant le centre de la tête et le centre de la main est extraite. Le capteur utilisé est une caméra stéréo.

2.5.3 Les modèles 2D/3D

Il existe 2 types de modèles pour les gestes : (1) les modèles 3D et (2) les modèles 2D. Contrairement aux méthodes basées sur l'apprentissage, la technique basée sur les modèles se compose d'une seule étape. Elle consiste à extraire les paramètres de la cible réelle et les adapter au modèle adéquat du geste. L'ajustement du modèle est réalisé par la minimisation d'une mesure résiduelle entre le modèle projeté et les contours de la personnes (par exemple : les bords de son corps). Cela exige une très bonne segmentation des parties du corps. Ces techniques nécessitent donc des séquences vidéos de bonnes qualité qui ne présentent pas de bruit important. [Chu and Cohen, 2005] proposent une méthode de reconnaissance de gestes en utilisant un modèle géométrique 3D (3D Visual Hull). L'approche reconstruit le modèle 3D du corps à partir de plusieurs points de vues capturés avec plusieurs caméras. Un descripteur 3D de formes est ensuite calculé, il contient un ensemble de propriétés géométriques du modèle 3D. Finalement, un matching est réalisé en utilisant un modèle de la posture, connue à priori, et un Modèle de Markov Caché à deux états. [Muñoz-Salinas et al., 2008] proposent d'appliquer une représentation de profondeur de la silhouette (c'est à dire une silhouette 3D combinant la silhouette 2D et la profondeur).

2.5.4 Le Template Matching

Dans les méthodes basées sur le Template Matching il n'y a ni extraction de caractéristiques ni modèle de geste, on considère le geste entier comme un template. [Roh et al., 2006]

proposent un template de Mouvement de Volume (VMT) pour une reconnaissance de gestes à vue invariante. L'apprentissage de la base de données consiste en la projection des images des VMTs de chaque geste appris. Pour faire le matching pour un nouveau geste, l'algorithme des k-proches voisins est utilisé. Ces méthodes sont différentes des méthodes 2D puisque les dimensions temporelle et spatiale sont incluses dans le même modèle. Un système de reconnaissance n'est donc pas nécessaire. En plus, c'est différent des modèles basés sur le mouvement où les descripteurs locaux ou globaux (par exemple : HOG, SIFT) sont extraits puis appris. Ici, le template du geste entier est utilisé comme un modèle d'apprentissage. L'inconvénient de ces méthodes est la taille énorme des données d'apprentissage qui influe sur le coût de calcul du processus de matching.

2.6 Synthèse et discussion

Plusieurs approches ont été proposées pour reconnaître des gestes en se basant sur les dispositifs de vision dans une vaste gamme d'applications. Nous exposons ci-dessous les avantages et les inconvénients de chaque catégorie d'approches ensuite nous présentons notre méthode.

L'efficacité

Nous avons vu que parmi les méthodes les plus efficaces pour reconnaître des gestes il y a celles qui utilisent un modèle 3D du corps humain. Cependant, ces méthodes efficaces qui reposent sur l'apparence sont influencées par les conditions de l'environnement. Pour palier ce problème, des travaux de recherche ont combiné les informations couleurs avec l'information de profondeur. Ces dernières méthodes semblent être les plus prometteuses. En effet, nous avons vu dans les sections précédentes que les méthodes basées sur les modèles exigent une très bonne segmentation des parties du corps et que la conception d'un tel modèle peut être complexe. En ce qui concerne les algorithmes d'apprentissage et les algorithmes basés sur la modélisation statistique, ils semblent être plus flexibles puisqu'ils ne limitent pas le nombre de gestes à reconnaître.

La simplicité

Il n'y a aucun doute que la description la plus fidèle des gestes est obtenue en utilisant les modèles 3d du corps humain avec mouvement. Néanmoins, en prenant en compte tout les aspects du mouvement, la conception et le paramétrage d'un tel modèle sont souvent trop complexes. Les méthodes basées sur les apparences sont plus simples à modéliser et à

paramétrer. La récupération de l'information de la profondeur est devenue directe et facile avec le capteur Kinect. Son utilisation seule nécessite la définition d'un bon descripteur, c'est ce que nous allons détailler dans les chapitre suivants. Les approches basées sur le mouvement sont plus faciles à mettre en oeuvre puisqu'elles incluent dans le même modèle l'aspect spatial et temporel du geste.

Chapitre 3

Modélisation du geste

3.1 Introduction

3.1.1 Les objectifs

Notre objectif est de reconnaître des gestes humains dynamiques. Nous voulons reconnaître deux types de gestes dynamiques : (1) des gestes simples (c'est à dire une seule action) et (2) des gestes continus (c'est à dire deux ou plusieurs gestes simples successifs).

3.1.2 Les contraintes

De nombreuses techniques on été déjà proposées pour la reconnaissance de gestes dans des environnement spécifiques (des laboratoires) soit en combinant plusieurs capteurs (caméra et marqueurs) ou bien plusieurs types d'informations (profondeur et couleur). Cependant, La reconnaissance de gestes reste sensible et dépend souvent des dispositifs utilisés et de l'environnement. Nous proposons de réduire ces contraintes afin de concevoir un système permettant de reconnaître les gestes quel que soit l'environnement en utilisant un seul capteur et un descripteur basé sur un seul type d'informations (la profondeur) pour simplifier les traitements et réduire le temps de calcul.

3.1.3 L'approche proposée

Afin de reconnaître les gestes simples et continus, nous proposons une approche composée de trois étapes. Dans la première étape, nous calculons pour la personne détectée, un

ensemble de descripteurs 3D, basés sur les informations 3D (c'est à dire : la position x et y et la profondeur) de chaque articulation de la partie supérieure du corps. La deuxième étape consiste à faire l'apprentissage des descripteurs pour les gestes identifiés avec une base de donnée d'apprentissage. Dans la dernière étape, nous extrayons le descripteur 3D du geste de la nouvelle séquence de données. Ensuite, nous le classifions par rapport aux gestes déjà appris, en utilisant une des deux méthodes de classification que nous avons développées (chapitre 4).

3.2 Représentation du flux de données

3.2.1 Les capteurs

Parmi les caméras utilisées dans les travaux récents, il y a celles basées sur la lumière infrarouge et celles basées sur la lumière visible. On peut citer par exemple : la caméra temps de vol (Time-of-Flight), qui utilise la lumière infrarouge modulée et génère une image d'intensité avec l'information de la distance codée dans chaque pixel du capteur (Photonic Mixer Device PMD), la caméra stéréo qui est utilisée pour restituer la 3D, et enfin la kinect qui fournie une image RGB et une image de profondeur, et pour laquelle nous opterons dans notre travail.

La caméra temps de vol 3D

C'est une caméra basée sur le principe du temps de vol (figure 3.1). Elle permet de mesurer, en temps réel, une scène en 3 dimensions en fournissant jusqu'à 50 images par seconde. Pour ce faire, la caméra illumine d'abord la scène par une pulsation lumineuse et mesure le temps que ce pulse prend pour effectuer le trajet entre l'objet et la caméra. La mesure de temps de vol est effectuée indépendamment pour chaque pixel de la caméra, permettant ainsi d'obtenir une image complète en 3D de l'objet mesuré, il s'agit d'une image des distances. Le temps de vol t est directement proportionnel à la distance d traversée : $d = \frac{c.t}{2}$, avec c : la vitesse de la lumière. La caméra TOF peut fournir les images de niveau de gris ainsi que la distance de chaque pixel. L'image des distances est indépendante de la texture et du niveau de luminosité. Cependant, elle est affectée par la couleur de l'objet parce que la distance de l'image dans la caméra TOF est calculée par la différence entre le rayon infrarouge transmis à l'objet et celui réfléchi par cet même objet. Or les couleurs ont

FIGURE 3.1 – Une caméra temps de vol (TOF).

différents facteurs de réflexion ce qui influence l'information de distance ; c'est-à-dire que deux objets qui sont à la même distance et avec des couleurs différentes peuvent renvoyer une information de distance différente. Ce problème est souvent cité dans les travaux sur la segmentation utilisant ce type de caméra. S.E. Ghobadi et al. [Ghobadi et al., 2007] ont proposé une solution basée sur la fusion des ensembles des vecteurs de l'intensité et de la distance afin de dériver un nouvel ensemble de données appelé *phase*, qui indique l'angle entre les deux ensembles de vecteurs d'intensité et de la distance. L'utilisation des données d'intensité et de distance dans chaque image produit un nombre complexe : $C_{rc} = g_{rc} + id_{rc}$, avec r, c, g et d désignent respectivement la ligne, la colonne, la valeur normalisée de gris et l'information de distance normalisée.

La caméra stéréoscopique

Une caméra stéréoscopique (figure 3.2) dite aussi stéréo est une caméra qui dispose de deux lentilles et prend deux photos en même temps. Ceci simule la perception humaine et crée ainsi l'effet 3D observé. Cette caméra se base sur la vision stéréo qui permet de reconstituer la structure tridimensionnelle d'une scène à partir de sa projection en deux images. Afin d'implémenter un système de vision stéréoscopique, quatre étapes sont nécessaires [Moreno et al., 2001] :

- La calibration de la tête stéréo : Dans cette étape, les matrices de projection sont déterminées en utilisant l'algorithme de calibrage.
- La rectification d'image : Après le calcul des matrices, l'image est transformée de

FIGURE 3.2 – Une caméra stéréoscopique.

telle sorte que les lignes épi-polaires deviennent colinéaires en chaque point d'image.

- La correspondance : C'est l'étape critique. Il s'agit d'identifier la projection du même point 3D dans les deux images. Dans [Moreno et al., 2001], les auteurs ont utilisé une méthode de corrélation qui prend le niveau de gris du voisinage d'un pixel d'intérêt, et recherche l'emplacement du pixel ayant une distribution du gris similaire dans l'autre image.
- Raffinement de la carte de profondeur : La dernière étape est le raffinement du calcul de correspondance.

Parmi les points faibles de cette caméra, elle nécessite des traitements de bas niveau et si le mouvement est rapide alors le suivi est perdu.

Le capteur Microsoft Kinect

Le capteur Microsoft Kinect (figure 3.3) est un capteur qui permet de réaliser de la capture d'images 3D, de la reconnaissance faciale et de la reconnaissance vocale. Le capteur Kinect est un outil connu dans le monde du jeu vidéo et aussi très utilisé dans le monde de la robotique pour les capacités qu'il offre en terme d'interaction homme-robot. Il est constitué :

- D'une base motorisée
- D'une caméra couleur RGB
- D'un capteur de profondeur
- D'un microphone
- D'un ensemble de logiciels permettant la reconnaissance de mouvements

FIGURE 3.3 – Un capteur Microsoft Kinect.

Le capteur de profondeur qui permet la mise en relief de l'image saisie par la Kinect est constitué d'un projecteur laser infrarouge et d'une mini caméra CMOS monochrome. Ce capteur de profondeur permet la capture d'images en 3D, quelles que soient les conditions lumineuses dans un milieu intérieur. Selon Microsoft, les logiciels de reconnaissance de mouvements représentent la principale innovation de la Kinect. La pile logicielle embarquée sur la Kinect, réalisée par Microsoft, permet de réaliser de la reconnaissance de gestes du corps humain. C'est cette fonctionnalité qui est mise à contribution lors de l'usage de la Kinect en tant que capteur sur la console de jeu XBox 360. La Kinect est capable de détecter 5 personnes et en suivre deux en même temps. Pour la personne activement reconnue (l'équivalent du joueur), 20 positions d'*ariculations* sont interprétées. Ce même logiciel permet également la reconnaissance faciale et la reconnaissance vocale (pour l'instant peu mise à contribution) [http ://www.generationrobots.com,].

3.3 L'extraction de données

Le suivi avec l'algorithme Skeleton (kinect)

Avant de passer à la reconnaissance des gestes, il est indispensable d'effectuer d'abord un suivi. Comme mentionné dans l'état de l'art, la plupart des méthodes existantes dans les applications interaction homme-système sont basées sur l'information de la couleur. Or, la couleur n'est pas stable et fortement influencée par différents facteurs, tels que le changement de la luminosité et les occultations. Par conséquent, les méthodes basées sur la couleur ne parviennent pas toujours à donner une posture 3D de la personne. Dans notre travail, nous avons choisi d'utiliser le capteur Kinect qui fournit la profondeur. La profondeur est la distance entre le capteur et un point dans la scène. La figure 3.4 représente le système de coordonnées du capteur utilisé. Les coordonnées x, y et z désignent

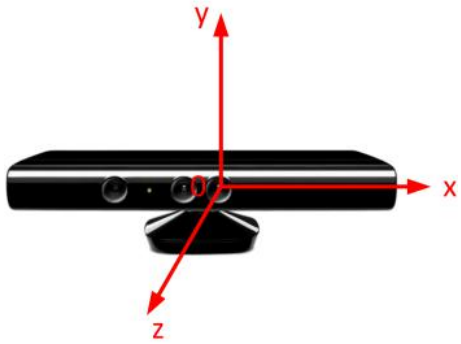

FIGURE 3.4 – Système de coordonnées de la Kinect.

respectivement les positions x et y et la profondeur. Nous effectuons le suivi en utilisant la méthode de Skeleton fournit par la kinect SDK. Le suiveur SDK détecte la position de l'ensemble des articulations, dans l'espace 3D, donné comme suit :

$$\begin{aligned} \text{G}=&\{g_i, i \in [1, I]\} \equiv \{Torso, Neck, Head, Left\ shoulder, \\ &Left\ elbow, Left\ wrist, Right\ shoulder, Right\ elbow, Right\ wrist, Left\ hip, Left\ knee, \\ &Left\ foot, Right\ hip, Right\ knee, Right\ foot, Spine, Right\ hip, Left\ hip, Center\ hip, \\ &Right\ ankle, Left\ Ankle\} \end{aligned} \quad (3.1)$$

La position de l'articulation g_i est définie par le vecteur $p_i(t) = [xyz]^T$, où t dénote l'instant où l'image a été acquise. L'origine du système de coordonnées XYZ est placée au centre du capteur kinect. Le module de suivi Skeleton nécessite la calibration de l'utilisateur afin d'estimer plusieurs caractéristiques du corps de la personne. La version récente du module, le mode "autocalibration" permet la calibration de la personne sans contraindre l'utilisateur. Comme mentionné avant, le suivi est une étape indispensable en amont de la reconnaissance. Il est généralement basé sur la détection et localisation d'un ou de plusieurs membres du corps humain. Souvent, des gants de couleurs différentes sont utilisées pour distinguer la main droite de la main gauche. Dans notre cas, nous utilisons le suivi de Skeleton de la kinect. Par conséquent, on peut distinguer les différentes parties du corps humain. La figure 3.5 montre la nature de l'information utilisée dans notre approche : l'image de profondeur (b) et le suivi du Skeleton (c).

En plus, concernant le geste « pointage », le calcul de l'orientation est souvent fait par

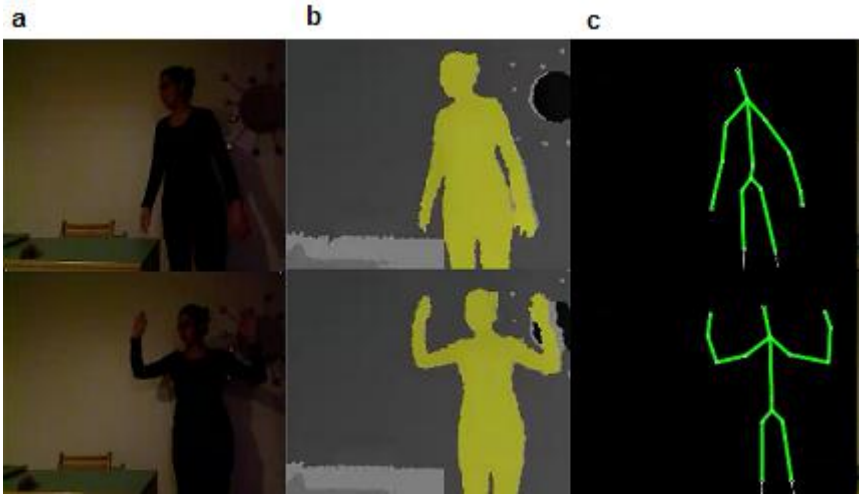

FIGURE 3.5 – (a) image RGB, (b) image de profondeur, (c) suivi du Skeleton.

le calcul de la ligne entre le centre de la main et la tête ce qui ne donne pas une direction précise vers le point visé. Mais, si on calcule la direction en se basant sur la position 3D du centre de la main et du coude on aura une estimation plus représentative du point visé ce que fait la Kinect. La figure 3.6 montre les articulations modélisées. Nous avons implémenté un programme en $C++$ qui permet de récupérer la position 3D des articulations qui composent les gestes. À l'aide de ces positions nous calculons les angles entre les articulations : angle coude et angle épaule. La récupération de ces informations se fait en temps réel et en parallèle avec l'exécution des gestes. La figure 3.7 montre un exemple de deux squelettes actifs suivis par la kinect.

3.4 Description des gestes

Dans le cadre de cette thèse, nous nous intéressons aux gestes déictiques notamment les gestes de contrôle et de pointage. Le travail consiste à reconnaître en trois dimensions cinq gestes dynamiques en se basant sur l'information de profondeur. Les cinq gestes que nous voulons reconnaître sont : *viens, recule, stop, pointage à droite* et *pointage à gauche*. Nous considérons ces cinq gestes, parce qu'ils sont parmi les gestes les plus utilisés dans l'interaction homme-homme et seront parfaitement adaptés aux interactions homme-système, notamment, l'interaction homme-robot. La figure 3.8 représente l'exécution de chacun de ces gestes. Pour ce faire, on part de la position 1 vers la position 5 en passant par les positions intermédiaires. Notre définition pour ces gestes nous a permis de constater qu'il y a trois angles actifs dans le déroulement de ces gestes : l'angle correspond au coude

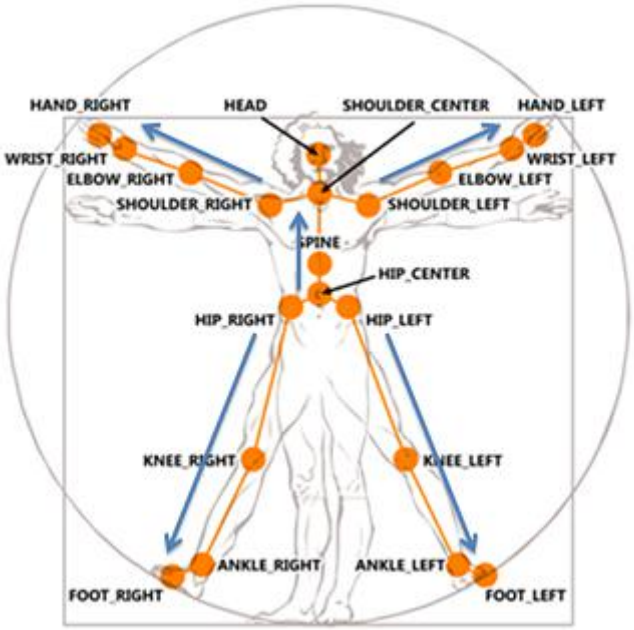

FIGURE 3.6 – Les articulations suivies par la Kinect

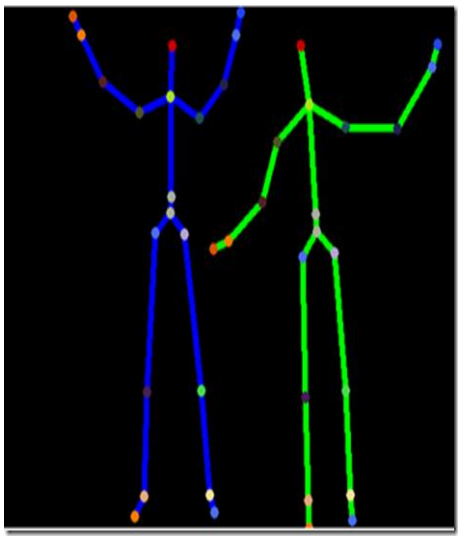

FIGURE 3.7 – Skeleton actif détecté par la Kinect

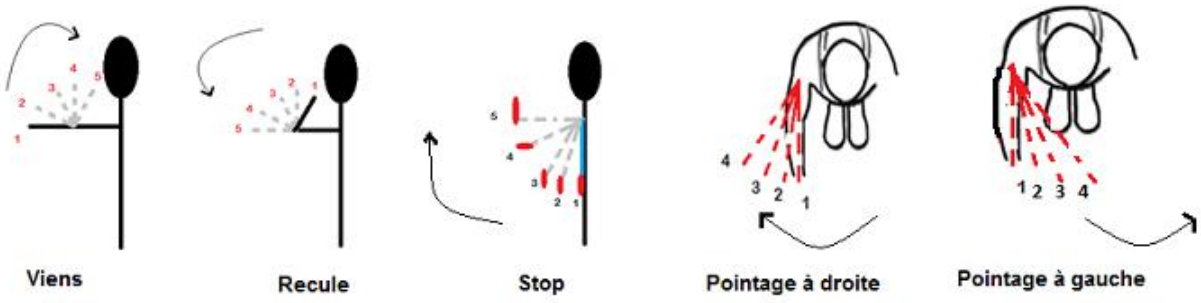

FIGURE 3.8 – Les cinq gestes à reconnaître.

α, épaule β et aisselle γ comme montré dans la figure 3.9. L'idée principale de notre approche est d'estimer, en temps réel, les variations de ces angles lors de l'exécution de chaque geste. Chaque angle est calculé à partir des coordonnées 3D des 3 articulations qui le composent comme suit :

- α (l'angle coude) est calculé à partir des coordonnées 3D des articulations : poignet, coude et épaule.
- β (l'angle épaule) est calculé à partir des coordonnées 3D des articulations : poignet, épaule droite et épaule gauche.
- γ (l'angle aisselle) est calculé à partir des coordonnées 3D des articulations : coude, épaule et hanche.

  Les variations des angles au cours du temps d'exécution de chaque geste sont présentées sur les figures de 3.10 à 3.14. D'après ces figures, on peut constater que chaque geste est caractérisé par un angle qui varie le plus, ce qui permet de bien le reconnaître. Ces variations sont aussi récapitulées sur la tableau 3.1. En exécutant un geste, on enregistre les valeurs des angles dans des vecteurs :

$$V_\alpha = [\alpha_1, \alpha_2, ..., \alpha_T] \tag{3.2}$$

$$V_\beta = [\beta_1, \beta_2, ..., \beta_T] \tag{3.3}$$

$$V_\gamma = [\gamma_1, \gamma_2, ..., \gamma_T] \tag{3.4}$$

  Avec T la longueur de la séquence du geste qui est variable selon le geste. Afin d'utiliser toutes ces caractéristiques, nous avons proposé un descripteur 3D qui combine

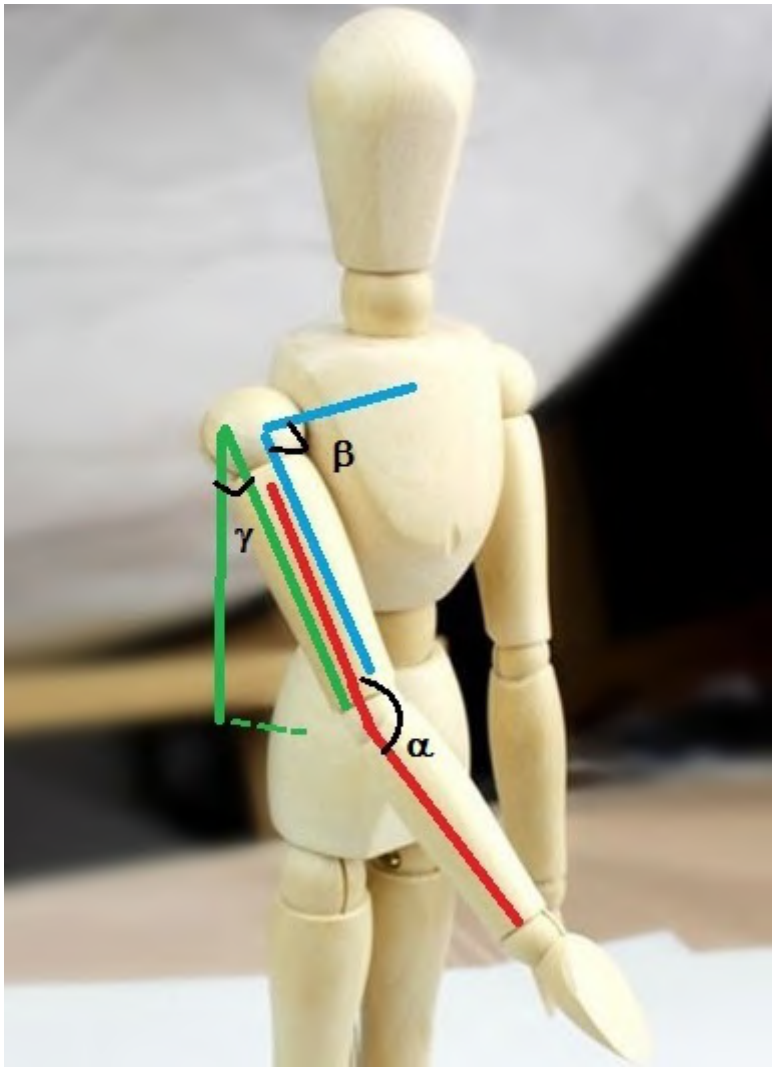

FIGURE 3.9 – Les angles α, β et γ.

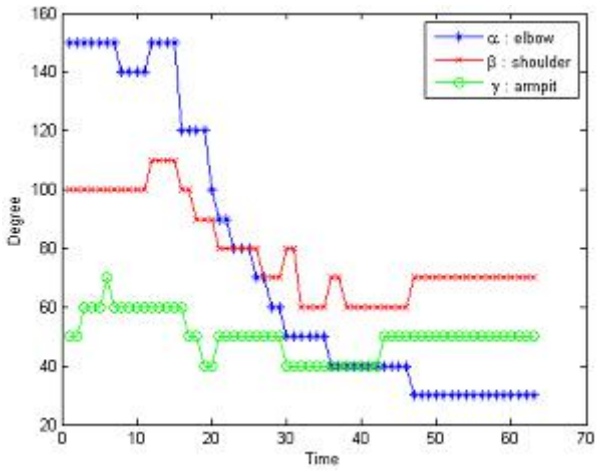

FIGURE 3.10 – Les variations des angles pour le geste *Viens*.

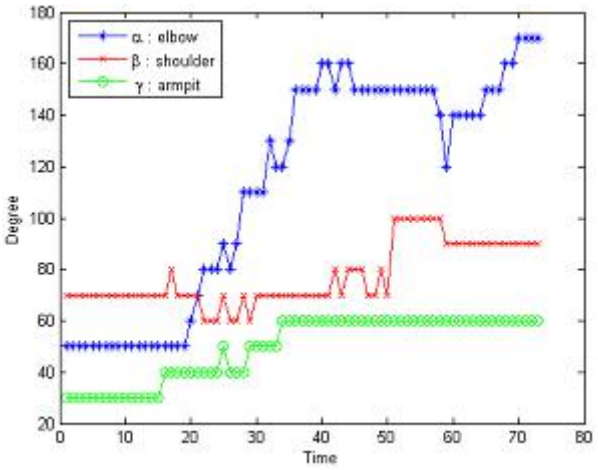

FIGURE 3.11 – Les variations des angles pour le geste *Recule*.

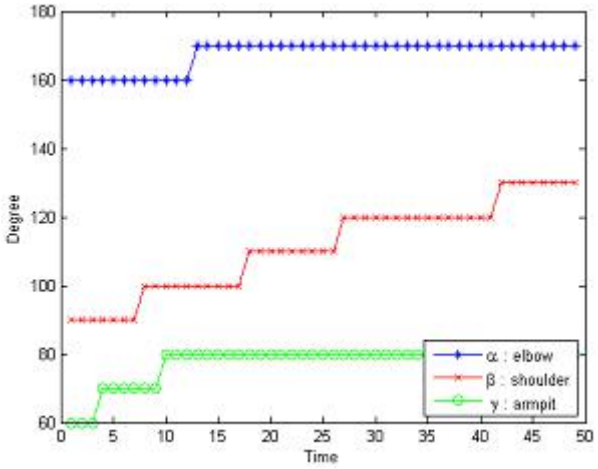

FIGURE 3.12 – Les variations des angles pour le geste *Pointage à droite*.

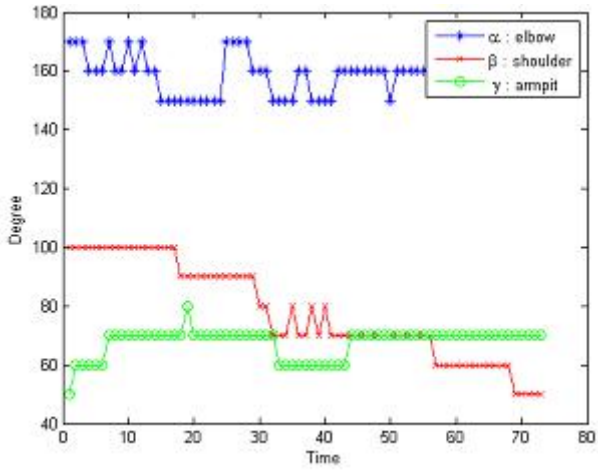

FIGURE 3.13 – Les variations des angles pour le geste *Pointage à gauche*.

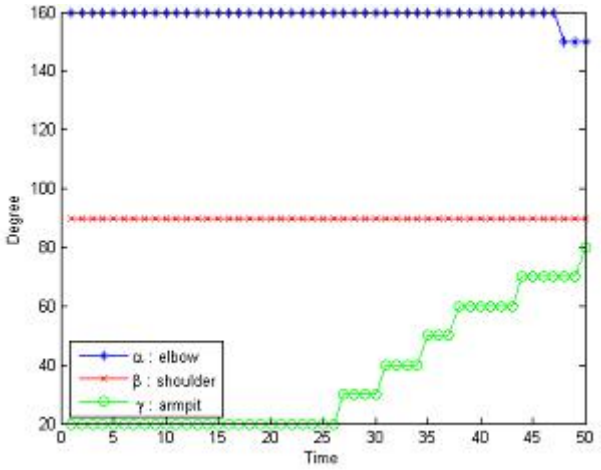

FIGURE 3.14 – Les variations des angles pour le geste *Stop*.

les variations de ces angles :

$$V_\alpha = [\alpha_1, \alpha_2, ..., \alpha_T, \beta_1, \beta_2, ..., \beta_T, \gamma_1, \gamma_2, ..., \gamma_T] \tag{3.5}$$

L'algorithme suivant (Algorithme 1) montre les étapes suivies pour calculer la valeur d'un angle A à partir de trois points a, b et c. Cette description basée sur les variations

Algorithm 2 Calcul de l'angle à partire de 3 points

Données : $(x_a,\ y_a,\ z_a)$, $(x_b,\ y_b,\ z_b)$, $(x_c,\ y_c,\ z_c)$

Sortie $A = \widehat{abc}$
Calculer $\vec{ab}$, $\vec{cb}$:
$\vec{ab} \leftarrow (\mathrm{x}_b - x_a,\ y_b - y_a,\ z_b - z_a)$
$\vec{cb} \leftarrow (\mathrm{x}_b - x_c,\ y_b - y_c,\ z_b - z_c)$
Normaliser $\vec{ab}$, $\vec{cb}$:
$\vec{ab} \leftarrow 1\frac{}{||\vec{ab}||}$
$\vec{cb} \leftarrow 1\frac{}{||\vec{cb}||}$
Le produit scalaire de $\vec{ab}$ et $\vec{cb}$:
product $\leftarrow (\mathrm{x}_{\vec{ab}} \cdot \mathrm{x}_{\vec{cb}}) + (y_{\vec{ab}} \cdot \mathrm{y}_{\vec{cb}}) + (z_{\vec{ab}} \cdot \mathrm{z}_{\vec{cb}})$
$A \leftarrow \arccos(product)$
changer l'unité d'angle du radian vers le degré. =0

des angles permet une distinction entre les gestes humains. En effet pour chaque geste on constate qu'il y a un angle principal qui change le plus et prend différentes valeurs tandis que les deux autres angles ne changent que légèrement. Considérons les cinq gestes définis auparavant ; l'angle variable est α dans *viens* et *recule*, γ dans *stop*, et β dans les deux gestes de pointage. Les variations de l'angle principal dans chaque

TABLE 3.1 – Les variations de l'angle principal dans chaque geste.

	α	β	γ
Viens	180° →30°	-	-
Recule	30° →180°	-	-
Pointage à D.	-	90° →150°	-
Pointage à G.	-	90° →40°	-
Stop	-	-	30° →80°

geste sont représentées dans le Tableau 1.

3.5 Protocole expérimental

Afin de réaliser des expérimentations pour évaluer le système de reconnaissance proposé, nous avons besoin d'une base de donnée contenant les positions 3D des articulations capturées par la Kinect. Nous avons trouvé deux bases de données réalisées avec la Kinect et contiennent les données 3D des articulations que nous cherchons à savoir la base de données nommée *Body and hands* et *CAD-60 CAD-120* téléchargeables respectivement sur [https ://www.microsoft.com/en us/download/details.aspx ?id=52283,] et [http ://pr.cs.cornell.edu/humanactivities/data.php,]. Cependant, les gestes proposés dans ces bases de données ne sont pas des gestes d'interaction mais des gestes visés plutôt pour la surveillance. Par exemple, parmi les gestes proposés dans la base *CAD-60 CAD-120*, il y a (parler sur le téléphone, brosser les dents, manger). Par manque de base de données qui représentent les gestes d'interaction, nous avons décidé d'en construire une.

3.5.1 La base de données

Nous avons construit une base de données à partir d'une population de 20 individus. Chaque personne est invité à exécuter les 5 gestes séparément. Le même geste est exécuté 5 fois par personne. En somme, on obtient 500 séquences. La personne doit se mettre en face du capteur Kinect. C'est à elle de choisir sa distance par rapport à la Kinect à condition que celle ci ne dépasse pas les $3m$. Premièrement, le programme d'acquisition est tourné. Un indice de début s'affiche sur l'écran indiquant à la personne qu'elle peut commencer l'exécution du geste, ce qui veut dire que la calibration du corps de la personne a été effectuée. La personne sait en avance le type de geste

à faire en s'aidant d'un descriptif sur le déroulement de la séance d'expérimentation. Lors de l'exécution du geste, le programme d'acquisition calcul en parallèle les valeurs des angles des articulations et les enregistre dans des vecteurs. A la fin du geste, nous arrêtons le programme. Les vecteurs sont rassemblés dans des fichiers selon le type du geste. La base de données est divisée en deux, 250 séquences sont utilisées pour l'entrainement (50 pour chaque type de geste) et 250 séquences pour le test (50 pour chaque type de geste). Pour la phase de validation, nous avons demandé aux sujets n'ayant pas participé à la phase d'entraînement d'exécuter chaque geste 10 fois.

3.5.2 Protocole expérimental

Avant de commencer les expériences, les sujets doivent connaître le type de geste à exécuter ainsi que son déroulement depuis le début jusqu'à la fin du geste. La durée de l'exécution d'un geste n'est pas fixée. Le sujet peut exécuter un geste lentement, comme il peut le faire rapidement. La distance entre la Kinect et le sujet doit être comprise entre 80 cm et 3 m afin de bien détecter la personne tandis que la caméra reste fixe. La figure 5 montre quelques cas où le Kinect n'arrive pas à détecter totalement le corps. L'environnement est plus ou moins encombré et il n'y a obstacle entre le sujet et la Kinect. Le sujet demeure debout et en face de la Kinect tout au long du geste.

Les conditions expérimentales :

* Objectif de l'expérimentation : La conception d'une base de données de cinq gestes canoniques simples
* Matériel choisi : Caméra Kinect
* Le facteur variable : Le sujet qui va faire l'expérience (20 personnes)
* Les facteurs constants : La distance entre la Kinect et le sujet : entre 80 et 300 cm (pour bien détecter la personne)
* Les gestes à effectuer : Viens, recule, stop, pointage à gauche, pointage à droite

Le déroulement de l'expérience :

* Geste 1 « Viens » : le sujet est invité à se tenir droit devant la Kinect. Il doit attendre que sa silhouette soit détectée par la kinect. Un message s'affiche indiquant que la silhouette a été détectée. Le sujet tend son bras droit vers la caméra, effectue le geste viens en pliant son bras au niveau du coude de tel sorte

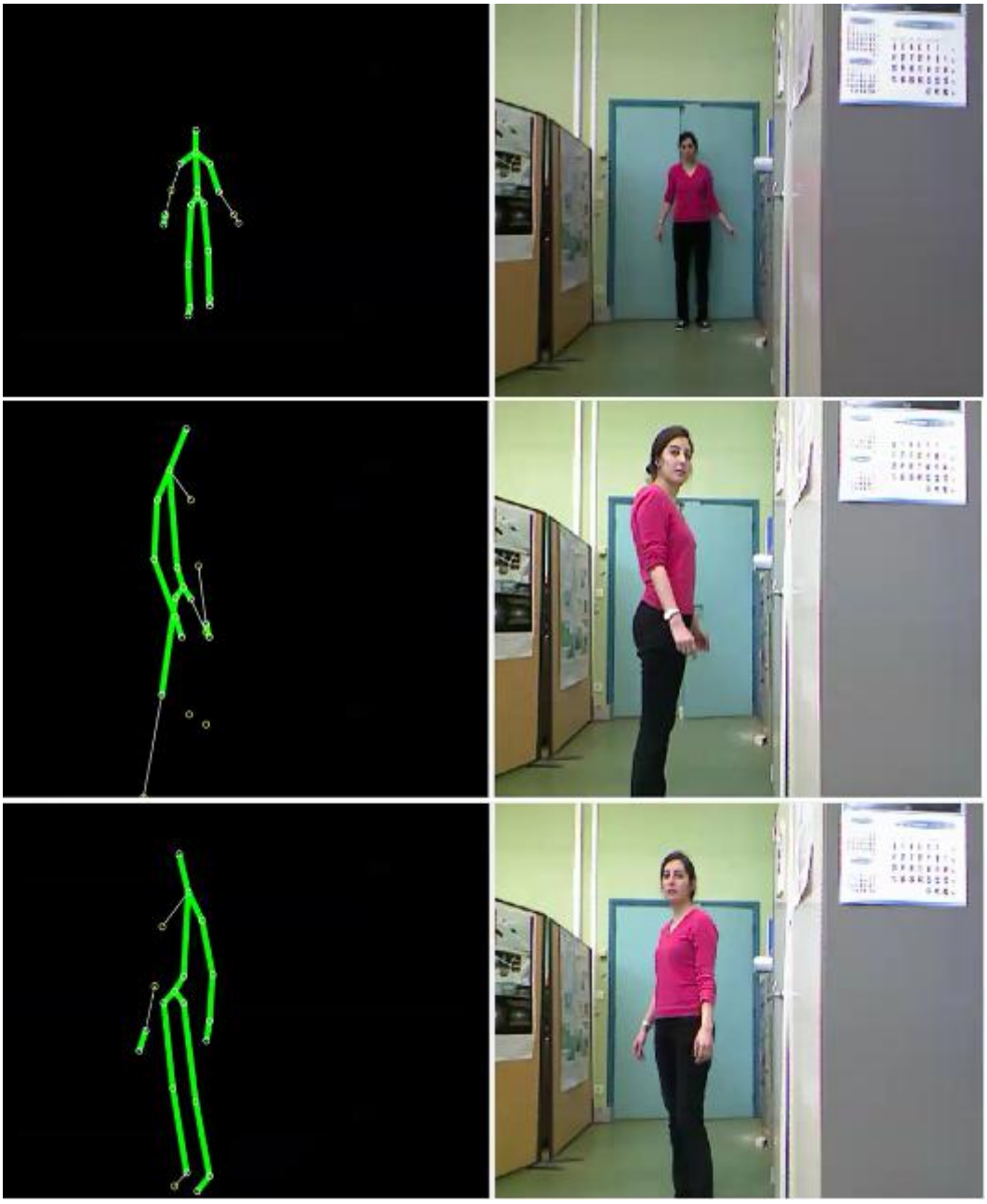

FIGURE 3.15 – Les cas d'échec de la détection par la Kinect ; première image : la distance est supérieure à $3m$, deuxième et troisième image : le sujet n'est pas face à la Kinect.

que sa main s'approche de son épaule au cours du geste.

* Geste 2 « Recule » : le sujet est invité à se tenir droit devant la Kinect. Il doit attendre que sa silhouette soit détectée par la kinect. Un message s'affiche indiquant que la silhouette a été détectée. Le sujet commence par la position finale du geste « Viens » et essaye d'ouvrir son bras en effectuant le geste recule comme s'il demande à quelqu'un de s'éloigner.
* Geste 3 « Stop » : le sujet est invité à se tenir droit devant la Kinect. Il doit attendre que sa silhouette soit détectée par la kinect. Un message s'affiche indiquant que la silhouette a été détectée. Le sujet lève son bras jusqu'à ce qu'il atteigne la hauteur de son épaule.
* Geste 4 « Pointage à droite » : le sujet est invité à se tenir droit devant la Kinect. Il doit attendre que sa silhouette soit détectée par la kinect. Un message s'affiche indiquant que la silhouette a été détectée. Le sujet tend son bras droit vers la caméra en le laissant un peu incliné. Il bouge son bras sur la même hauteur vers la droite en pointant sur un objet ou un lieu qui se trouve à sa droite.
* Geste 5 « Pointage à gauche » : le sujet est invité à se tenir droit devant la Kinect. Il doit attendre que sa silhouette soit détectée par la kinect. Un message s'affiche indiquant que la silhouette a été détectée. Le sujet tend son bras droit vers la caméra en le laissant un peu incliné. Il bouge son bras sur la même hauteur vers la gauche en pointant sur un objet ou un lieu qui se trouve à sa gauche.

Chaque sujet est invité à répéter le même geste 5 fois. Ce qui donne au final 25 expériences pour chaque sujet. Le milieu et la luminosité n'influencent pas l'acquisition de données, cependant il ne faut pas avoir un obstacle entre la kinect et la personne pour ne pas perdre le suivi. A partir de ces cinq gestes simples que nous avons présentés, nous avons construit une nouvelle base de données contenant deux types de gestes composés : (1) gestes composés sans transition entre eux et (2) gestes composés avec transition. Le premier ensemble des gestes composés contient une combinaison de gestes avec, à chaque fois, la fin du premier geste égal au début du geste qui le suit comme :

Geste continu N 1 = Stop+ viens
Geste continu N 2 = Stop+ pointage à droite

Geste continu N 3 = Recule + pointage à gauche

Geste continu N 4 = Viens + recule

Le deuxième ensemble de gestes composés contient une combinaison de gestes avec la fin du premier geste est différente du début du geste qui le suit tels que :

Geste continu N 1 = Viens + pointage à droite

Geste continu N 2 = Viens + stop

Geste continu N 3 = Recule + stop

Geste continu N 4 = pointage à gauche + stop

3.5.3 Critères d'évaluation

Dans cette sous section, nous définissons la métrique utilisée pour nos expérimentations. La validation d'un système d'apprentissage/classification est généralement effectuée via la métrique standard. Dans ce type de système, deux sortes d'erreur de classification peuvent survenir : (1) erreurs statistiques et (2)erreurs systématiques. Une erreur statistique est généralement causée par des fluctuation intrinsèques aléatoires et imprévisibles, de l'appareil de mesure (caméra ou encore processus de prétraitement) ou du système étudié. Cependant, une erreur systématique est généralement causée par des fluctuations non-aléatoires d'une source inconnue (c'est-à-dire une dérive constante) et qui,une fois identifiée, peut être éliminée. Puisque la méthode proposée est basée sur un modèle statistique de signature de mouvement local, nous supposons que toutes les erreurs générées par le système sont statistiques, ce qui, en réalité, n'est pas vrai. Ainsi, le résultat d'un classifieur appartient nécessairement à une de ces quatre catégories :

* Vrai positif : aussi connu comme classification positive, il se produit si le classifieur a détecté la même chose qui existe en vérité.
* Faux positif : aussi connu sous le nom de l'erreur α, c'est l'erreur de rejeter une hypothèse de classification quand elle est vraie. Autrement dit, il se produit lorsque nous reconnaissons un événement qui n'existe pas en vérité. Par exemple, un classifieur détecte un geste "Viens" alors qu'en réalité il n'en n'est pas un. C'est l'erreur de commission (c'est-à-dire la crédulité excessive) qui survient lorsque certains événements sont classés comme étant d'autres.

* Faux négatif : aussi connu comme l'erreur β, c'est l'erreur de ne pas rejeter une hypothèse de classification quand elle n'est en fait pas vraie (c'est-à-dire qu'elle aurait dû être rejetée). Par exemple, un classifieur suppose qu'il n'y a aucun geste quand en réalité il y'en a un. C'est l'erreur d'omission (c'est-à-dire le scepticisme excessif) qui survient lorsqu'un événement n'est pas classé correctement.
* Vrai négatif : aussi connu sous le nom de classification négative, il se produit si le classifieur ne détecte rien quand en réalité il n'y a aucun événement.

En se basant sur ces quatre catégories de résultats, plusieurs mesures de qualité peuvent être définies pour évaluer l'efficacité d'un classifieur :

* Justesse : elle mesure le degré d'exactitude du classifieur. C'est le degré de proximité d'un résultat de classification de sa valeur réelle. Son expression est donnée par :

$$justesse = \frac{nombre\ des\ V.P\ + nombre\ des\ V.N}{les\ V.P + les\ F.P + les\ V.N + les\ F.N} \tag{3.6}$$

  Il est difficile de calculer *la justesse* puisque le nombre de vrais négatifs ne peut pas être déterminé objectivement. De plus, avec l'hypothèse que seules les erreurs statistiques existent, la mesure de précision ne correspond pas à la définition exacte d'une justesse de système. Dans ce cas, une métrique plus adéquate peut être utilisée qui est la précision.
* Précision : la précision mesure la quantité de classifications correctes parmi toutes les classifications positives.

$$précision = \frac{nombre\ des\ V.P}{nombre\ des\ V.P + nombre\ des\ F.P} \tag{3.7}$$

Nous nous appuyons sur le calcul de la précision dans l'évaluation de notre système et de nos approches de reconnaissance proposées.

3.6 Conclusion

Dans ce chapitre, nous avons présenté les gestes que nous voulons reconnaître, ainsi que leurs descripteurs. Nous avons proposé un descripteur 3D basé sur la combinaison des variations des angles des articulations actives dans chaque geste. Nous avons détaillé le protocole expérimental, présenté la base de donnés réalisée et utilisée pour la classification, et nous avons donné le critère d'évaluation de notre système de

reconnaissance de gestes.

Chapitre 4

Reconnaissance de gestes simples

4.1 Introduction

La reconnaissance des gestes humains à partir des séquences vidéos se compose, généralement, de deux étapes : (1) la génération du descripteur de geste qui inclut aussi le tracking et (2) le processus de décision où les gestes sont effectivement reconnus. Ces deux étapes sont mutuellement reliées puisque le choix d'une représentation particulière du geste influence le processus de la décision et vice versa. Nous avons proposé de reconnaître les gestes en utilisant un descripteur 3D basé sur l'information de profondeur, comme détaillé dans le chapitre précédent. Puisque nous utilisons un descripteur local, alors le processus de décision adéquat pour ce genre de représentation correspond aux méthodes d'apprentissage/classification. En effet, ce type de processus est simple à mettre en oeuvre et flexible selon la variabilité du geste. La figure 4.1 représente le schéma général de notre système de reconnaissance de gestes.

Dans ce chapitre, nous expliquons le processus de décision proposé pour la reconnaissance des gestes qui se base sur les méthodes d'apprentissage/classification. Nous traitons le problème de la reconnaissance des gestes simples. Pour ce faire, nous utilisons les Modèles de Markov Cachés en créant un modèle pour chaque geste et en classifiant les gestes en fonction de la probabilité d'appartenance aux modèles. Par conséquent, le geste est affecté au modèle avec la plus grande probabilité d'apparte-

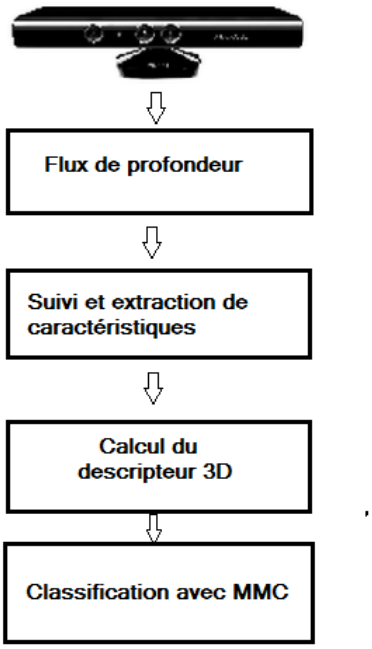

FIGURE 4.1 – Les principales étapes de notre système de reconnaissance

nance. La deuxième section présente la méthode basées sur les Modèles de Markov Cachés pour la reconnaissance des gestes simples. La troisième section présente l'extension de cette méthode proposée dans le but de l'améliorer en rejetant les gestes mal reconnus au lieu de les affecter aux mauvaises classes, en utilisant une méthode de seuillage basée sur la méthode « Déformation Temporelle Dynamique (DTW) ».

4.2 La méthode basée MMC

4.2.1 Formalisme

Le problème le plus difficile dans la reconnaissance de gestes dynamiques est la variabilité spatio-temporelle du geste. En effet, le même geste peut différer selon la vitesse, la forme et la durée. Ces caractéristiques rendent la reconnaissance des gestes dynamiques plus difficile comparé à celle des gestes statiques. Les Modèles de Markov Cachés sont des modèles statistiques très utilisés dans la reconnaissance de l'écriture manuscrite, la parole et les caractères. Grâce à leur capacité de modéliser des séries de temps spatio-temporel, les HMMs ont été utilisés avec succès dans la reconnaissance des gestes de la main. En effet, ils peuvent conserver l'identité spatio-temporelle du

geste de la main et ont une capacité de segmenter automatiquement le geste.
Dans ce travail, nous proposons d'utiliser les séquences des variations des angles comme des vecteurs d'entrées de notre système de reconnaissance. Un MMC peut être exprimé par le triplet des paramétres $\lambda = (A, B, \pi)$ et décrit par les éléments suivants :

a) Un ensemble de N états $S = \{s_1, s_2, \ldots, s_N\}$.

b) Une distribution de la probabilité initiale pour chaque état $\Pi = \{\pi_j\}$, $j = \{1, 2, \ldots, N\}$, avec $\pi_j = \mathrm{P}(S_j \text{ à } t = 1)$.

c) Une matrice de transition d'ordre $N \times N$, $A = \{a_{ij}\}$, avec a_{ij} la probabilité de transition de s_i vers s_j ; $1 \leq i,j \leq N$. La somme des éléments de chaque ligne de la matrice doit être égale à 1 parce qu'elle correspond à la somme des probabilités pour réaliser une transition d'un état donné à chacun des autres états.

d) Un ensemble d'observations $O = \{o_1, o_2, ..., o_t\}$, $t = \{1, 2, ..., T\}$.

e) Un ensemble de m symboles discrets $U = \{u_1, u_2, ..., u_m\}$.

f) Une matrice d'observation d'ordre $N \times M$, $B = \{b_{im}\}$, avec b_{im} la probabilité de génération du symbole u_m par l'état s_i. La somme des éléments de chaque ligne de la matrice doit être égale à 1 pour la même raison citée auparavant.

Un état est une partie du geste. Un geste peut être divisé en N états selon sa complexité ; si le geste est simple le nombre N est petit et vice versa. Chaque état génére un où plusieurs symboles (dits aussi observations). Dans notre application, une observation correspond à un élément du vecteur d'entrée (c'est à dire une valeur d'angle) et un état correspond à une suite d'observations. Il y a trois problèmes majeurs dans les MMCs : l'évaluation, le décodage et l'entrainement qui sont résolus en utilisant, respectivement, l'algorithme Forward, de Viterbi et Baum Welch [Laurent Bréhélin, 2010]. En outre, les MMCs ont trois topologies : le modèle ergodique où tous les états sont liés entre eux, le modèle Left-Right où chaque état est lié avec tous les états qui se produisent après et enfin le modèle Left-Right Banded où chaque état et lié seulement avec l'état qui le suit (voir la figure 4.2). Dans notre cas, nous avons choisi le modèle Left-Right Banded (figure 4.2(a)) parce qu'il est simple, et modélise bien les séries temporelles dont les propriétés changent dans le temps.

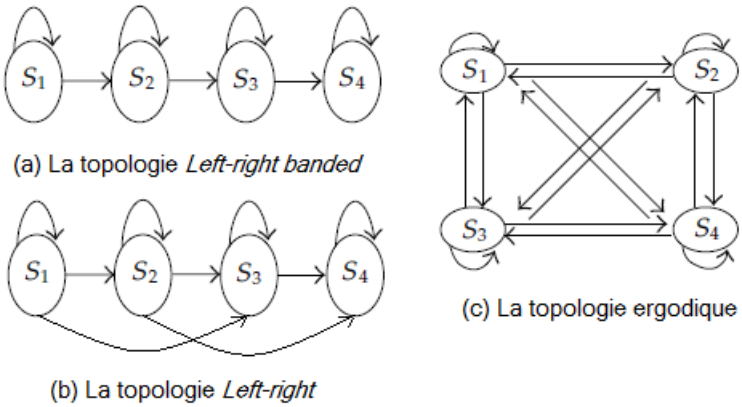

FIGURE 4.2 – Les topologies des MMCs

Les séquences d'observation

En utilisant les données extraites de la Kinect, nous avons défini deux catégories d'observations : un ensemble d'observations pour l'entraînement et un ensemble d'observations pour le test. Une observation dans notre cas est un vecteur qui contient les mesures des angles nommées les symboles. Sachant qu'à chaque geste sont associées plusieurs observations qui sont été enregistrées a priori lors de l'exécution de ce geste devant la caméra.

Initialisation des paramètres pour le modèle LRB

Nous avons réalisé cinq MMCs, un MMC pour chaque geste. Tout d'abord, il faut initialiser les paramètres du MMC. On commence d'abord par le nombre d'états. Ce dernier varie selon la complexité du geste. Dans notre travail, le nombre d'état varie entre 8 et 12 états. Le choix du nombre d'états pour chaque geste est expliqué dans la sous-section 4.2.2. La construction d'un modèle de markov caché se fait comme suit : tout d'abord, il faut définir les 3 paramètres qui le caractérisent et qui sont la distribution des états initiaux Π, la matrice de transitions entre les états A, et B la matrice d'émission de chaque symbole k par létat j. Dans un même geste on commence toujours par un même état qui est l'état initial du geste. Ainsi, la probabilité que l'état S_i se produit égale à 1, et la probabilité que chacune des autres états soit la première est 0. Ce qui nous donne $\Pi_1 = 1$ et $\Pi_i = 0$ avec $2 \leq i \leq N$ où N est le nombre des états c'est-à-dire le vecteur $\Pi = \{10...0\}$. Concernant les paramètres A et B du modèle HMM, ils sont initialisés aléatoirement et ré-estimés

en utilisant l'algorithme Baum-Welch. Pour initialiser la matrice de transition A, on lui affecte dans un premier temps des valeurs aléatoires. Comme nous utilisons la typologie Left-Right banded (c'est-à-dire qu'il n'existe que la transition entre l'état courant et létat suivant où l'état lui-même) alors la probabilité d'avoir une transition entre l'état courant et l'état précédent égale à 0. On affecte donc à ces transitions la valeur 0 dans la matrice A. Et pour les autres valeurs qui restent, c'est-à-dire les probabilités de transition de l'état courant vers lui-même et vers l'état suivant, on les modifie toujours aléatoirement de telle sorte que la somme de chaque ligne soit égale à1. Après on passe à la matrice d'émission B et on lui affecte directement des valeurs aléatoires en respectant la contrainte que la somme de chaque ligne soit égale à 1. Après l'initialisation des 3 paramètres du modèle HMM, on passe à la phase de l'entrainement. Ici, on fait appel à l'algorithme Baum-Welch qui est un algorithme itératif permettant d'estimer les paramètres du modèle qui maximisent la probabilité d'une séquence d'observations. L'algorithme essaye en premier temps de trouver les paramètres ML (Maximum Liklihood) c'est-à-dire estimer le maximum de vraisemblance des trois paramètres Π, A et B en utilisant les premières valeurs aléatoires données à ces paramètres comme étant les estimations initiales. Pour l'estimation, on fait appel à l'algorithme Expectation Maximization (EM). L'algorithme EM contient deux étapes E et M. L'étape E est réalisée en utilisant une astuce de programmation dynamique qui utilise l'indépendance conditionnelle entre les futurs états cachés et les derniers états cachés, étant donné les paramètres de l'état caché courant. L'étape M essaye de trouver les paramètres de Π, A et B qui maximisent la probabilité des données observées (qui sont les données de la base des gestes dans notre cas). L'algorithme s'arrête lorsqu'il converge vers le maximum local, c'est-à-dire lorsqu'il trouve les valeurs maximales des probabilités des états initiaux, les probabilités de transition et d'émission des données observées. Ensuite, on passe à l'étape du calcul de la vraisemblance d'une nouvelle donnée avec les modèles appris. Pour ce faire, on fait appel à une fonction qui calcule le logarithme de la vraisemblance d'un ensemble de données en utilisant un HMM discret. Pour chaque nouvelle donnée à reconnaitre, on répète ce dernier calcul pour tous les modèles HMM. Finalement, le modèle HMM dont la valeur de vraisemblance aux données est plus grande est considéré comme la classe recherchée. Voici un exemple d'initialisation des paramètres d'un modèle à 8 états. Le premier paramètre qui est le vecteur des probabilités initiales sera donc

désigné par :

$$\Pi = (1\ 0\ 0\ 0\ 0\ 0\ 0\ 0) \tag{4.1}$$

Le premier élément vaut 1 afin de s'assurer que le MMC commence par le premier état. Le deuxième paramètre est la matrice de transition A :

$$A = \begin{matrix} a_{ii} & 1-a_{ii} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{ii} & 1-a_{ii} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_{ii} & 1-a_{ii} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{ii} & 1-a_{ii} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{ii} & 1-a_{ii} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{ii} & 1-a_{ii} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_{ii} & 1-a_{ii} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{ii} \end{matrix} \tag{4.2}$$

avec a_{ii} la probabilité de transition entre deux états est initialisée par une valeur aléatoire. Le dernier paramètre est la matrice d'émission B déterminée par :

$$B = \{b_{im}\} \tag{4.3}$$

avec b_{im} la probabilité de génération d'un symbole par un état et initialisée aléatoirement.

Entrainement et évaluation

Après avoir initialisé les paramètres des MMCs, on passe à l'étape de l'entrainement et à l'évaluation. Nous utilisons l'algorithme Baum-Welch pour faire un entrainement complet des paramètres initialisés des MMCs $\lambda = (\Pi, A, B)$. Notre système est entrainé en variant le nombre d'état entre 3 et 12. Après l'entrainement on obtient des nouveaux paramètres $\lambda' = (\Pi', A', B')$ pour chaque geste. Ces nouveaux paramètres sont les entrées des algorithmes Forward et Viterbi pour le test. Pour un vecteur d'entrée discret, l'algorithme Forward calcule la probabilité de son appartenance à chaque MMC en changeant le nombre des états. Ainsi le chemin de Viterbi reconnu correspond à la probabilité maximale entre les cinq MMCs.

Apprentissage et reconnaissance

Le problème connu dans l'utilisation des HMMs est la construction du modèle. On peut distinguer deux cas de figure différents, suivant que la structure (le nombres d'états du HMM et les transitions autorisées) soit connue ou pas. Lorsque la structure est connue, le problème se traduit par un problème d'entraînement qui consiste à estimer les paramètres numériques, les distributions de probabilité de transition et

de génération (émission) de manière à mieux expliquer les séquences d'apprentissage. Dans le cas d'apprentissage à partir d'une structure connue, on dispose d'un ensemble d'apprentissage composé de séquences supposées représentatives des séquences que l'on souhaite modéliser. L'apprentissage est assuré, ici, par un algorithme d'apprentissage comme l'entraînement de Viterbi ou l'entraînement de Baum-Welch. Pour certaines applications, on ne dispose pas de connaissance suffisantes pour inférer naturellement la structure de HMM. L'apprentissage devient alors plus difficile. Il ne suffit pas de paramétrer une structure mais il faut également déduire cette structure à partir des exemples fournis. Pour ce faire, il existe des algorithmes comme l'apprentissage par généralisation et l'apprentissage par spécialisation [Laurent Bréhélin, 2010].

Citons les trois problèmes dits « classiques »traités par les MMCs :

1. Étant donnée une séquence d'observations $O = \{O_1, ..., O_T\}$ de taille T et un modèle $\lambda = (\pi, A, B)$, comment peut-on calculer efficacement la probabilité $P(O|\lambda)$ de l'apparition de cette séquence O connaissant le modèle λ ?
2. Étant donnée une séquence d'observations $O = \{O_1, ..., O_T\}$ de taille T et un modèle $\lambda = (\pi, A, B)$, quelle est la séquence d'états $Q = \{q_1, ..., q_T\}$ qui explique le mieux l'observation ?
3. Comment ajuster le modèle $\lambda = (\pi, A, B)$ afin qu'il explique le mieux une séquence d'observation O, c'est-à-dire qu'il maximise $P(O|\lambda)$?

L'apprentissage d'un HMM consiste donc à résoudre le problème 3 en modifiant itérativement π, A et B afin de maximiser $P(O|\lambda)$, et ce pour chaque séquence d'observations O que contiendra le corpus d'apprentissage. Pour ce faire, il faut résoudre également le problème 2, c'est-à-dire trouver le chemin dans le modèle qui explique le mieux l'observation afin de pouvoir le modifier si cela permet d'augmenter $P(O|\lambda)$. La résolution du problème 1 est obligatoire puisqu'il est nécessaire d'évaluer le modèle pour pouvoir l'améliorer. Il existe plusieurs algorithmes pour faire tout cela : l'algorithme de Baum-Welch, l'algorithme Expectation-Maximization (ou EM) ou même simplement l'algorithme Viterbi. Voici une description du fonctionnement des trois algorithmes utilisés : Baum-welch pour l'entrainement, Forward pour l'évaluation et Viterbi pour le décodage.

* L'algorithme Baum-Welch : étant donné un ensemble d'apprentissage composé de séquences représentatives des séquences que l'ont veut modéliser, le but est

d'estimer les paramètres $\lambda' = (A', B')$ d'un modèle H dont les paramètres originaux sont $\lambda = (A, B)$ dans le but de maximiser l'équation suivante (maximiser les probabilités de génération) :

$$P(O|H) = \Pi_{k=1}^{K} P(O^k|H) \tag{4.4}$$

Les entrées de la nouvelle matrice de transition A' sont données par :

$$A'_{ij} = \frac{E\left[nombre\ de\ transitions\ de\ l'état\ s_i\ à\ s_j\right]}{E\left[nombre\ de\ transitions\ de\ l'état\ s_i\right]} = \frac{\sum_{k=1}^{t-1} \xi(s_i, s_j)}{\sum_{k=1}^{t-1} \gamma_k(s_i)} \tag{4.5}$$

Les entrées de la nouvelle matrice d'observation B' sont données par :

$$B'_{im} = \frac{E\left[nombre\ de\ génération\ du\ symbole\ m\ par\ s_i\right]}{E\left[nombre\ des\ fois\ s_i\right]} = \frac{\sum_{k=1}^{t} \gamma_k(s).1(z_k = m)}{\sum_{k=1}^{t} \gamma_k(s)} \tag{4.6}$$

Le nouveau modèle est calculé de telle sorte que :

$$P(O|\lambda') \geq P(O|\lambda) \tag{4.7}$$

∗ L'algorithme Forward : étant donnée un modèle H de paramètres $\pi = (A, B)$, on veut calculer la probabilité de générer la séquence de symboles $O = o_1, o_2, ..., o_t$ à l'aide de H. La variable Forward est définie par la probabilité d'avoir généré la séquence O partant de l'état initial et arrivant à l'état i à l'instant t :

$$\alpha_t(i) = P(o_1, o_2, ..., o_t, s_t = i|\lambda) \tag{4.8}$$

Ensuite, on peut constater que la relation récursive suivante devient :

$$\alpha_{t+1}(i) = b_j(\alpha_{t+1}) \sum_{i=1}^{N} \alpha_t(i)\alpha_{ij}, 1 \leq j \leq N, 1 \leq t \leq T-1 \tag{4.9}$$

avec,

$$\alpha_1(j) = \pi_j b_j(o_1), 1 \leq j \leq N \tag{4.10}$$

Par ailleurs, on peut facilement écrire la relation récursive

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{4.11}$$

∗ L'algorithme de Viterbi :

L'initialisation : pour $1 \leq i \leq N$,

$$\delta_1(i) = \Pi_i.b_i(o_1) \tag{4.12}$$

$$\phi_1(i) = 0 \tag{4.13}$$

La récursion : pour $2 \leq t \leq T$;$1 \leq j \leq N$;

$$\delta_t(i) = max\left[\delta_{t-1}(i).a_{ij}\right].b_i(o_t) \tag{4.14}$$

$$\phi_t(i) = argmax\left[\delta_{t-1}(i).a_{ij}\right] \tag{4.15}$$

La terminaison :

$$p^* = max\delta_T(i) \tag{4.16}$$

$$q_T^* = argmax\delta_T(i) \tag{4.17}$$

La reconstruction : pour $t = T-1, T-2, ..., 1$

$$q_t^* = \phi_{t+1}(q_{t+1}^*) \tag{4.18}$$

La trajectoire résultante de la séquence optimale des états est $q_1^*, q_2^*, ..., q_T^*$ avec a_{ij} la transition de l'état i à l'état j, $b_j(o_t)$ la probabilité de générer le symbole o à l'instant t par l'état j, $\delta_t(i)$ représente la valeur maximale de l'état j à l'instant t, $\phi_t(j)$ est l'index de l'état j à l'instant t et p^* est la fonction de vraisemblance de l'état optimisé.

4.2.2 Résultats expérimentaux

Afin de déterminer le nombre final des états exigés pour chaque MMC, nous l'avons varié et, d'après les résultats des expérimentations, nous avons retenu et fixé pour chaque MMC le nombre d'état qui fournit le meilleur taux de reconnaissance. Nous avons trouvé que le taux de reconnaissance est le plus élevé quand le nombre d'états est égale à 11 pour les gestes *viens* et *recule* ainsi que *pointage à droite*, 12 pour le geste *pointage à gauche*, et 8 pour le geste *stop* (figure 4.3). Par conséquent, nous utilisons ces paramètres dans la suite de notre approche.

Les résultats de reconnaissance sont présentés dans les tableaux 4.1 et 4.2. Le tableau

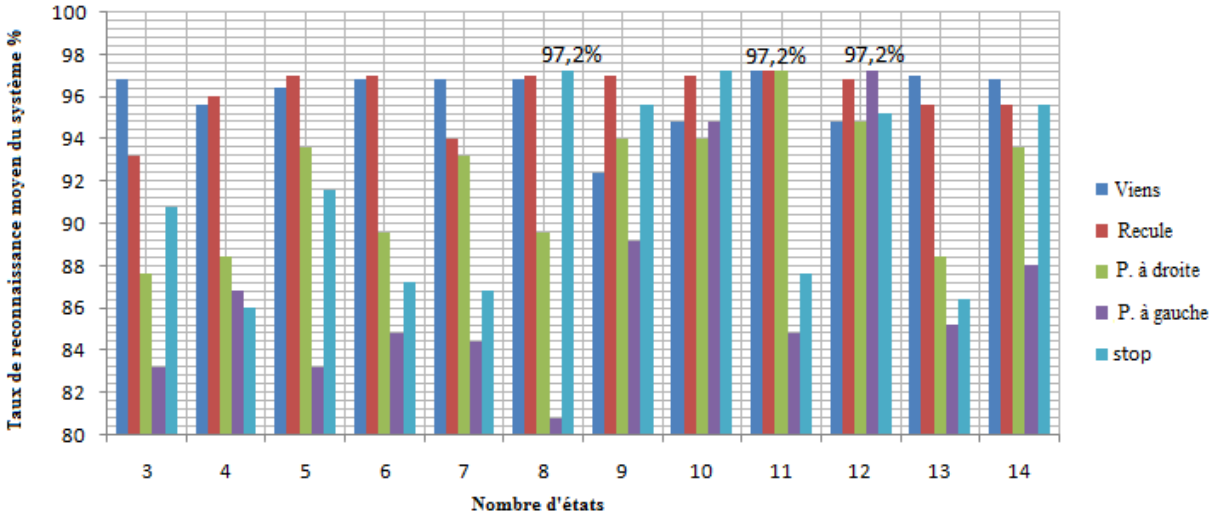

FIGURE 4.3 – Le taux de reconnaissance moyen du système de reconnaissance en variant le nombre d'états des cinq MMCs de 3 à 14 états.

4.1 correspond à la matrice de confusion entre les gestes. Pour le premier geste, il n'y a aucune mauvaise classification tandis que pour les autres on trouve 1 mauvaise reconnaissance pour le geste 3, 2 pour le geste 4 et 4 pour le geste 5. On peut remarquer que la méthode proposée donne de très bons résultats de reconnaissance et notamment pour les gestes opposés comme *viens* et *recule*, *pointage à droite* et *à gauche*. Ceci revient au fait que, dans les deux premiers gestes, l'angle qui change est le même mais il varie dans deux sens différents ; il décroît dans le geste *viens* et accroît dans *recule*. Le même raisonnement peut être donné pour les deux autres gestes. Le tableau

TABLE 4.1 – La matrice de confusion et la précision des différents gestes avec entraînement.

	viens	recule	pointage à droite	pointage à gauche	stop	Précision
viens	50	0	0	0	0	100%
recule	0	50	0	0	0	100%
pointage à droite	0	0	49	0	1	98%
pointage à gauche	0	0	0	48	2	96%
stop	1	0	0	3	46	92%
Précision moyenne 97,2%						

4.2 représente une comparaison entre notre méthode et celle présentée dans l'article [Gu et al., 2012]. Les auteurs de cet article utilisent l'orientation des articulations *coude* et *épaule* du bras gauche pour caractériser les gestes. Leur base de données contient cinq gestes. Ils ont entraîné leur base de données avec une seule personne et

la testent avec deux personnes. La durée des gestes est fixée au préalable. Le taux de reconnaissance en mode hors ligne et avec les participants à l'entrainement est de 85% pour leur méthode et 97,2% pour notre méthode. Le taux de reconnaissance pour les non participants à l'entrainement est de 73% pour leur méthode et 82% pour la nôtre.

Les gestes que nous avons défini pour l'interaction homme-système sont naturels. Ils représentent des gestes utilisés au quotidien. Par contre, la plupart des méthodes dans l'état de l'art sont basées sur des gestes contraignants en utilisant des signes sous forme de gestes statiques (pouce en haut, pouce en bas...) ce qui rend l'interaction moins naturelle.

TABLE 4.2 – La comparaison des performances de notre méthode avec la méthode de l'article [Gu et al., 2012] (N. de P. : Nombre de personnes).

Méthodes	[Gu et al., 2012]	Notre méthode
Nature des gestes	Dynamique	Dynamique
Info. utilisées	Angles des articulations par rapport au torse	Angles intérieurs des articulations
N. des gestes	5	5
N. d'articulations	2	5
Données utilisées	Segmentées	Brutes
Classification	MMC	MMC
La base d'entraînement	75	500
N. de p. pour le test	2	20
Durée de geste	Fixe	Variable
Taux de recon. avec entraînement	85%	97,2%
Taux de recon. sans entraînement	73%	82%

Bien que la méthode de reconnaissance proposée donne un taux de reconnaissance élevé, il y a quelques gestes mal reconnus (7 gestes de test sur 200). Pour pallier ce problème nous avons proposé une extension de la méthode dans le but de rejeter les gestes mal reconnus en fixant des seuils. La section suivante détaille la solution proposée.

4.3 La méthode basée sur MMC et DTW

4.3.1 Formalisme

Dans cette section, nous proposons une méthode de classification robuste qui combine les MMCs avec le DTW afin d'éliminer les mauvaises classifications des gestes en les rejetant.

L'algorithme DTW

La Déformation Temporelle des Données (DTW) et un algorithme bien connu qui vise à comparer et à aligner deux séquences de données (dites aussi séries temporelles). Bien qu'il ait été développé à l'origine pour la reconnaissance vocale [1], il a également été appliqué dans de nombreux autres domaines à savoir la bioinformatique, l'économétrie et la reconnaissance d'écriture. Considérons deux séquences A et B, composées respectivement de n et m vecteurs de caractéristiques.

$$A = a_1, a_2, ..., a_i, ..., a_n \tag{4.19}$$

$$B = b_1, b_2, ..., b_j, ..., b_m \tag{4.20}$$

Chaque vecteur de caractéristiques est de dimension d et peut donc être représenté par un point dans un espace de dimension d. Par exemple dans la reconnaissance d'écriture, nous pouvons utiliser directement l'état brut des coordonnées du mouvement du stylo (x, y) ce qui va créer des séquences avec des vecteurs de deux dimensions. Il est également intéressant de noter que les séquences A et B peuvent être de longueur différente. DTW fonctionne par la déformation (d'où le nom) de l'axe du temps de manière itérative jusqu'à ce qu'une adéquation optimale entre les deux séquences soit trouvée.

La figure 4.4 représente un exemple de deux séquences de données à une dimension. L'axe du temps est déformé de sorte que chaque point de données dans la séquence verte soit orienté de manière optimale vers un point dans la séquence bleue. Nous pouvons ainsi construire une matrice de distance $n * m$ dont chaque cellule (i, j) représente la distance entre l'élément de rang i de la séquence A et l'élément du rang j de la séquence B (foir figure 4.5). La distance métrique utilisée dépend de l'application, une mesure communément utilisée la distance euclidienne. Dans chaque

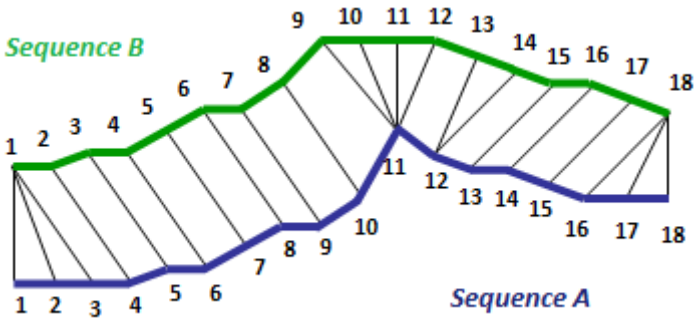

FIGURE 4.4 – Alignement de deux séquences avec DTW.

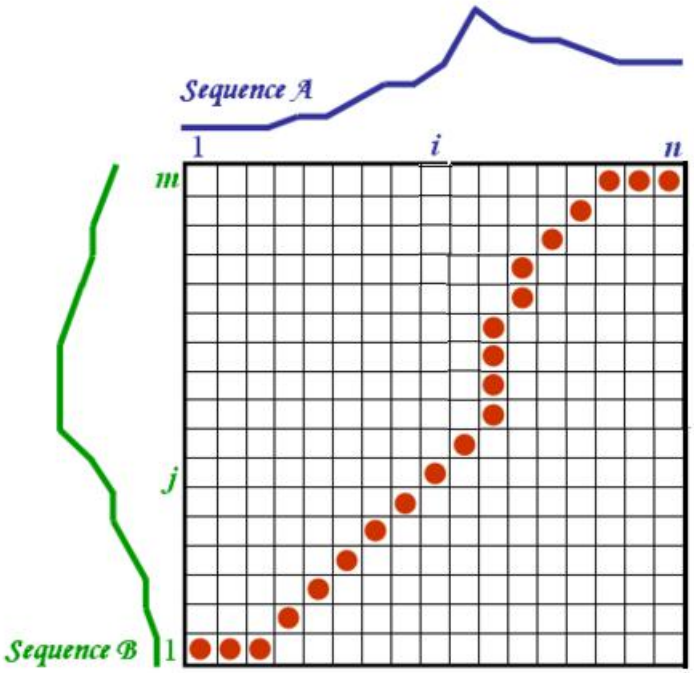

FIGURE 4.5 – Matrice de distance DTW entre deux séquences.

cellule de la matrice, une mesure de distance peut être placée en comparant les éléments correspondant de chaque séquence. Trouver le meilleur alignement entre deux séquences peut être vu comme déterminer le chemin le plus court pour aller de la cellule en bas à gauche à la cellule en haut à droite de cette matrice. La longueur d'un chemin est simplement la somme de toutes les cellules qui ont été visitées le long de ce chemin. Plus le chemin optimal est loin de la diagonale, plus les deux séquences doivent être déformées pour correspondre l'une à l'autre. Afin de limiter le nombre de chemins à explorer, DTW peut imposer plusieurs types de contraintes raisonnables, à savoir :

* La monotonie : le chemin d'alignement ne fait pas de retour en arrière. Les rangs i et j restent augmentent ou demeurent les mêmes, mais en aucun cas diminuent. Cela garantit que la même donnée ne soit pas répétée dans l'alignement.
* La continuité : le chemin d'alignement avance d'un seul pas à la fois en commençant de la cellule en bas à gauche et se terminant par la cellule en haut à droite. Les indices i et j peuvent augmenter d'un pas au maximum, cela garantit qu'aucune donnée ne soit omise.
* La contrainte sur la limite (la frontière) : L'alignement commence au bas à gauche et se termine en haut à droite. Cela garantit que les séquences ne sont pas considérées que partiellement mais dans leur totalité.
* La contrainte sur la fenêtre de déformation : Un bon chemin d'alignement a une faible probabilité de se trouver trop loin de la diagonale. Cela garantit que l'alignement ne cherche pas à sauter les données différentes ou coller à des données similaires.
* La contrainte sur la forme : Les chemins alignés ne doivent pas être trop raid ou trop peu profond. La condition peut être exprimée par un ratio p/q où p est le nombre permis des pas dans la même direction (horizontalement ou verticalement). Après avoir avancé de p pas dans la même direction, il n'est pas permis d'avancer plus dans la même direction qu'après avoir avancé au moins de q pas dans la direction de la diagonale.

Les contraintes précédentes permettent de limiter les déplacements qui peuvent être faites de n'importe quel point dans le chemin et ainsi limiter le nombre de chemins à considérer. Le pouvoir de l'algorithme DTW est dans le fait qu'au lieu de trouver tous les parcours possibles, par la matrice, qui satisfont les conditions ci-dessus, DTW permet de garder la trace du coût du meilleur chemin à chaque point dans la matrice. Pendant le processus de calcul de la matrice de distance DTW, on ne connaît pas le chemin global de la distance minimale, mais celui ci peut être retracé quand le point final est atteint.

Profitant de ces contraintes, DTW utilise la programmation dynamique pour trouver le meilleur alignement de manière récursive. A l'origine, la cellule (i, j) de la matrice de distance a été définie comme « la distance entre l'élément de rang i de la séquence A et l'élément j de la séquence B ». Dans le cas de la programmation dynamique, cette définition a été modifiée. Ainsi, la cellule (i, j) est définie comme la longueur la

plus courte du trajet amenant jusqu'à cette cellule. La cellule (i, j) peut être définie de manière récursive, comme suit :

$$Cell(i,j) = distance\_locale(i,j) + MIN\left[cell(i-1,j), cell(i-1,j-1), cell(i,j-1)\right] \tag{4.21}$$

Ici, la récursivité signifie que le chemin le plus court jusqu'à la cellule (i, j) est défini en fonction du plus court chemin jusqu'aux cellules adjacentes. Une fois que l'algorithme a atteint la cellule haute de droite, nous faisons un alignement dans le sens inverse, c'est-à-dire de la cellule haute de droite vers la cellule de bas à gauche. Nous considérons au final le meilleur des deux alignements. Cependant, si on s'intéresse uniquement à la comparaison entre séquences, alors la cellule en haut à droite de la matrice correspondra à la longueur du plus court chemin. On peut donc utiliser la valeur stockée dans cette cellule comme la distance entre les deux séquences. DTW a la propriété intéressante d'être symétrique et donc :

$$DTW(A,B) = DTW(B,A) \tag{4.22}$$

La combinaison MMC/DTW

Comme dans la précédente méthode, les variations des angles durant l'exécution d'un geste est utilisée comme entrée pour les MMCs. Ensuite, la sortie de ces derniers est donnée comme entrée à DTW dans le but de mesurer la distance entre le geste de test et la séquence de référence du modèle MMC. La décision finale est trouvée en comparant la distance calculée par DTW et le seuil qui correspond au modèle. Lors du calcul de la distance, nous ne prenons pas en compte toute la séquence du geste mais uniquement la partie des variations de l'angle principal qui caractérise le geste dont le modèle a été trouvé par MMC. La distance DTW est calculée entre cette partie du geste et la même partie dans une séquence de référence. Ensuite, cette distance est comparée à un seuil prédéfini (sous-section suivante). Ainsi, si elle est inférieure au seuil nous retenons le résultat fourni par le MMC en considérant que le geste a été bien reconnu. Dans le cas contraire, le geste est rejeté et considéré inconnu. Par conséquent, un geste mal exécuté par l'utilisateur va être rejeté au lieu d'être mal classé. La figure 4.6 montre les étapes du traitement proposé. Tout d'abord, la méthode MMC classe un geste de test (Gtest) dans une classe parmi les cinq.

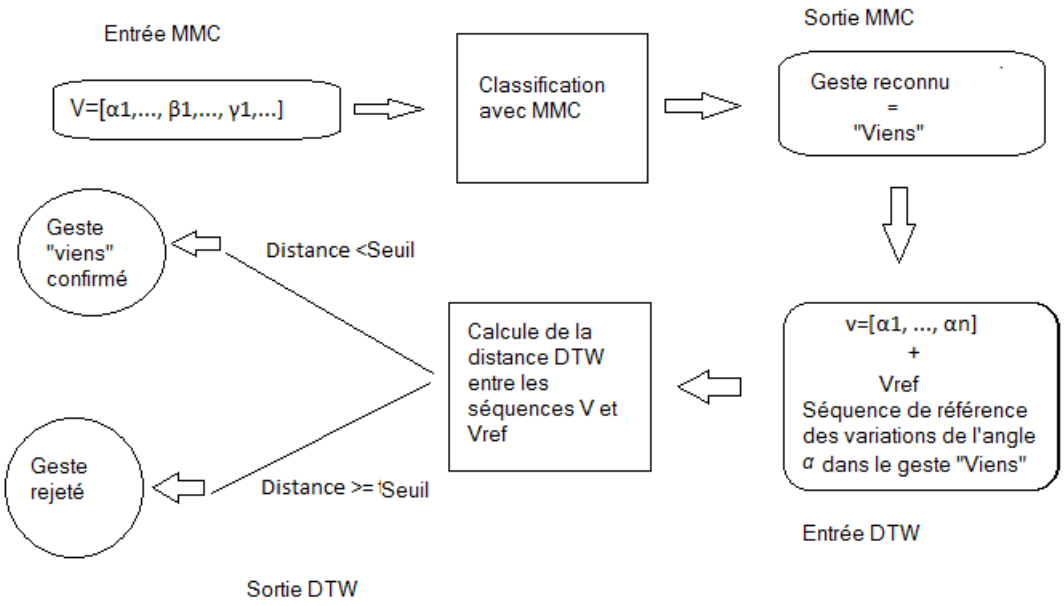

FIGURE 4.6 – Combinaison MMC et DTW pour la reconnaissance de gestes.

Après, la méthode donne le résultat c'est à dire le type du geste (ou bien la classe à laquelle il appartient) par exemple : *Viens*. Comme mentionné auparavant, l'angle qui caractérise le geste *Viens* est l'angle du coude désigné par α. Ainsi, nous prenons la première partie de la séquence du geste (Gtest) qui correspond aux variations de l'angle α, et nous prenons aussi la même partie d'un geste *Viens* de référence. Ensuite, nous calculons la distance euclidienne entre ces deux séquences en utilisant DTW. La distance résultante est comparée au seuil prédifini pour le geste *Viens*. L'algorithme 1 représente les instructions réalisées pour trouver la distance minimale entre une séquence de test T et des séquences de référence R_i. Chaque classe de geste lui est associée une séquence de référence qui s'agit d'un geste exécute une seule fois. Cet algorithme trouve, parmi les séquences des références R_i, celle la plus proche de la séquence de test T, en cherchant la distance DTW minimale entre elle $dist_{DTW(R_j,T)}$.

Le calcul des seuils

Nous avons utilisé DTW pour mesurer les distances entre les différentes séquences. Les séquences que nous utilisons contiennent des données d'une seule dimension car chaque donnée représente une valeur d'angle. Nous avons calculé empiriquement cinq

Algorithm 3 Calcul de la distance DTW minimale

```
Début
séquences de référence R_i, séquence de test T
dist_min ← dist_DTW(R1,T)
for j=2 to i do
      if dist_DTW(Rj,T) < dist_min
        dist_min ← dist_DTW(Rj,T)
        indice_référence ← j
      end if
end for
Fin
```

seuils, un pour chaque classe de geste. Tout d'abord, nous considérons pour chaque geste la séquence de référence qui le représente. Les séquences de référence des classes *Viens*, *Recule*, *Pointage à droite*, *Pointage à gauche* et *Stop* contiennent respectivement les variations de l'angle α, α, β, β et γ durant l'exécution du geste. Le seuil de chaque classe correspond à la distance maximale entre sa propre séquence de référence et 50 séquences de test. La distance est donnée par DTW et les séquences de test sont extraites de la base d'entraînement. Le tableau 4.3 montre la valeur du seuil calculé pour chaque classe :

TABLE 4.3 – Les seuils des classes

Type du geste	Le seuil calculé
Viens	52800
Recule	29400
Pointage à droite	85800
Pointage à gauche	39101
Stop	29100

4.3.2 Résultats

Les résultats expérimentales sont donnés dans le tableau 4.4. Le taux de reconnaissance peut toujours atteindre 100% pour certain types de geste comme *Viens* et *Recule* et la combinaison des MMCs avec DTW évite la mauvaise classification. La méthode proposée détecte les gestes faux positifs et les rejette comme étant gestes inconnus (7 différents faux positifs). Le taux de mauvaise reconnaissance du système tombe donc à 0%.

TABLE 4.4 – La matrice de confusion de la méthode MMC/DTW

	Viens	Recule	P. à droite	P. à gauche	Stop	Rejeté	Précision
Viens	50	0	0	0	0	0	100%
Recule	0	50	0	0	0	0	100%
P. à droite	0	0	49	0	0	1	98%
P. à gauche	0	0	0	48	0	2	96%
Stop	0	0	0	0	46	4	92%
Précision moyenne 97.2%							

4.4 Conclusion

Dans ce chapitre, nous avons présenté une approche de reconnaissance des gestes en combinant les Modèles de Markov Cachés et la Déformation Temporelle Dynamique. La méthode proposée permet de résoudre le problème de la mauvaise classification que nous avions rencontré avec la précédente méthode de reconnaissance. L'idée est de faire un test avec DTW sur le résultat des MMCs afin de rejeter les gestes faux positifs au lieu de les mal classer. D'après les résultats des expérimentations conduites, la méthode est avérée efficace et élimine toutes les mauvaises classifications.

Chapitre 5

Reconnaissance de gestes composés

5.1 Introduction

Dans ce chapitre nous traitons le problème de la reconnaissance des gestes continus. Dans ce cas, nous proposons de combiner la Déformation Temporelle des Données (DTW) avec une fenêtre adaptative. La nouvelle approche ainsi développée portera le nom de « Déformation Temporelle Adaptative des données (ADTW) ». Cette méthode permettra de reconnaître deux gestes successifs ou plus dans la même séquence. Nous avons proposé deux versions de la méthode ADTW afin de traiter deux cas de reconnaissance : cas(1) reconnaissance en ligne et cas(2) reconnaissance hors ligne. Nous rappelons que dans la reconnaissance en ligne, les traitements doivent se faire en parallèle de l'acquisition. Afin de traiter ce cas, nous avons proposé la version »Forward Adaptative Dynamic Time Warping « dont le traitement ne fait qu'avancer dans le temps. Tandis que la deuxième version est »Backward Adaptative Dynamic Time Warping « qui est combinée avec la première version pour traiter le cas de la reconnaissance hors ligne.

5.2 La méthode ADTW

5.2.1 DTW pour la reconnaissance de gestes

La classification en utilisant DTW n'exige pas une base de données. En effet, il suffit d'avoir une bonne séquence de référence qui représente le mieux possible le geste. La classification est réalisée comme suit : chaque séquence de test est comparée à toutes les séquences de références. Dans chaque comparaison nous obtenons une distance. Ainsi, la séquence de test appartient à la classe représentée par la séquence de référence qui donne la distance minimale.

Afin d'évaluer notre méthode, nous avons construit un programme de simulation pour la reconnaissance en ligne. Dans les applications en ligne, la reconnaissance procède en acquérant progressivement les nouvelles données. Cependant, il est nécessaire de stocker un nombre suffisant de données entrantes dans un buffer avant de commencer la reconnaissance. En effet, afin de débuter le processus de la reconnaissance de gestes, on a besoin d'un minimum de données sur lesquelles le traitement sera effectué. Ce nombre de données peut être choisi selon la nature et la durée du type de geste à reconnaître. Afin de déterminer ce nombre minimum de données, nous avons fixé un taux de reconnaissance de 80%. Nous avons réalisé un programme qui incrémente, progressivement, le nombre de données à stocker et qui calcule ensuite le nombre de gestes bien reconnus en utilisant l'algorithme classique DTW. Le traitement a été effectué sur toutes les séquences de la base de données à savoir 500 séquences. La figure 5.1 montre les résultats des tests qui ont été conduits. Selon ces résultats, le nombre minimal de données nécessaires pour avoir 80% de reconnaissance est égale à 50 données. Nous avons utilisé ce résultat dans les expérimentations. Ainsi, dans chaque test, nous avons commencé la phase de reconnaissance après avoir acquis les 50 premières données.

5.2.2 La fenêtre adaptative

La méthode ≪ Déformation Temporelle Dynamique Adaptative ≫est la combinaison de la méthode ≪ Déformation Temporelle Dynamique ≫avec une fenêtre glissante adaptative. Une fenêtre adaptative est une fenêtre dont les paramètres changent selon les données entrantes et le processus de reconnaissance. Ces paramètres sont : $start_{t_i}$, end_{t_i}, et $length_{t_i}$, qui désignent respectivement, l'indice de début, l'indice de

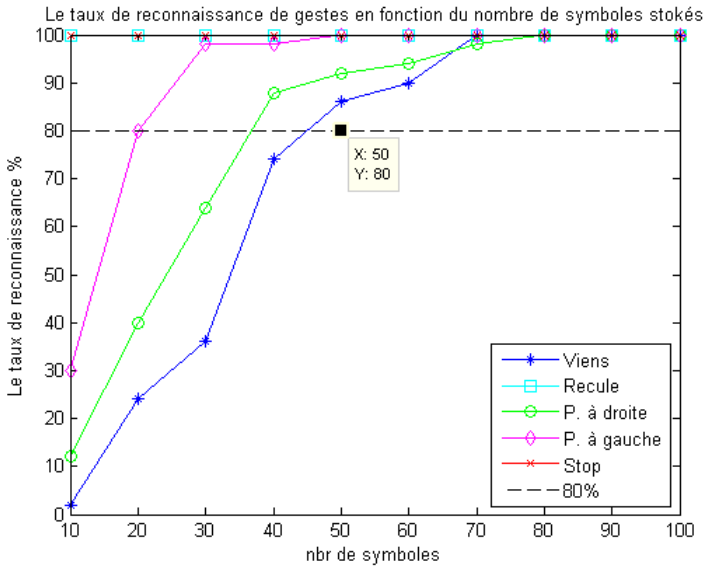

FIGURE 5.1 – Le nombre des données nécessaires pour une bonne reconnaissance.

fin et la longueur de la fenêtre à l'instant t_i.

$$Window_{t_i} = Window(Start_{t_i}; End_{t_i}; Length_{t_i}) \tag{5.1}$$

Nous utilisons la fenêtre adaptative dans les deux versions, *Forward* et *Backward*. Dans la première version (Forward) de l'approche ATDW proposée, l'indice de début est fixé tandis que l'indice de fin s'incrémente avec le temps. Dans la deuxième version (Backward) l'indice de fin est fixé et celui de début se varie.

5.2.3 ADTW pour la reconnaissance de gestes

Reconnaissance en ligne

La reconnaissance du geste composé via notre nouvelle méthode ADTW ne nécessite pas une base d'entraînement. Cependant, elle exige des séquences de référence qui représentent parfaitement les gestes simples qui composent le geste continu. La reconnaissance d'un geste simple via DTW est donnée dans le chapitre 4. Comme le geste composé contient une série de gestes simples, nous avons besoin de chercher ces derniers dans la séquence du geste composé en faisant un nouveau test après chaque nouvelle donnée acquise.

Tout d'abord, nous considérons R_i la séquence de référence du geste i, et S_c la

séquence du geste composé qui contient deux gestes ou plus. Afin de reconnaître, dans l'ordre, les gestes composant la séquence continue, nous allons, dans un premier lieu, tester la première partie de la séquence en la considérant comme étant un geste. Pour ce faire, nous enregistrons les premières données de la séquence S_c dont le nombre a été fixé préalablement (50 données), ensuite nous calculons la distance DTW entre cette partie et chacune des séquences de référence. Nous gardons la distance minimale et la séquence de référence liée à cette distance. Initialement, nous considérons que le premièr geste de la séquence appartient à la même classe de gestes représentée par la séquence de référence trouvée. Ensuite, nous faisons un traitement progressif pour reconnaître les gestes qui suivent dans la séquence composée. L'idée principale est d'appliquer une fenêtre glissante adaptative dont la taille s'adapte aux nouvelles données. Le but est de trouver la distance DTW minimale entre la séquence de test et la séquence de référence, en alimentant la fenêtre à chaque fois par une nouvelle donnée et en augmentant sa taille. La méthode utilisée dans ce cas est la version *Forward* de la méthode ADTW, qui est détaillée dans l'algorithme A suivant. Afin de détecter la fin du premier geste et le début du geste suivant, nous faisons une boucle où dans chaque itération, nous augmentons la taille de la fenêtre en l'alimentant avec une nouvelle donnée, et nous recalculons la distance DTW entre la séquence définit par la nouvelle fenêtre, et la même séquence de référence. Si la nouvelle distance calculée est inférieure à la précédente, alors cela voudra dire que le geste actuel n'est pas encore terminé. En effet, en ajoutant de nouvelles données la séquence de test devient de plus en plus proche de la séquence de référence, ce qui fait que la distance entre elles diminue. Dans ce cas, nous continuons à incrémenter la taille de la fenêtre et l'alimenter avec de nouvelles données. Quand la valeur de la distance calculée devient supérieure à celle qui précède, cela veut dire que nous commençons à s'éloigner du geste de référence et que les nouvelles données ne font plus partie de ce geste mais du nouveau geste. Dans ce cas, nous réinitialisons la fenêtre adaptative et l'alimentons avec les 50 nouvelles données et nous répétons les mêmes traitements. La figure 5.2 montre un exemple de reconnaîssance en ligne d'un geste simple dans une séquence. Dans l'exemple de la figure 5.2, nous pouvons conclure que le premier geste commence par la donnée s_1 et se termine par la donnée s_k, et le deuxième geste commence par la donnée s_{k+1}. Ainsi, effectuant le même traitement tout au long de la séquence du geste composé, nous pouvons reconnaître tout les

Algorithm 4 Forward Adaptive Dynamic Time Warping Algorithm

Données : séquences de référence R_i, séquence composée test T de taille m
$a \leftarrow 1$
$b \leftarrow 50$
$n \leftarrow b$
while $n \leq m$ **do**
    $Fenêtre \leftarrow T(a; n)$
    $dist_{min} \leftarrow dist_{DTW(R_1, Fenêtre)}$
    for j=2 to i **do**
        If $dist_{DTW(R_j, Fenêtre)} < dist_{min}$
            $dist_{min} \leftarrow dist_{DTW(R_j, Fenêtre)}$
            $indice_{référence} \leftarrow j$
        End If
    end for
    while $(dist_{DTW(R_j, Fenêtre)} \leq dist_{MIN})$ **do**
        $dist_{min} \leftarrow dist_{DTW(R_j, Fenêtre)}$
        $n \leftarrow n + 1$
        $Fenêtre \leftarrow T(a; n)$
    Else
        $indice_{fin} \leftarrow n$
        $Geste \leftarrow T(a; n)$
        $a \leftarrow n$
        $n \leftarrow n + 50$
    End While
    End While

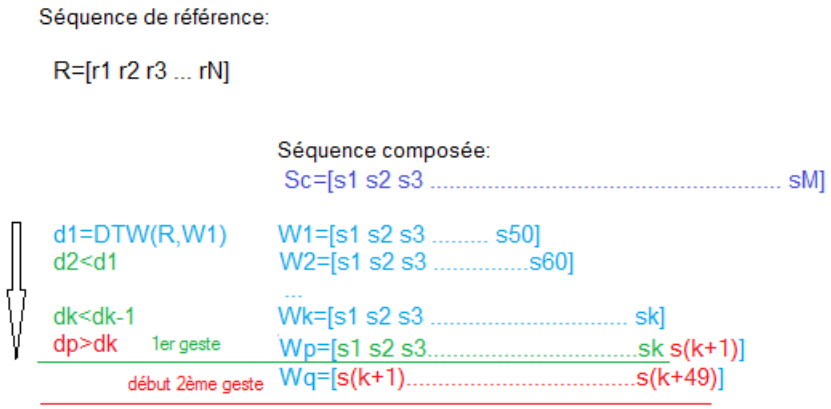

FIGURE 5.2 – La méthode *Foward Déformation Temporelle Dynamique Adaptative.*

gestes qui le constituent.
généralement, dans la reconnaissance en ligne, on essaye de réduire le temps des traitements pour que la reconnaissance se fasse au fur et à mesure de l'acquisition du flux. Dans ce cas on ne s'intéresse qu'aux nouvelles données. La démarche directe (*Forward*) que nous avons proposée détecte la fin du geste en cours mais ne peu pas détecter exactement son indice de début. Par conséquent, les données de transition entre deux gestes successifs ne peut être détectées dans la séquence composée. Ainsi, elles feront automatiquement partie du prochain geste, ce qui produira quelques cas de mauvaise classification. Par contre, dans le cas de la reconnaissance hors ligne, nous pouvons analyser les données de la séquences en effectuons des démarches indirectes afin de détecter le début exacte du geste et ainsi, détecter la partie de transition.

Reconnaissance hors ligne

Dans le cas de la reconnaissance hors ligne, nous effectuons les deux démarches *Forward* et *Backward* d'une manière respective. Dans un premier temps, nous effectuons la démarche *Forward*. Ensuite, nous faisons le traitement dans le sens inverse ce que nous avons appelé la démarche *Backward*. Après avoir reconnaître le geste avec la démarche *Forward* et récupérer sa séquence qui commence par des données de transitions puis le début du geste (que nous ne savons pas encore), le geste et sa fin que nous savons. Cette fois ci, on va décrémenter la taille de la fenêtre glissante en supprimant la première donnée et recalculer la distance DTW. Là encore, si la nouvelle distance et inférieure de la précédente, on continue à décrémenter. Lorsque la distance augmente, ce qui voudra dire que nous commençons à s'éloigner du geste de référence en supprimant les données qui en font partie. Dans ce cas, on s'arrête et en retient la taille et l'indice du début de la fenêtre courante ainsi que la distance DTW. L'algorithme 2représente les instructions réalisées dans la démarche *Backward* de la méthode ADTW proposée.

Ainsi, La première donnée de la fenêtre finale est considérée le début du geste et la données qui se trouvent entre la première donnée de la fenêtre d'entrée et la première donnée de la fenêtre de sortie sont considérées des données de transitions. Un exemple de ce traitement est donnée dans la figue 5.3.

Algorithm 5 Backward Adaptive Dynamic Time Warping Algorithm

Données : la séquence de référence R_j, la séquence du geste T_g trouvée par *Forward*.
$Fenêtre \leftarrow T_g(a;n)$
$dist_{min} \leftarrow dist_{DTW(R_j,Fenêtre)}$
while $(dist_{DTW(R_j,Fenêtre)} \leq dist_{MIN})$ **do**
    $dist_{min} \leftarrow dist_{DTW(R_j,Fenêtre)}$
    $a \leftarrow a + 1$
    $Fenêtre \leftarrow T(a;n)$
Else
    $indice_{début} \leftarrow a$
    $Geste \leftarrow T(a;n)$
End While

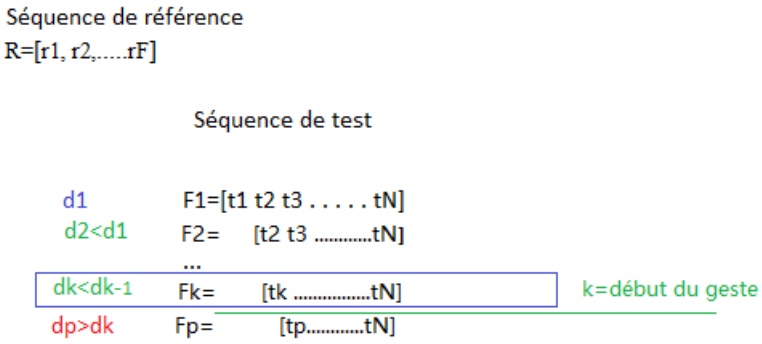

FIGURE 5.3 – La méthode *Backward Déformation Temporelle Dynamique Adaptative.*

5.2.4 Résultats expérimentaux

Nous avons étudié deux cas de gestes composés :

* cas 1 : la fin du premier geste est identique au début du second geste.
* cas 2 : la fin du premier geste est différente du début du second geste. Nous avons introduit des données de transition entre les deux gestes successifs.

Les tableaux 5.1 et 5.2 représentent le taux de reconnaissance de quelque gestes composés sans et avec transition. Comme on peut le constater, la reconnaissance des gestes basée sur la méthode ADTW donne de bons résultats. Cependant, le taux moyen de la reconnaissance des gestes composés avec transition est inférieur à celui des gestes composés sans transition. En effet, les données de transition ne sont pas éliminées par notre méthode. Ainsi, elles deviennent une partie du geste suivant. Par conséquent, quelques fausse-classifications de gestes simples qui composent la séquence continue peuvent se produire dans la reconnaissance en ligne. Notamment, quand la durée de la transition devient longue. Ici, nous donnons un exemple de séquences qui contiennent deux gestes successifs. Pourtant, notre méthode peut reconnaître un nombre illimité des gestes dans une même séquence tant que la reconnaissance se fait en ligne.

TABLE 5.1 – Le taux de reconnaissance des gestes composés sans transition.

Geste composé	**Taux de reconnaissance**
Stop + P. à gauche	100%
Recule + P. à gauche	98%
Recule + P. à droite	92%
Stop + P. à droite	86%
Stop + Recule	84%
Taux moyen 92%	

TABLE 5.2 – Le taux de reconnaissance des gestes composés avec transition.

Geste composé	**Taux de reconnaissance**
P. à gauche + Stop	92%
P. à droite + P. à gauche	84%
P. à gauche + P. à droite	84%
P. à gauche + Viens	80%
Recule + Stop	74%
Taux de reconnaissance 82.8%	

Gestes avec début ambigu

Dans le but d'analyser le comportement de notre système de reconnaissance dans le cas d'un début ambigu dans les gestes. Nous avons ajouter à des séquences des données aléatoires en les plaçant au début de chaque séquence. Les figures 5.4, 5.5, 5.6, 5.7, 5.8 et 5.9 représentent respectivement, le taux de reconnaissance des gestes *viens*, *recules*, *pointage à droite*, *pointage à gauche* et *stop* sans et avec un début ambigu. Prenons l'exemple du geste *viens* sur la figure 5.4. Dans les deux cas (avec et sans début ambigu) le système de reconnaissance confond les premières données du gestes avec d'autres classes. Mais, avec l'acquisition de plus de données le système commence à s'approcher de la vrai classe du geste. Cependant, dans le cas sans début ambigu le système avait besoin de trentaine de données afin de reconnaître définitivement le geste (la distance DTW diminue et la courbe du geste viens tend vers l'axe des abscisses) et ne remonte plus. Dans le cas avec début ambigu, le système avait besoin de plus de données, à savoir 50, pour bien connaître le geste *viens*.

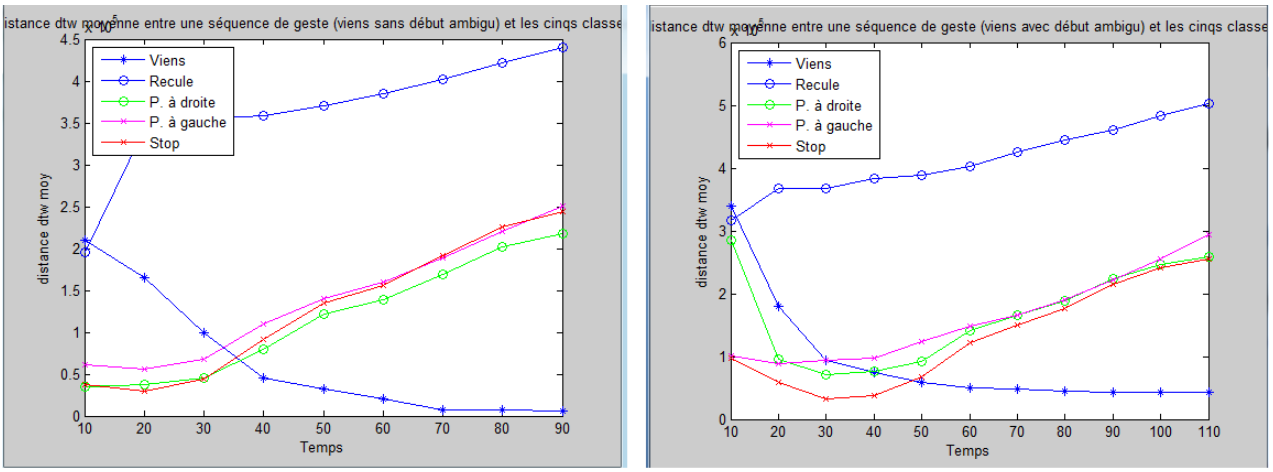

FIGURE 5.4 – Le taux de reconnaissance du geste *viens* sans et avec début ambigu

.

5.3 Conclusion

Dans ce chapitre, une méthode de reconnaissance de gestes composés a été présentée. La méthode proposée est une combinaison de la méthode DTW et une fenêtre glissante adaptative dont les paramètres varient au cours du processus de la reconnaissance. Deux version de cette méthode ont été proposée : (1) démarche *Forward* et

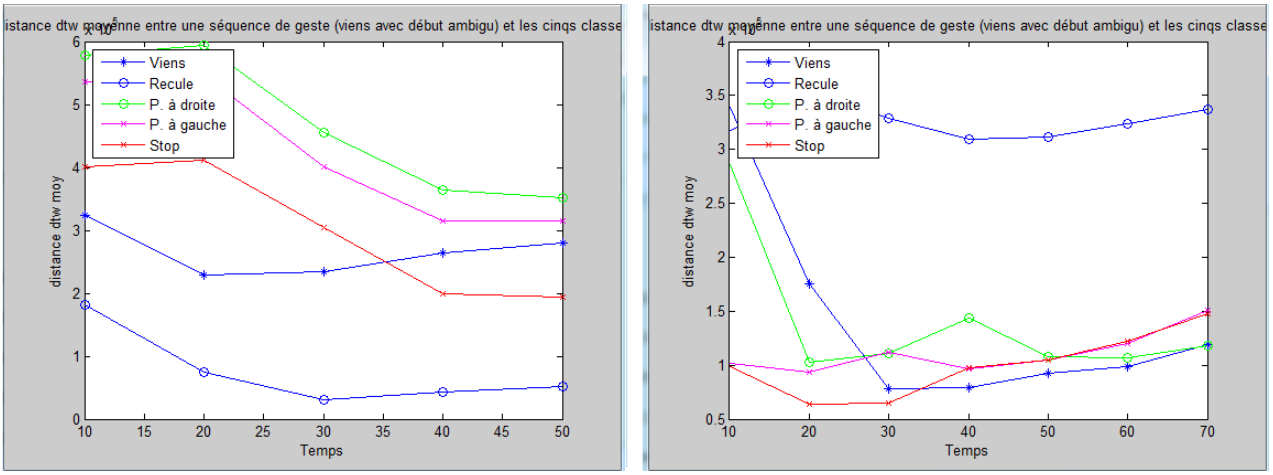

FIGURE 5.5 – Le taux de reconnaissance du geste *recule* sans et avec début ambigu

.

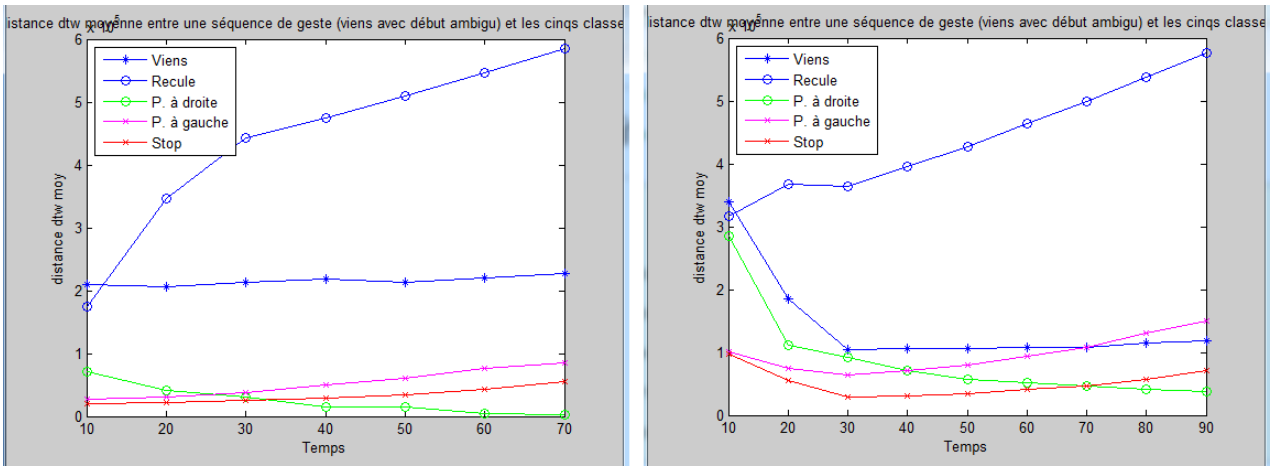

FIGURE 5.6 – Le taux de reconnaissance du geste *pointage à droite* sans et avec début ambigu

.

(2) démarche *Backward*. Deux cas de reconnaissance ont été traités : (1) la reconnaissance en ligne où la démarche *Forward* est utilisée (2) et la reconnaissance hors ligne où les deux démarches ont été utilisées. Aussi, nous avons testé notre approche sur deux types de gestes composés : (1) gestes composés avec transition et (2) gestes composés sans transition. Les résultats montrent que l'approche donne un bon taux de reconnaissance, cependant ce dernier diminue un peu dans le cas des gestes avec transition. En effet, les données de transition non supprimées affecte la reconnaissance du geste.

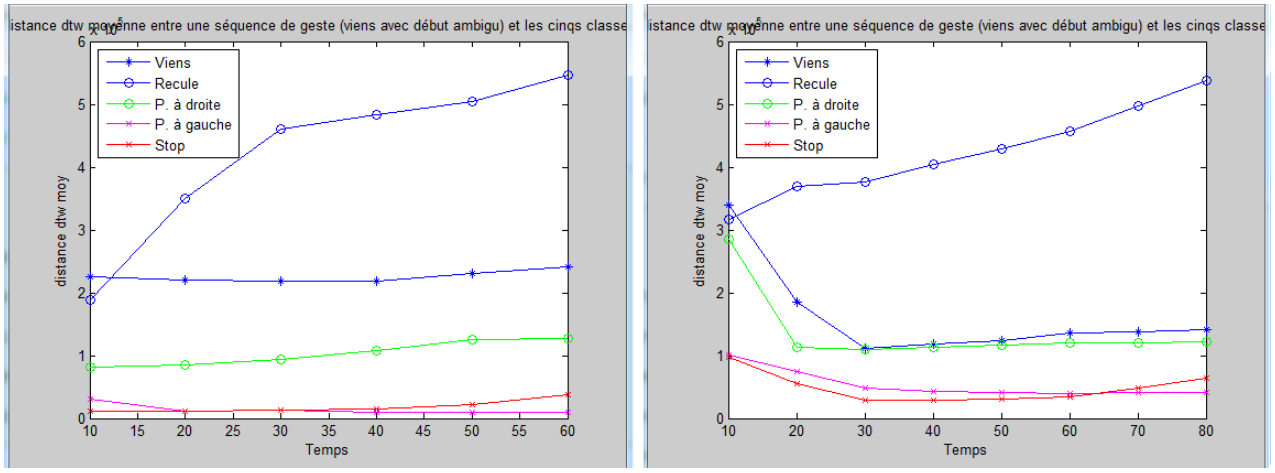

FIGURE 5.7 – Le taux de reconnaissance du geste *pointage à gauche* sans et avec début ambigu

.

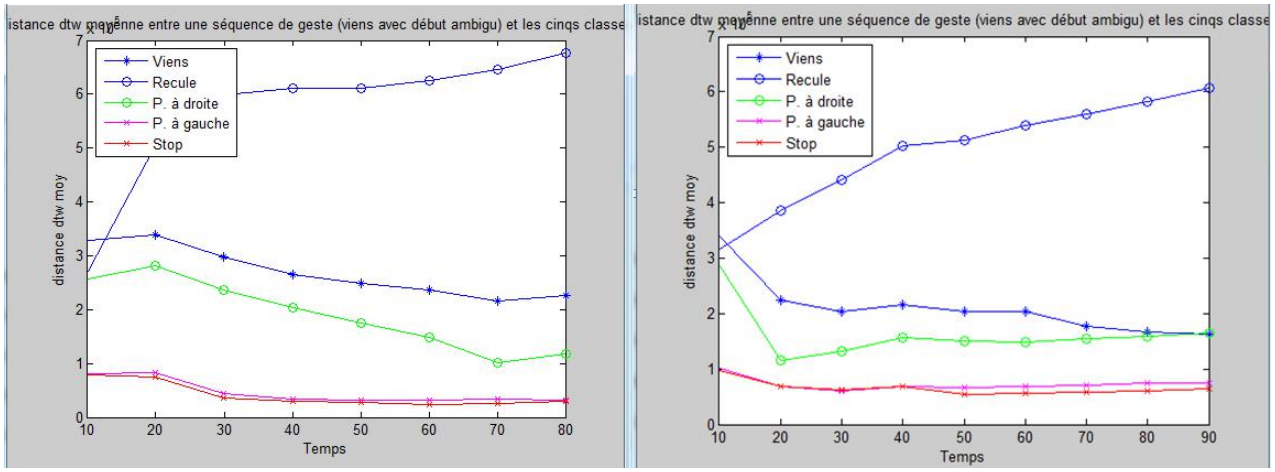

FIGURE 5.8 – Le taux de reconnaissance du geste *stop* sans et avec début ambigu

.

Conclusion et perspectives

Conclusion

Au cours de cette thèse, nous avons proposé un système de reconnaissance de gestes humains basé sur les données de profondeur. Nous avons validé nos approches sur une base de données que nous avons construite, et es résultats obtenus sont satisfaisants.Notre système de reconnaissance de gestes est robuste vis à vis des conditions réelles d'utilisation, il peut être adapté à la reconnaissance en ligne.

Le système proposé pour la reconnaissance de gestes implique trois étapes de traitement qui sont la génération de descripteurs de gestes, l'apprentissage et la classification de gestes. Notre approche a introduit une nouvelle représentation gestuelle qui est un descripteur 3D généré à partir des images de profondeur est constitué des variations des angles des articulations. De plus, elle utilise des méthodes de classification pour la reconnaissance des gestes adaptées à ces nouveaux descripteurs.

Nous avons présenté une méthode de classification de gestes simples. Cette méthode est basée sur les Modèles de Markov Cachés. En effet, nous avons proposé un modèle MMC pour chaque geste. Les modèles ont été entraînés par un nombre importants de séquences pour chaque type de geste. Chaque modèle est alimenté par un descripteur en entrée, et donne en sortie la classe à laquelle le geste en entrée appartient, en fonction de la probabilité de vraisemblance la plus élevée. Afin d'éliminer les cas de mauvaise classification, nous avons ajouter à cette méthode une règle de décision basé sur le seuillage. En effet, nous avons utilisé la méthode de Déformation Temporelle Dynamique pour calculer les seuils de rejet. Ainsi, le taux des mauvaises classifications a chuté à 0%.

Une autre méthode de classification a été proposée dans cette thèse. Il s'agit d'une méthode de classification pour la reconnaissance des gestes composés basée sur la

combinaison de la méthode Déformation Temporelle Dynamique avec une fenêtre glissante adaptative. Deux versions de cette méthode ont été proposées : (1) démarche directe *Forward*, et (2) démarche inverse *Backward* pour les deux cas de reconnaissance, en ligne et hors ligne. Nous avons aussi traité deux types de gestes composés à savoir, les gestes composés avec et sans transition. Citons les caractéristiques de notre système de reconnaissance de gestes :

* Premièrement, la phase d'entraînement est simple, il suffit d'enregistrer le geste lors de son exécution. La génération des descripteurs se fait en ligne en parallèle avec l'acquisition.
* deuxièmement, Le système peut reconnaître les gestes même si l'emplacement des personnes et/ou la distance entre eux et le capteur changent.
* Troisièmement, bien que la vitesse des gestes peut varier d'une personne à une autre, le système reste capable de reconnaître le geste.
* Finalement, le changement de la durée d'un geste d'une personne à une autre n'influence pas la reconnaissance.

Limitations et perspectives

Le système de reconnaissance de gestes proposé a quelques limitation :

1. Notre approche ne traite pas le cas des occultations. En effet, si un obstacle se présente entre la personne et le capteur, alors le tracking sera perturbé, et donc la reconnaissance sera affectée.
2. Le nombre de gestes que nous avons traités est limité. En effet, pour reconnaître un nouveau geste, il est nécessaire de le faire apprendre par ce système.
3. La reconnaissance en ligne des gestes composés via la méthode proposée n'arrive pas à détecter la phase de transition d'un geste à un autre.

Dans les travaux futurs, nous voulons enrichir notre base de données avec de nouveaux types de gestes. Nous voulons aussi combiner l'information de profondeur avec la reconnaissance de la parole pour pouvoir automatiser la détection du début et la fin d'un geste et rendre la reconnaissance plus robuste. Nous envisageons aussi de tester nos approches sur d'autres bases de données afin d'analyser le comportement du système proposé face à des nouvelles situations. Tester la méthode SVM (les

machines à supports vectoriels) et la comparer aux modèles de Markov Cachés sera ainsi intéressant.

Bibliographie

[Beleboni, 2015] Beleboni, M. G. S. (2015). A brief overview of microsoft kinect and its applications. *University of Southhampton : Southampton, UK.*

[Birdwhistell, 1963] Birdwhistell, R. L. (1963). The kinesic level in the investigation of the emotions. *Expression of the emotions in man*, pages 123–139.

[Bleiweiss and Werman, 2009] Bleiweiss, A. and Werman, M. (2009). Fusing time-of-flight depth and color for real-time segmentation and tracking. In *Dynamic 3D imaging*, pages 58–69. Springer.

[Bourke et al., 2007] Bourke, A., O'brien, J., and Lyons, G. (2007). Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. *Gait & posture*, 26(2) :194–199.

[Bretzner et al., 2002] Bretzner, L., Laptev, I., and Lindeberg, T. (2002). Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering. In *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, pages 423–428. IEEE.

[Chen et al., 2003] Chen, F.-S., Fu, C.-M., and Huang, C.-L. (2003). Hand gesture recognition using a real-time tracking method and hidden markov models. *Image and vision computing*, 21(8) :745–758.

[Chu and Cohen, 2005] Chu, C.-W. and Cohen, I. (2005). Posture and gesture recognition using 3d body shapes decomposition. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops*, pages 69–69. IEEE.

[Correa et al., 2009] Correa, M., Ruiz-del Solar, J., Verschae, R., Lee-Ferng, J., and Castillo, N. (2009). Real-time hand gesture recognition for human robot interaction. In *Robot Soccer World Cup*, pages 46–57. Springer.

[Cutler and Turk, 1998] Cutler, R. and Turk, M. (1998). View-based interpretation of real-time optical flow for gesture recognition. *fg*, 98 :416.

[Elmezain et al., 2009] Elmezain, M., Al-Hamadi, A., Appenrodt, J., and Michaelis, B. (2009). A hidden markov model-based isolated and meaningful hand gesture recognition. *International Journal of Electrical, Computer, and Systems Engineering*, 3(3) :156–163.

[Fontmarty et al., 2007] Fontmarty, M., Lerasle, F., Danès, P., and Menezes, P. (2007). Filtrage particulaire pour la capture de mouvement dédiée à l'interaction homme-robot. *Congrès francophone ORASIS*.

[Ghobadi et al., 2007] Ghobadi, S., Loepprich, O., Hartmann, K., and Loffeld, O. (2007). Hand segmentation using 2d/3d images. In *IVCNZ 2007 Conference, Hamilton, New Zealand*, volume 5.

[Gilbert et al., 2009] Gilbert, A., Illingworth, J., and Bowden, R. (2009). Fast realistic multi-action recognition using mined dense spatio-temporal features. In *2009 IEEE 12th International Conference on Computer Vision*, pages 925–931. IEEE.

[Grange, 2007] Grange, S. (2007). Medical/operating room interaction system.

[Grest et al., 2007] Grest, D., Krüger, V., and Koch, R. (2007). Single view motion tracking by depth and silhouette information. In *Scandinavian Conference on Image Analysis*, pages 719–729. Springer.

[Gu et al., 2012] Gu, Y., Do, H., Ou, Y., and Sheng, W. (2012). Human gesture recognition through a kinect sensor. In *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on*, pages 1379–1384. IEEE.

[Hall, 1973] Hall, E. T. (1973). *The silent language*. Anchor Books.

[Hiyadi et al., 2015a] Hiyadi, H., Ababsa, F., Montagne, C., Bouyakhf, E. H., and Regragui, F. (2015a). A depth-based approach for 3d dynamic gesture recognition. In *Informatics in Control, Automation and Robotics (ICINCO), 2015 12th International Conference on*, volume 2, pages 103–110. IEEE.

[Hiyadi et al., 2016a] Hiyadi, H., Ababsa, F., Montagne, C., Bouyakhf, E. H., and Regragui, F. (2016a). Adaptive dynamic time warping for recognition of natural gestures. In *International Conference on Image Processing Theory, Tools and Applications IPTA, Oulu, Finland, December 12-15*. IEEE.

[Hiyadi et al., 2016b] Hiyadi, H., Ababsa, F., Montagne, C., Bouyakhf, E. H., and Regragui, F. (2016b). Combination of hmm and dtw for 3d dynamic gesture recognition using depth only. In *Informatics in Control, Automation and Robotics 12th International Conference, ICINCO 2015 Colmar, France, July 21-23, 2015 Revised Selected Papers*, pages 229–245. Springer.

[Hiyadi et al., 2016c] Hiyadi, H., Ababsa, F., Montagne, C., Bouyakhf, E. H., and Regragui, F. (2016c). Dynamic gesture recognition for natural human system interaction. *Journal of Theoretical and Applied Information Technology*, 91(2) :374.

[Hiyadi et al., 2015b] Hiyadi, H., Ababsa, F.-E., Bouyakhf, E. H., Montagne, C., and Regragui, F. (2015b). Reconnaissance 3d des gestes pour l'interaction naturelle homme robot. In *15ème édition des journées francophones des jeunes chercheurs en vision par ordinateur (ORASIS 2015)*, pages to–appear.

[http ://pr.cs.cornell.edu/humanactivities/data.php,] http ://pr.cs.cornell.edu/humanactivities/data.php.

[https ://www.microsoft.com/en us/download/details.aspx ?id=52283,] https ://www.microsoft.com/en us/download/details.aspx ?id=52283.

[http ://www.generationrobots.com,] http ://www.generationrobots.com.

[Imagawa et al., 2000] Imagawa, I., Matsuo, H., Taniguchi, R.-i., Arita, D., Lu, S., and Igi, S. (2000). Recognition of local features for camera-based sign language recognition system. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 4, pages 849–853. IEEE.

[Jalal et al., 2015] Jalal, A., Kamal, S., and Kim, D. (2015). Depth silhouettes context : A new robust feature for human tracking and activity recognition based on advanced hidden markov model.

[Jhuang et al., 2007] Jhuang, H., Serre, T., Wolf, L., and Poggio, T. (2007). A biologically inspired system for action recognition. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. Ieee.

[Kaâniche, 2009] Kaâniche, M. (2009). *Gesture recognition from video sequences.* PhD thesis, Université Nice Sophia Antipolis.

[Kage et al., 2007] Kage, H., Seki, M., Sumi, K., Tanaka, K.-i., and Kyuma, K. (2007). Pattern recognition for video surveillance and physical security. In *SICE, 2007 Annual Conference*, pages 1823–1828. IEEE.

[Kevin et al., 2004] Kevin, N. Y. Y., Ranganath, S., and Ghosh, D. (2004). Trajectory modeling in gesture recognition using cybergloves® and magnetic trackers. In *TENCON 2004. 2004 IEEE Region 10 Conference*, pages 571–574. IEEE.

[Krout, 1935] Krout, M. H. (1935). Autistic gestures : An experimental study in symbolic movement. *Psychological Monographs*, 46(4) :i.

[Laptev et al., 2008] Laptev, I., Marszalek, M., Schmid, C., and Rozenfeld, B. (2008). Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE.

[Larsen et al., 2011] Larsen, A. B. L., Hauberg, S., and Pedersen, K. S. (2011). Unscented kalman filtering for articulated human tracking. In *Scandinavian Conference on Image Analysis*, pages 228–237. Springer.

[Laurent Bréhélin, 2010] Laurent Bréhélin, O. G. (2010). *Modèles de Markov Cachés et Apprentissage des Séquences*. Département d'informatique fondamentale et applications, LIRMM.

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

[Lu et al., 2003] Lu, S., Metaxas, D., Samaras, D., and Oliensis, J. (2003). Using multiple cues for hand tracking and model refinement. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–443. IEEE.

[Malik et al., 2002] Malik, S., McDonald, C., and Roth, G. (2002). Hand tracking for interactive pattern-based augmented reality.

[Marszalek et al., 2009] Marszalek, M., Laptev, I., and Schmid, C. (2009). Actions in context. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2929–2936. IEEE.

[Mckenna and Morrison, 2004] Mckenna, S. J. and Morrison, K. (2004). A comparison of skin history and trajectory-based representation schemes for the recognition of user-specified gestures. *Pattern Recognition*, 37(5) :999–1009.

[McNeill, 1992] McNeill, D. (1992). *Hand and mind : What gestures reveal about thought*. University of Chicago press.

[Migniot and Ababsa, 2013] Migniot, C. and Ababsa, F. (2013). 3d human tracking from depth cue in a buying behavior analysis context. In *International Conference*

on Computer Analysis of Images and Patterns, pages 482–489. Springer.

[Moreno et al., 2001] Moreno, F., Andrade-Cetto, J., and Sanfeliu, A. (2001). Localization of human faces fusing color segmentation and depth from stereo. In *Emerging Technologies and Factory Automation, 2001. Proceedings. 2001 8th IEEE International Conference on*, volume 2, pages 527–535. IEEE.

[Muñoz-Salinas et al., 2008] Muñoz-Salinas, R., Medina-Carnicer, R., Madrid-Cuevas, F. J., and Carmona-Poyato, A. (2008). Depth silhouettes for gesture recognition. *Pattern Recognition Letters*, 29(3) :319–329.

[Nickel and Stiefelhagen, 2007] Nickel, K. and Stiefelhagen, R. (2007). Visual recognition of pointing gestures for human–robot interaction. *Image and Vision Computing*, 25(12) :1875–1884.

[Niebles et al., 2008] Niebles, J. C., Wang, H., and Fei-Fei, L. (2008). Unsupervised learning of human action categories using spatial-temporal words. *International journal of computer vision*, 79(3) :299–318.

[Noury et al., 2003] Noury, N., Barralon, P., Virone, G., Boissy, P., Hamel, M., and Rumeau, P. (2003). A smart sensor based on rules and its evaluation in daily routines. In *Engineering in medicine and biology society, 2003. Proceedings of the 25th annual international conference of the IEEE*, volume 4, pages 3286–3289. IEEE.

[Oka et al., 2002] Oka, K., Sato, Y., and Koike, H. (2002). Real-time fingertip tracking and gesture recognition. *IEEE Computer graphics and Applications*, 22(6) :64–71.

[Ong and Bowden, 2004] Ong, E. and Bowden, R. (2004). Detection and segmentation of hand shapes using boosted classifiers. In *Proc. IEEE 6th International Conference on Automatic Face and Gesture Recognition*, pages 889–894.

[Ottenheimer, 2012] Ottenheimer, H. J. (2012). *The anthropology of language : An introduction to linguistic anthropology*. Cengage Learning.

[Park and Lee, 2011] Park, C.-B. and Lee, S.-W. (2011). Real-time 3d pointing gesture recognition for mobile robots with cascade hmm and particle filter. *Image and Vision Computing*, 29(1) :51–63.

[Pei, 1984] Pei, M. (1984). The story of language, plume ; rep rev edition. *ISBN-13*, pages 978–0452008700.

[Plagemann et al., 2010] Plagemann, C., Ganapathi, V., Koller, D., and Thrun, S. (2010). Real-time identification and localization of body parts from depth images. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 3108–3113. IEEE.

[Ramamoorthy et al., 2003] Ramamoorthy, A., Vaswani, N., Chaudhury, S., and Banerjee, S. (2003). Recognition of dynamic hand gestures. *Pattern Recognition*, 36(9) :2069–2081.

[Rauschert et al., 2002] Rauschert, I., Agrawal, P., Sharma, R., Fuhrmann, S., Brewer, I., and MacEachren, A. (2002). Designing a human-centered, multimodal gis interface to support emergency management. In *Proceedings of the 10th ACM international symposium on Advances in geographic information systems*, pages 119–124. ACM.

[Rautaray and Agrawal, 2011] Rautaray, S. S. and Agrawal, A. (2011). A real time hand tracking system for interactive applications. *International journal of computer Applications*, 18(6) :28–33.

[Roh et al., 2006] Roh, M.-C., Shin, H.-K., Lee, S.-W., and Lee, S.-W. (2006). Volume motion template for view-invariant gesture recognition. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 2, pages 1229–1232. IEEE.

[Schlömer et al., 2008] Schlömer, T., Poppinga, B., Henze, N., and Boll, S. (2008). Gesture recognition with a wii controller. In *Proceedings of the 2nd international conference on Tangible and embedded interaction*, pages 11–14. ACM.

[Schmidt et al., 2008] Schmidt, A., Gellersen, H., van den Hoven, E., Mazalek, A., Holleis, P., and Villar, N. (2008). *TEI'08 : Proceedings of the 2nd international conference on Tangible and embedded interaction*. ACM.

[Schuldt et al., 2004] Schuldt, C., Laptev, I., and Caputo, B. (2004). Recognizing human actions : a local svm approach. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 32–36. IEEE.

[Sigal et al., 2004] Sigal, L., Sclaroff, S., and Athitsos, V. (2004). Skin color-based video segmentation under time-varying illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7) :862–877.

[Stiefelhagen et al., 2004] Stiefelhagen, R., Fugen, C., Gieselmann, R., Holzapfel, H., Nickel, K., and Waibel, A. (2004). Natural human-robot interaction using speech, head pose and gestures. In *Intelligent Robots and Systems, 2004.(IROS 2004).*

Proceedings. 2004 IEEE/RSJ International Conference on, volume 3, pages 2422–2427. IEEE.

[Swee et al., 2007] Swee, T. T., Salleh, S.-H., Ariff, A., Ting, C.-M., Seng, S. K., and Huat, L. S. (2007). Malay sign language gesture recognition system. In *Intelligent and Advanced Systems, 2007. ICIAS 2007. International Conference on*, pages 982–985. IEEE.

[Takahashi, 2009] Takahashi, D. (2009). Microsoft games exec details how project natal was born. *VentureBeat. Retrieved June*, 6.

[Terrillon and Akamatsu, 1999] Terrillon, J.-C. and Akamatsu, S. (1999). Comparative performance of different chrominance spaces for color segmentation and detection of human faces in complex scene images. In *Vision Interface*, volume 99, page 1821. Citeseer.

[Wang et al., 2012] Wang, X., Xia, M., Cai, H., Gao, Y., and Cattani, C. (2012). Hidden-markov-models-based dynamic hand gesture recognition. *Mathematical Problems in Engineering*, 2012.

[Webel et al., 2008] Webel, S., Keil, J., and Zoellner, M. (2008). Multi-touch gestural interaction in x3d using hidden markov models. In *Proceedings of the 2008 ACM symposium on Virtual reality software and technology*, pages 263–264. ACM.

[Wu et al., 2000] Wu, P., Manjunath, B., Newsam, S., and Shin, H. (2000). A texture descriptor for browsing and similarity retrieval. *Signal Processing : Image Communication*, 16(1) :33–43.

[Wu and Huang, 2002] Wu, Y. and Huang, T. S. (2002). Nonstationary color tracking for vision-based human-computer interaction. *IEEE transactions on neural networks*, 13(4) :948–960.

[Xia et al., 2011] Xia, L., Chen, C.-C., and Aggarwal, J. K. (2011). Human detection using depth information by kinect. In *CVPR 2011 WORKSHOPS*, pages 15–22. IEEE.

[Xu et al., 2011] Xu, D., Chen, Y.-L., Wu, X., Ou, Y., and Xu, Y. (2011). Integrated approach of skin-color detection and depth information for hand and face localization. In *Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on*, pages 952–956. IEEE.

[Yao and Cooperstock, 2002] Yao, J. and Cooperstock, J. R. (2002). Arm ges-

ture detection in a classroom environment. In *Applications of Computer Vision, 2002.(WACV 2002). Proceedings. Sixth IEEE Workshop on*, pages 153–157. IEEE.

[Yao et al., 2004] Yao, Y., Zhu, M., Jiang, Y., and Lu, G. (2004). A bare hand controlled ar map navigation system. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 3, pages 2635–2639. IEEE.

[Yeffet and Wolf, 2009] Yeffet, L. and Wolf, L. (2009). Local trinary patterns for human action recognition. In *2009 IEEE 12th International Conference on Computer Vision*, pages 492–497. IEEE.

[Yilmaz et al., 2004] Yilmaz, A., Li, X., and Shah, M. (2004). Contour-based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Transactions on pattern analysis and machine intelligence*, 26(11) :1531–1536.

[Yuan et al., 2008] Yuan, M., Farbiz, F., Manders, C. M., and Tang, K. Y. (2008). Robust hand tracking using a simple color classification technique. In *Proceedings of The 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, page 6. ACM.

[Zhang et al., 2001] Zhang, Z., Wu, Y., Shan, Y., and Shafer, S. (2001). Visual panel : virtual mouse, keyboard and 3d controller with an ordinary piece of paper. In *Proceedings of the 2001 workshop on Perceptive user interfaces*, pages 1–8. ACM.

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Reconnaissance 3D de gestes pour l'interaction homme-système

Mots clefs : Interaction homme-système, Reconnaissance des gestes dynamiques, Descripteur 3D de gestes, Apprentissage automatique, Classification de gestes, Information de profondeur.

Résumé : Le but des applications visées par l'interaction homme-système est de parvenir à une interaction naturelle qui simule l'interaction homme-homme. Comme dans la communication homme-homme, les gestes sont aussi très utilisés dans la communication homme-système. Cette thèse porte sur la reconnaissance de gestes pour l'interaction naturelle homme-système basée sur les gestes. L'objectif des travaux menés durant cette thèse est de proposer des approches de reconnaissance de différents types de geste dynamiques : gestes simples et gestes composés. Tous dabord, nous avons proposé un nouveau descripteur 3D de gestes calculé par les angles des articulations du corps humain à partir d'un flux de profondeur fourni par le capteur Kinect. Ensuite, nous avons proposé deux approches pour la reconnaissance de gestes dynamiques : a) une approche de reconnaissance des gestes simples, b) une approche de reconnaissance des gestes composés. La première approche est basée sur les Modèles de Markov Cachés. Un modèle MMC pour chaque geste a été réalisé. La variation des angles entre les articulations est utilisée comme entrée des Modèles de Markov Cachés. Cette méthode a été combinée avec la méthode de la Déformation Temporelle Dynamique (Dynamic Time Warping) pour éliminer les mauvaises classifications. La deuxième approche traite le cas des gestes composés et successifs dans une même séquence. Cette approche combine la méthode de la Déformation Temporelle Dynamique avec une fenêtre glissante adaptative d'où le nom de l'approche: Adaptive Dynamic Time Warping. Deux versions de cette approche ont été proposées : version Forward et version Backward pour la reconnaissance en ligne et hors ligne.

Title : 3D gesture recognition for human-system interaction

Keywords : Human-system interaction, Dynamic gesture recognition, 3D gesture descriptor, Maching learning, Gesture classification, Depth data.

Abstract : The goal of Human System Interaction (HSI) research is to increase the performance of human system interaction in order to make it similar to human-human interaction. As for communication between humans, gestural communication is also widely used in human system interaction. This thesis is about gesture recognition for natural human system interaction based on gestures. The objectif of works conducted in this thesis is to propose recognition approches for different kind of dynamic gesture: simple gesture and composed gesture. Fist of all, we proposed a novel 3D gesture descriptor computed by human body joints angles provided by Kinect sensor. Second, we proposed two dynmaic gestures recognition approaches: a) simple gestures recognition approach, b) composed gestures recognition approach. The first approach is based on Hidden Markov Models (HMM). One HMM was created for each gesture. The joints angles variations have been used as input for HMMs. Then, this method has been combined with the Dynamic Time Warping algorithm in order to eliminate bad classification. The second approach treats the case of composed and successive gestures in the same sequence. This approach combines Dynamic Time Warping method with an adaptative window, hence the name: Adaptive Dynamic Time Warping. Two versions have been proposed: Forward and Backward version for online and offline recognition.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France