LA RECONNAISSANCE DU LANGAGE DE SIGNES PAR LES FONCTIONS DE TAILLE

par

Handouyahia Mohamed

mémoire présenté au Département de mathématiques et d'informatique
en vue de l'obtention du grade de maître ès sciences (M.Sc.)

FACULTÉ DES SCIENCES
UNIVERSITÉ DE SHERBROOKE

Sherbrooke, Québec, Canada, décembre 1998

Le 20 Novembre 1998., le jury suivant a accepté ce mémoire dans sa version finale.
date

Président-rapporteur: M. Béchir Ayeb
Département de mathématiques et d'informatique

Membre: M. Shengrui Wang
Département de mathématiques et d'informatique

Membre: M. Djemel Ziou
Département de mathématiques et d'informatique

Membre: M. Tomasz Kaczynski
Département de mathématiques et d'informatique

Sommaire

Le principal objectif de mon projet de maîtrise est la conception d'un système de reconnaissance adapté au canal gestuel afin de pouvoir intégrer de nouvelles possibilités d'interaction au sein d'interfaces Personne-Machine. Une attention particulière a été portée au problème de la reconnaissance de l'alphabet du langage des signes italien (LSI).

Après avoir étudié les différents paramètres du langage des signes, et les différentes techniques utilisées précédemment dans ce domaine, nous avons choisi l'approche visuelle pour bénéficier des techniques de traitement d'images afin d'obtenir une meilleure représentation des gestures et pour ainsi mieux distinguer les différentes postures du langage des signes. Nous avons ainsi défini un ensemble de caractéristiques pertinentes et effectué l'expérimentation de plusieurs techniques d'extraction de caractéristiques.

Le problème que nous avons soulevé est celui de trouver une caractéristique plus générale et plus adaptée au canal gestuel. Au niveau de la représentation des signes, nous avons opté pour l'utilisation d'un outil topologique appelé «*Fonctions de taille*», lui-même basé sur un autre paramètre essentiel appelé «*Fonctions de mesure*». Nous avons opté pour l'utilisation des moments d'inertie dans le but de trouver une fonction de mesure spécifique à la représentation de l'alphabet du langage des signes. Notre schéma de reconnaissance est construit autour d'un classificateur neuronal.

Remerciements

Je voudrais remercier mon directeur de recherche, le professeur Djemel Ziou, qui m'a permis de poursuivre mes études supérieures de deuxième cycle et qui m'a accordé son support tout au long de mon projet, me permettant ainsi de contribuer à la recherche dans le domaine de la vision par ordinateur. Je remercie également tous les autres collègues et les professeurs du Département de mathématiques et d'informatique qui m'ont soutenu de près ou de loin durant mes travaux de recherche, tout particulièrement le professeur Shengrui Wang pour sa contribution au sein de notre équipe de recherche. Merci au professeur Patrizio Frosini de l'Université de Bologne en Italie pour m'avoir fourni les images des signes et avoir répondu à mes questions.

Je tiens aussi à remercier mon amie Maryse Bérubé qui m'a aidé tout au long de la rédaction de mon mémoire.

Je suis aussi reconnaissant envers mes collègues et mes amis du Département, tout particulièrement Madjid Allili, qui ont plusieurs fois accepté de répondre à mes questions.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

Introduction

Cette étude s'inscrit dans le domaine de la communication Personne-Machine et plus spécifiquement dans celui de la communication gestuelle. Le principal objectif de la présente étude est la conception d'un système de reconnaissance et de compréhension adapté au canal gestuel afin de pouvoir intégrer de nouvelles possibilités d'interaction au sein d'interfaces Personne-Machine. Une attention particulière a été portée au problème de la reconnaissance et de la compréhension des postures de l'alphabet du langage des signes.

Nous utilisons constamment nos mains pour interagir avec les objets : pour les prendre, les bouger ou transformer leurs formes. Fondamentalement, déjà nous gesticulons fréquemment pour nous exprimer comme pour dire «stop», «là-bas», «non» et plus. Les gestures sont donc une forme naturelle d'interaction et de communication. Les chercheurs ont compris l'importance des interfaces «Personne-Machine». Selon l'angle sous lequel nous regardons le problème, deux questions peuvent se poser :

- du point de vue informatique, que peuvent être les apports des interfaces gestuelles à la communication Personne-Machine?
- du point de vue humain, que peut apporter une interface gestuelle aux personnes qui utilisent un ordinateur?

La première question pose le problème de l'intégration des interfaces gestuelles au sein d'applications informatiques. L'apparition de nouveaux capteurs de gestes (gant numé-

rique, caméra, capteur de position, oculomètre ...) a favorisé l'émergence de nouveaux thèmes de recherche visant à intégrer la modalité gestuelle au sein d'applications informatiques. Ce canal d'information apporte alors de nouvelles possibilités d'interaction.

La seconde question pose le problème des applications possibles pour les utilisateurs d'ordinateurs en général et pour les personnes pour lesquelles les interfaces gestuelles sont primordiales.

Nous pouvons distinguer deux types d'applications d'interaction Personne-Machine :

- gestures comme un langage symbolique ou linguistique comme par exemple, le logiciel d'interprétation des signes et la traduction vers un langage parlé ou écrit;

- gestures pour fournir un contrôle des événements dans une application informatique comme par exemple, les commandes de contrôle de la télévision ou de robots.

Nous souhaitons, par l'intermédiaire de ce mémoire, contribuer à l'évolution des connaissances dans le domaine de la reconnaissance et de la compréhension des gestes afin de permettre une avancée dans le domaine de la communication Personne-Machine, et aussi dans le domaine plus spécifique des didacticiels dédiés à la langue des signes.

Le chapitre 1 présente les différents paramètres du langage des signes. Nous allons étudier la physiologie de la main et les différents éléments qui distinguent les différents gestes statiques. Nous avons choisi d'étudier les gestes statiques de la langue des signes italiens dans le but de comparer nos résultats à ceux des autres expériences réalisées sur la même base de données qui est composée uniquement de postures.

Le chapitre 2 a pour objectif de comparer plusieurs systèmes de reconnaissance du langage des signes existants applicables principalement aux gestes statiques afin de choisir un schéma de représentation et un système de reconnaissance des mieux adaptés.

Dans le chapitre 3, nous exposons différentes représentations des gestes statiques. De plus, nous évaluons ces différentes représentations selon plusieurs critères particulièrement l'invariance à la translation, à l'échelle et à la rotation.

Le chapitre 4 présente un nouvel outil mathématique de représentation des formes, appelé fonctions de taille. Nous décrivons ses propriétés et nous discutons son adaptation à l'analyse du langage des signes. Nous nous attachons à mettre en évidence et à justifier les choix des algorithmes conçus, réalisés et validés sur un système de reconnaissance qui sera défini dans les chapitres suivants.

Le chapitre 5 a pour but de proposer et de décrire deux fonctions de taille choisies pour représenter les différents gestes de l'alphabet du langage des signes. Ces fonctions de taille seront ensuite intégrées dans un nouveau modèle de représentation.

Finalement, le chapitre 6 présente l'architecture retenue pour l'élaboration du système de reconnaissance ainsi que les différents modules de traitement. Il s'agit d'un classificateur neuronal multi-couche. Ce système doit permettre de reconnaître toutes les gestures statiques de la langue des signes. Après une présentation des différents modules qui composent le système, nous dévoilons les résultats de plusieurs expériences afin d'évaluer le fonctionnement du prototype qui a été developpé.

La dernière partie présente la conclusion et les perspectives relatives au travail réalisé dans le cadre de cette étude.

CHAPITRE 1

Paramètres du langage des signes

Avant de présenter la structuration des signes suivant les cinq paramètres couramment utilisés (configuration, mouvement, orientation, emplacement et mimique faciale), nous devons d'abord introduire les données physiologiques qui différencient les langues gestuelles des langues orales. Quelques études de type phonologique menées sur les langues des signes sont ensuite brièvement exposées.

1.1 Langues orales VS langues gestuelles

Une étude réalisée sur des gestes statiques de l'ASL (American Sign Language) et des mots de l'américain parlé [28, 7] ainsi qu'une analyse des différentes postures de l'ISL (Italian Sign Language) ont montrées qu'il existe trois données physiologiques qui différencient radicalement les langues des signes des langues orales :

- pour un même concept, le temps d'émission moyen d'un geste statique est approximativement deux fois plus long que celui d'un mot et à contenu et quantité d'information égals, le temps d'émission d'un discours en langue des signes et en américain parlé est approximativement le même.

- Le système auditif humain est adapté à la discrimination temporelle tandis que le système visuel est adapté à la discrimination spatiale.

- Le signal capté par l'oeil correspond directement au mouvement des articulateurs (les mains, les bras...), tandis que l'oreille est sensible que pour l'effet sonore produit par le mouvement des articulateurs (les cordes vocales).

Les informations contenues dans l'émission des signes statiques et dynamiques sont portées par des paramètres de type phonologique qui sont la configuration, le mouvement de la main, l'emplacement et l'orientation de la main par rapport au corps et à la mimique faciale [36]. Parfois les deux mains interviennent et parfois une seule est utilisée. Ces paramètres sont détaillés dans le paragraphe suivant.

Plusieures informations peuvent être émises simultanément selon la variation de l'un ou l'autre de ces paramètres. Nous pouvons imaginer toute l'économie de temps réalisée si ces cinq paramètres sont émis simultanément et si chaque paramètre est exploité au niveau syntaxique.

Notons que d'autres parties du corps participent lors de l'émission d'un message en langage des signes français (LFS) [31], en particulier les épaules et le buste. Un geste en langue des signes représente un ensemble de mouvements effectués par différentes parties du corps simultanément. Mais seuls les paramètres cités précédemment ont été considérés.

1.2 Définition des paramètres

Sans entrer dans le détail de la définition des systèmes de communication gestuelle ou du langage des signes gestuels, voyons les éléments (paramètres) qui interviennent dans la production des signes gestuels [7, 31].

Nous nous sommes intéressés aux mouvements des parties du corps (mains et bras) lors de l'émission du message gestuel, indépendamment de leur rôle linguistique. De ce point de vue, les muscles qui permettent de faire bouger les doigts sont différents de ceux qui déplacent le bras. De plus, pour un même déplacement du bras, il est possible, à l'aide de muscles spécifiques, d'effectuer une rotation ou une flexion du poignet. Ainsi, contrairement à ce qui est souvent considéré dans les études phonologiques, nous avons considéré plusieurs types de mouvements tels que les mouvements des doigts, de la main et du bras. D'où les définitions suivantes, adaptées à l'étude des gestes de la main :

- la configuration de la main : La configuration représente la forme et le mouvement des doigts de la main. Les doigts peuvent être immobiles ou non, donc la configuration peut être statique ou dynamique.

- L'orientation de la main : Elle se définie par l'orientation de l'axe de la main, celle de la paume, et l'état de l'articulation du poignet (repos, rotation ou flexion). Ces valeurs peuvent varier durant l'exécution du signe. Ce paramètre peut être statique ou dynamique. La main peut prendre 5 orientations possibles : La paume peut être tournée vers le signeur, le capteur, le haut, le sol où encore de profil (une seule position de profil est possible). Le signe pour dire «porte» consiste à mimer l'ouverture de la porte avec les mains. Les mains sont placées à la verticale, la paume tournée vers le signeur avec l'extrémité des doigts en contact, nous ouvrons ensuite les mains vers l'extérieur jusqu'à la position de profil.

- Les mouvements : Ils représentent la trajectoire du déplacement de l'extrémité de l'avant-bras (côté poignet). Lorsque le bras est immobile, nous parlerons de paramètre mouvement «statique». Nous pouvons avoir jusqu'à 30 mouvements que l'on regroupe en plusieures catégories : Les mouvements verticaux, horizontaux, latéraux, circulaires, d'ouverture et de fermeture ou combinés, etc.

- La position dans l'espace : L'emplacement représente la zone dans laquelle le signe est effectué par rapport au signeur. À chaque fois que le bras bouge, l'emplacement varie. Il y a une corrélation directe entre le paramètre de mouvement et la position. Ce paramètre peut être statique ou dynamique.

- L'expression faciale : Le dernier élément considéré par ISL (Italian Sign Language) comme élément pouvant intervenir dans la production d'un signe gestuel est l'expression faciale. En langage des signes, le sens des signes peut différer si l'on varie l'expression du visage. L'expressivité faciale n'intervient pas en principe au niveau de l'organisation du code dans les langages gestuels formels.

Nous chercherons dans la section suivante, à énumérer toutes les informations physiques et géométriques pouvant distinguer les différents signes et les différents paramètres cités précédemment. Dans ce qui suit, nous présenterons la physiologie de la main ainsi que le modèle perceptuel des postures afin d'obtenir les paramètres essentiels dans l'exécution de celles-ci. Puisqu'il s'agit seulement de postures, nous négligerons les paramètres de la position de la main dans l'espace et de l'expression faciale.

1.3 Physiologie et modélisation de la main

Pour mieux comprendre le langage des signes, il est nécessaire et utile d'étudier les informations physiques de la main et la manière dont la main peut s'opérer et de quelle façon ces mouvements peuvent être complexes. Cela nécessite un examen des structures internes de la main.

Normalement, la main se constitue de 15 jointures (incluant le poignet) et quelques 20 (ou plus) segments engendrés, constituant la paume et les doigts.

Les jointures sont appelées selon leurs différentes locations dans la main. Les jointures *MetaCorpoPhalageal* (joignant les doigts et la paume) et les jointures *InterPhalangeal* (joignant les segments des doigts) (voir fig. 1.1).

Une simple inspection nous confirme que les neufs jointures *InterPhalangeal* qui sont divisées en deux catégories, l'une pour les jointures InterPhalangeal distals (DIP) et l'autre pour les jointures InterPhalangeal Proximals (PIP), peuvent avoir seulement un degré de liberté et la possibilité de flexion-extension.

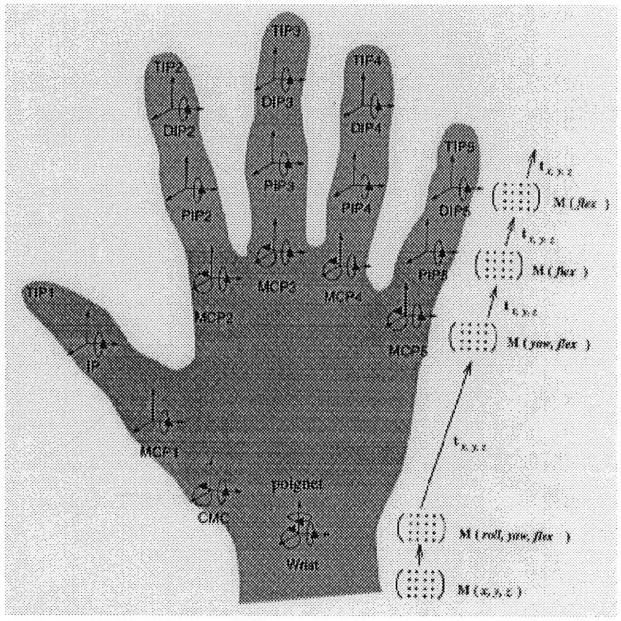


Figure 1.1: *Une vue physique de la main.*

La situation est plus compliquée pour le cas des jointures *MetaCarpophalageals*, toutes les cinq ont été décrites dans la littérature comme des jointures à deux degrés de liberté, avec la possibilité d'abduction-adduction en plus de flexion-extension comme dans les jointures InterPhalangeal (repos, rotation, flexion). C'est le cas des quatre doigts excepté le pouce ayant une possibilité d'abduction/adduction très restreinte.

Il y a au total 23 degrés de liberté dans les mouvements du poignet et des doigts (voir

fig. 1.2) dont :

- les jointures des doigts avec la paume (MCP) : 9 degrés de liberté au total;
- Les jointures des segments des doigts (IP = DIP + PIP) : 9 degrés de liberté au total;
- Les jointures du poignet : 3 degrés de liberté au total;
- Les jointures (CarpoMetaCarpal) du pouce comme une partie de la paume : 2 degrés de liberté au total.

La forme mathématique de la main est définie par un ensemble de matrices de transformations qui relie les systèmes de coordonnées locales des segments. Plus précisément, la valeur de flexion/extension et abduction/adduction des différentes jointures qui donnent aux matrices de transformation, les valeurs d'angle de rotation.

La main a 15 jointures, chacune d'elle est capable de flexion/extension, 5 de ces jointures sont capables aussi d'abduction/adduction. Les symboles θ_1, θ_2 et θ_3 représentent les valeurs angulaires de flexion/extension des 2 jointures de chaque doigts (15 jointures au total). Le symbole ρ représente l'abduction/adduction d'une autre jointure de chaque doigt (5 jointures au total). Donc, une forme particulière de la main est déterminée par 20 paramètres. Les symboles x, y et z représentent la position et finalement les symboles α, β, γ représentent l'orientation de la main (voir fig. 1.2).

La main peut bouger du bas vers le haut sur l'axe vertical (l'axe des y), de gauche à droite sur l'axe horizontal (l'axe des x) et de l'avant vers l'arrière sur l'axe du capteur (l'axe des z). Elle peut aussi faire des rotations autour de ces axes. Plusieurs signes se diffèrent selon ces facteurs.

Pour chaque doigt excepté pour le pouce, deux valeurs angulaires sont nécessaires à mesurer (fig. 1.2), il s'agit de la flexion de l'articulation métacarpophalangienne θ_1 et de la flexion de l'articulation interphalangienne θ_2. L'angle entre la dernière phalange et la deuxième θ_3 n'est pas à mesurer. De toute manière, ce n'est pas indispensable puisque des études ont montré que la valeur de cette articulation est entièrement corrélée aux valeurs des autres articulations.

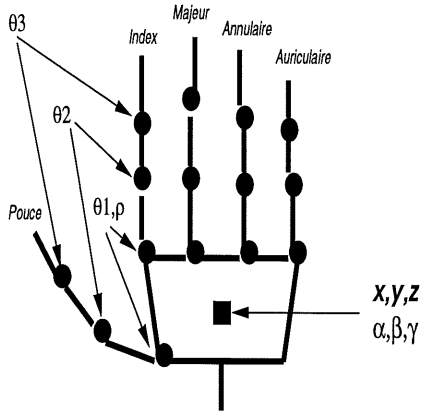


Figure 1.2: *Les degrés de liberté de la main.*

En ce qui concerne le pouce, seuls θ_2 et θ_3 sont nécessaires, ce qui n'est pas suffisant pour calculer θ_1 car les axes de flexion et de rotation du pouce ne sont pas triviaux, du fait que le pouce fait partie de la paume de la main et ne soit pas parallèle aux autres doigts.

Les valeurs extrêmes pour chaque flexion, sont décrites dans le tableau 1.1.

Tableau 1.1: *Les angles de flexion.*

flexion	pouce	flexion	index	majeur	annulaire	auriculaire
θ_2	[0,45]	θ_1	[0,90]	[0,90]	[0,90]	[0,90]
θ_3	[0,90]	θ_2	[0,90]	[0,90]	[0,90]	[0,90]

1.4 Modèle perceptuel des postures

À travers l'étude effectuée dans les sections précédentes, nous pouvons distinguer deux catégories de gestures : Les gestures statiques et les gestures dynamiques.

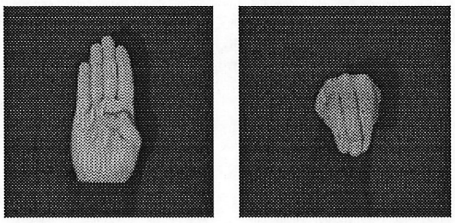
Dans le cas des gestures statiques, la configuration particulière de la main est représentée par une image fixe de la main. Les gestures dynamiques sont des gestures en mouvements représentées par une séquence d'images. Il est nécessaire de faire une différence entre «Reconnaissance de postures» et la «Reconnaissance de gestures». La reconnaissance de postures est la reconnaissance des positions statiques des parties du corps comme par exemple, un signe représentant une lettre de l'alphabet du langage des signes. La reconnaissance de gestures, contrairement aux postures, doit inclure les changements dynamiques dans la posture. La reconnaissance de postures est un sous-ensemble de la reconnaissance de gestures.

La reconnaissance du langage des signes est un exemple d'application spécifique de la reconnaissance de gestures. Les recherches sont orientées vers cette application puisque le langage est déjà prédéfini et connu, et la taille du vocabulaire est largement suffisante pour nous permettre de faire la validation. Nous nous limiterons dans ce travail, à la reconnaissance des poses statiques. Perceptuellement, nous pouvons distinguer les différentes postures par le nombre et la position des doigts, l'orientation des doigts et l'orientation de la paume. Avant de présenter ces différents paramètres, nous considérons que l'axe optique de la caméra traverse la main.

1.4.1 Le nombre de doigts et leurs positions

D'après le modèle physiologique décrit précédemment, la main peut être considérée comme un objet qui a un centre de gravité. Les doigts peuvent prendre deux positions

différentes dépendamment de l'extension ou de la flexion des jointures qui séparent les 4 doigts (l'index, le majeur, l'annulaire et l'auriculaire) de la paume (voir fig. 1.2). Si les jointures sont en extension (fig. 1.3(a)), la position des doigts est au-dessus du centre de gravité de la main. Dans le cas des jointures en flexion (fig. 1.3(b)) la position des doigts est au-dessous du centre de gravité.



(a) *(b)*

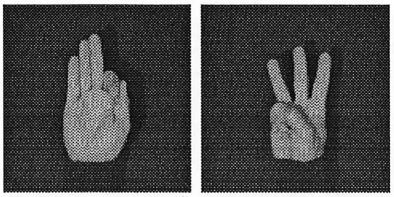
Figure 1.3: *La position des doigts : (a) les jointures en extension pour le signe «B» et (b) les jointures en flexion pour le signe «M».*

Formellement, dans les chapitres suivants, l'axe qui sépare les doigts de la paume sera considéré comme étant l'axe horizontal qui passe par le centre de gravité de la main.

Pour chaque forme possible d'une position de la main, la position des 5 doigts est une information importante. Nous pouvons constater que dans le cas de la (fig. 1.3(a)), les quatre bouts des doigts sont en haut contrairement à ceux de la figure 1.3(b). Ceci nous permet de distinguer un bon nombre de postures de la main par rapport à la position des bouts des doigts, il suffit seulement de détecter le nombre de doigts au-dessous et au-dessus de l'axe horizontal qui passe par le centre de gravité de la main.

Dans la figure 1.4, nous pouvons constater que les signes «F» et «W» ont le même nombre de bouts de doigts au-dessus de l'axe horizontal qui passe par le centre de gravité de la main, mais ils sont interprétés différemment. Le signe «F» (fig. 1.4(a)) a 3 bouts de doigts en haut, il s'agit des bouts de l'auriculaire, l'annulaire et le majeur. Par contre, pour le signe «W» (fig. 1.4(b)), il s'agit des bouts de l'annulaire, du majeur et de l'index. La

détection du nombre de doigts n'est pas suffisante pour distinguer toutes les postures, comme nous pouvons le voir dans la figure 1.4.



(a) *(b)*

Figure 1.4: *Le même nombre de doigts avec des interprétation diffrentes: (a) le signe «F» avec trois doigts collés et orientés vers le haut et (b) le signe «W» avec trois doigts décollés et orientés vers le haut.*

Ces postures ont les mêmes positions des doigts relativement à la paume de la main dont les jointures qui séparent les doigts de la paume sont en extension, mais elles sont différentes. Il faut donc recenser la position de chaque doigt relativement aux autres doigts voisins. Nous pouvons penser au nombre et à l'ordre d'apparition des vallées et des bouts des doigts au-dessus et au-dessous de l'axe horizontal qui passe par le centre de gravité de la main. Ceci permet de distinguer entre les deux gestures de la figure 1.4 car, la gesture de la figure 1.4(a) a trois bouts de doigts et ne présente aucune vallée contrairement à la gesture de la figure 1.4(b) qui a trois vallées et trois bouts de doigts.

1.4.2 L'orientation des doigts

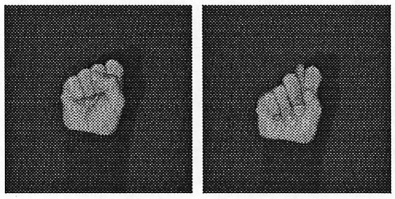
Dans un premier temps, nous pouvons classer d'une façon intuitive les postures du langage des signes en deux grandes classes :

- les lettres : dans le cas des lettres, les doigts sont orientés d'une façon verticale ou horizontale, vers le haut (fig. 1.5(a)), vers le bas (fig. 1.5(c)) ou à droite du plan de l'image (fig. 1.5(b)).

(a) *(b)* *(c)*

Figure 1.5: *L'orientation des doigts : (a) les bouts des doigts sont orientés verticalement pour le signe «B», (b) à droite du plan image pour le signe «H» et (c) vers le bas pour le signe «N».*

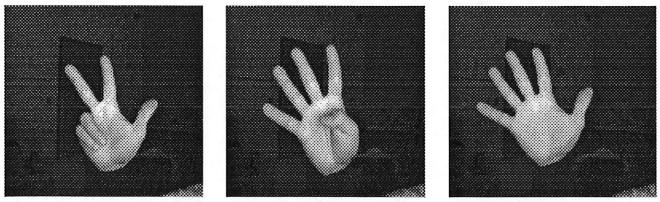
Au niveau de la classe des lettres, nous devons absolument ajouter l'information sur les flexions des doigts. En l'absence de bouts de doigts, nous pouvons confondre facilement les signes «S» et «T» qui sont presque équivalents sur tous les plans plus particulièrement en présence du bruit (voir fig. 1.6).



(a) *(b)*

Figure 1.6: *Les doigts des deux signes «S» et «T» sont fléchis vers la paume sauf pour le pouce : (a) le pouce en flexion dessus l'index et le majeur pour le signe «S» et (b) le pouce en extension entre l'index et le majeur pour le signe «T».*

- Les chiffres : dans le cas des chiffres, les doigts sont orientés vers la gauche du plan de l'image (approximativement 30 degrés). Chaque chiffre se distingue des autres par le nombre de doigts étendus au-dessus de la paume (voir fig. 1.7).



(a) (b) (c)

Figure 1.7: *L'orientation inclinée des postures correspondantes aux différents chiffres : (a) le chiffre «3», (b) le chiffre «4» et (c) le chiffre «5».*

1.4.3 L'orientation de la paume

La deuxième caractéristique perceptuelle des gestures statiques consiste à détecter l'orientation de la paume.

(a) *(b)* *(c)*

Figure 1.8: *Les différentes orientations de la paume : (a) de face pour le signe «F», (b) de profil pour le signe «D» et (c) vers le bas pour le signe «M».*

Les signes peuvent être distingués aussi par l'orientation de la paume, soit face à la caméra, vers le bas ou encore vers la gauche du plan de l'image (fig. 1.8).

Nous pouvons distinguer les signes comme par exemple, le signe «F» dont la paume est orientée face à la caméra, le signe «D» dont la paume est orientée vers la gauche et le signe «M» dont la paume est orientée vers le bas et cachée par les trois doigts.

Enfin, l'espace arrière-plan pose un problème majeur au niveau de la segmentation. Il existe à ce jour plusieures techniques pour remédier à ce problème, peu d'entre elles donnent des résultats efficaces [15]. Pour éviter les problèmes de segmentation, nous

avons utilisé des images de signes prises dans un arrière-plan noir.

1.5 Conclusion

Nous avons présenté dans ce chapitre, les paramètres essentiels dans l'émission et l'exécution d'un signe, plus particulièrement, les paramètres liés aux postures dont nous pouvons citer, la position, le nombre ainsi que l'orientation des doigts et l'orientation de la paume. Nous pouvons conclure que la grande majorité des postures peuvent être distinguées par la position, le nombre et l'orientation des doigts relativement à un axe de référence associé à chaque signe que nous allons tenter de trouver dans le chapitre 5. Dans les chapitres suivants, nous analyserons les techniques déjà utilisées pour la reconnaissance des signes et leur comportement face au respect de ces paramètres étudiés dans ce chapitre. Ainsi que dans les chapitres 3 et 5, où nous devons choisir et admettre un type de caractéristique en fonction des paramètres décrits précédemment.

Nous concluons que dans la reconnaissance de l'alphabet du langage des signes, nous devons espérer que pour admettre deux formes de signes comme similaires, elles doivent être aussi semblables relativement aux paramètres cités dans ce chapitre. Dans le chapitre 3, nous exposerons plus en détail l'influence de ces paramètres dans le choix des caractéristiques.

CHAPITRE 2

État de l'art

La reconnaissance des formes fait l'objet de recherches assidues depuis plus de trente ans [39, 10, 23]. Les diverses applications de la reconnaissance des formes ainsi que les nombreux problèmes posés par ces applications sont d'une grande motivation pour les chercheurs. Les études réalisées dans le domaine de la reconnaissance de gestes statiques [32, 42, 47, 48, 15, 12, 24, 19] et dynamiques [32, 41, 27, 40, 33, 24, 21, 37, 43, 19] sont récentes. Si depuis deux ou trois ans, beaucoup d'études ont été réalisées, il s'agit pour la plupart d'études de faisabilité permettant de tester, pour un système de reconnaissance donné et utilisé dans un domaine particulier. C'est sans doute l'une des raisons pour lesquelles il ne s'est pas dégagé à ce jour un consensus en ce qui concerne le type de méthode utilisée, de même qu'il n'existe pas de corpus universel permettant de comparer les performances des différentes méthodes.

Les méthodes existantes sont variées, de même que le vocabulaire et le langage des signes étudié. Il peut s'agir de postures (gestes statiques) de la main ou de gestures (gestes dynamiques). De plus, certaines études s'attachent à la reconnaissance de gestes isolés, tandis que d'autres s'intéressent à des séquences de gestes enchaînés. Dans ce dernier cas, différentes méthodes sont utilisées pour segmenter les gestes.

Après avoir présenté brièvement la terminologie relative au domaine de la reconnaissance de gestes, nous répertorions ensuite les méthodes les plus courantes dans le domaine de reconnaissance de gestes pour les représenter et les classifier. Nous décrivons ensuite ces méthodes dans un tableau comparatif portant sur la taille du vocabulaire et les performances des systèmes. En conclusion, nous indiquons quels sont les choix qui nous semblent les meilleurs en fonction du type d'application choisie qui est celle de l'alphabet du langage des signes italien.

2.1 Terminologie

La terminologie employée dans notre contexte est issue des domaines de la reconnaissance des formes en général et des gestes en particulier.

2.1.1 Reconnaissance des formes

La reconnaissance des formes peut se définir comme l'ensemble des techniques informatiques de représentation et de décision permettant aux machines d'interpréter des événements issus de capteurs physiques [5, 4, 8, 17, 22, 23]. L'interprétation consiste à catégoriser le phénomène perçu : il s'agit de passer d'une représentation numérique, c'est-à-dire continue, à une représentation symbolique, ou encore discrète. Il faut construire des programmes qui, à partir de données topologiques à valeur dans un espace de représentation, permettent de décider automatiquement à quelle classe, dans un espace d'interprétation, appartiennent les données. Les systèmes de reconnaissance sont composés de deux sous-systèmes dédiés respectivement aux processus de représentation et de décision (voir fig. 2.1) :

- *le processus de représentation* transforme les données brutes issues des capteurs en une représentation particulière de la forme. En général, la représentation interne de la forme est constituée d'un vecteur de paramètres extraits des données

brutes. Notons que le mot paramètre possède ici un sens différent de celui employé au chapitre précédent concernant la langue des signes (configuration, mouvement, orientation, emplacement et expression faciale). Afin d'éviter toute confusion par la suite, le terme paramètres de représentation sera utilisé quand il sera question d'algorithmes de reconnaissance des formes.

- *Le processus de décision* prend en entrée la sortie du processus de représentation et produit en sortie, si c'est possible, une classification de la forme.

Les systèmes de reconnaissance travaillent en général sur un vocabulaire bien défini. Ce vocabulaire peut être appris par certains types de systèmes de reconnaissance. Cette étape est appelée apprentissage (fig. 2.1). Pour chaque unité de ce vocabulaire, une classe ou un vecteur de référence doit être déterminé. Le processus de décision est chargé de comparer le vecteur de paramètres de représentation de la forme reconnue avec les différents vecteurs de référence et choisir celui qui est le plus proche.

Les systèmes de reconnaissance des formes en général et des signes en particulier, présentent différentes étapes de traitements plus ou moins développées suivant les diverses applications. La nature des informations contenues dans l'image initiale et les informations utilisées dans la reconnaissance font que plusieures étapes sont nécessaires pour arriver jusqu'à la décision. Le rôle de ces étapes est de réduire successivement la taille de l'information pour la rendre utilisable par le module de décision. Les principales étapes rencontrées [5, 4, 23] sont généralement présentées comme dans la figure 2.1.

2.1.2 Reconnaissance des gestes

Jusqu'alors, le terme geste a été utilisé pour représenter une information quelconque portée par une partie du corps en mouvement. Dans le domaine de la reconnaissance des

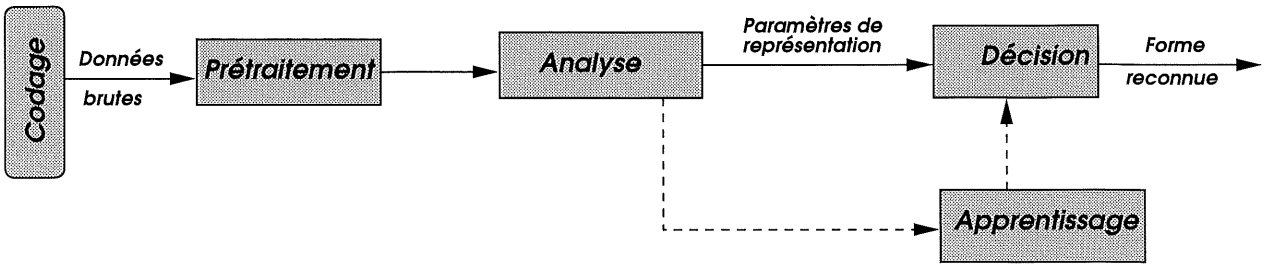


Figure 2.1: *Les principales étapes de la reconnaissance des formes.*

gestes, les approches utilisées dépendent du type de gestes à étudier. La forme de base peut être soit statique, soit dynamique. La terminologie employée ici est la suivante :

- une posture correspond à un vecteur de données à un instant t. Il peut être constitué d'une configuration, d'un emplacement et d'une orientation de la main. Ces paramètres ont été présentés en détail dans le chapitre précédent.
- Un geste est une séquence de postures.
- Les postures ou les gestes peuvent être étudiés séparément. Nous parlerons alors de postures ou de gestes isolés.
- Des postures ou des gestes sont connectés s'ils sont exécutés les uns à la suite des autres sans qu'il y ait de recouvrement entre les gestes ou les postures.
- Des gestes sont enchaînés si la fin d'un geste est modifiée en fonction du début du geste suivant et si le début du geste suivant est modifié en fonction de la fin du geste précédent. Il s'agit du phénomène de coarticulation.

Dans la paragraphe suivant, nous étudierons les techniques utilisées pour la reconnaissance de gestures en générale et de postures en particulier qui utilisent une variété de capteurs d'informations (gants numériques, caméra, etc.) et nous concluons par la suite,

le type de capteur, le type de représentation ainsi que le modèle de reconnaissance que nous avons choisi pour arriver à notre objectif.

2.2 Étude et analyse des systèmes existants

Cette section a pour objectif de décrire les méthodes de reconnaissance des gestes statiques et dynamiques existantes selon les caractéristiques, les modèles de représentation et de reconnaissance utilisés et les résultats obtenus.

La reconnaissance de gestures peut être vue comme un problème de reconnaissance de formes dont lequel les formes à classifier sont des images de postures ou des sorties de capteurs de pose. Les techniques utilisées pour la reconnaissance de gestes sont très variées et même si elles prennent en compte une ou plusieures des difficultés énumérées dans le chapitre 1, rares sont celles qui permettent de toutes les gérer.

La reconnaissance de gestures se distingue de la reconnaissance de postures par le fait qu'elle requièrt la reconnaissance des formes de la main non seulement dans l'espace mais aussi dans le temps. Nous allons examiner ces techniques selon leurs types de représentation, leurs types de décision et leurs performances.

2.2.1 Analyse et représentation

Cette étape dépend de beaucoup de paramètres et varie énormément suivant les systèmes considérés. Son rôle essentiel, en plus de réduire la taille de l'information manipulée, est de fournir des entrées assez pertinentes à l'algorithme de décision pour que celui-ci soit efficace. Les méthodes de représentation peuvent être classées selon le type de paramètres ou de caractéristiques de représentation qu'ils fournissent. Les trois méthodes les plus répandues en reconnaissance des gestes sont les prototypes (template), la discrétisation de l'espace en zone et les caractéristiques cinématiques et géométriques.

Les prototypes

L'élaboration des prototypes ne nécessite aucun calcul : il s'agit simplement des données brutes captées en entrée sous leur forme brute. Ces techniques sont généralement utilisées dans la technologie des gants numériques. S'il est question de posture, C. Gourley [19, 33] a considéré les prototypes comme étant des vecteurs constitués de 10 valeurs de flexion des doigts. Parfois, les trois valeurs d'orientation sont également utilisées [32]. S'il s'agit de gestes dynamiques, les prototypes sont constitués d'une séquence complète de postures (doigts, orientation, position) d'une taille fixe [24].

Le défaut des prototypes est qu'ils ne prennent pas en compte la variabilité du signal. Un geste donné n'est jamais exactement reproductible. Il faut tenir compte des variations du signal qui peuvent avoir de multiples sources (utilisateur, capteur) et des variations inter-utilisateurs.

Afin de diminuer la taille du vocabulaire, une autre méthode consiste à calculer, à partir des données, les regroupements de postures selon leur ressemblance. Ces regroupements sont utilisés pour définir des codes pour chaque posture [42]. Même avec cette méthode, le vecteur de représentation obtenu n'est pas toujours très représentatif, car il est calculé sur une petite taille de vocabulaire et la variabilité du signal n'est pas très bien représentée.

La discrétisation de l'espace en zones

Cette technique est généralement utilisée dans le cas d'acquisition des données avec des dispositifs électroniques externes. La discrétisation de l'espace en zones est une technique simple qui consiste à segmenter l'espace de représentation des données en zones délimitées par des valeurs fixées a priori. Cette division de l'espace en zones permet d'apporter une certaine souplesse dans la représentation du geste. Les gestes sont décrits comme faisant partie d'un interval de valeurs possibles, plutôt que par des valeurs uniques. Cela permet de diminuer la taille du vocabulaire. Ces zones peuvent être utilisées pour représenter des postures ou des gestes. Dans le cas des postures, les zones peuvent correspondre à

des intervals, par l'intermédiaire de valeurs de flexion minimale et maximale pour chaque doigt, comme par exemple, la flexion des doigts pour le signe «M» est comprise entre 20^o et 90^o [42].

Avec ce type de représentation, il est nécessaire de définir des valeurs limites, qui sont généralement fixées de manière arbitraire ou empirique. Si le geste subit une variation importante, il risque de se produire des erreurs au niveau de la représentation, qui vont fausser l'étape de décision. Nous pouvons aussi ajouter que ce type de représentation n'est pas très robuste à la variabilité du signal.

Les caractéristiques cinématiques, géométriques et topologiques

C'est la technique la plus populaire dans le cas d'utilisation de caméras. Les caractéristiques cinématiques et géométriques sont des informations globales sur le geste ou la posture qui nécessitent des calculs plus ou moins complexes. Les caractéristiques cinématiques sont utilisées dans le cas de gestures dynamiques pour calculer la vitesse, l'accélération etc.. Dans le cas de postures, nous utilisons généralement les caractéristiques géométriques, photométriques et topologiques concernant la forme du geste (rayon de courbure, moments, histogrammes, chaînes numériques, etc.). Ce type de représentation est souvent utilisé en reconnaissance de geste 2D. Dans ce cas, les caractéristiques sont calculées sur l'image de niveaux de gris ou sur le contour représentant l'image du signe. Parmi les travaux de référence, nous pouvons citer Grimson [20] et Ayache [3] qui ont utilisé des informations géométriques calculées sur les contours des images des courbes planaires. Ils n'ont pas obtenus un taux de reconnaissance élevé mais leur approche est simple à mettre en oeuvre. Pour ajouter de l'efficacité à ces systèmes, un nouvel outil topologique a été proposé récemment par C.Uras et A. Verri [49, 45, 47, 48] pour l'analyse du langage des signes. Dans leurs récents travaux, C. Uras et A. Verri ont utilisés un nouvel outil topologique appelé *fonctions de taille* qui permet non seulement d'extraire les informations métriques (quantitatives) et géométriques, mais aussi, les informations

topologiques (qualitatives) liées aux contours des images des postures. Nous allons décrire ce type de représentation dans les chapitres 4 et 5, car ils ont obtenu un bon taux de reconnaissance. D'autres types de représentation basés sur le calcul de caractéristiques, ont été proposés récemment pour une taille de vocabulaire restreinte. Wen [15] a utilisé la dérivée de la chaîne numérique et il a constitué un vecteur de caractéristiques de dimension 80. Il existe aussi d'autres techniques utilisées directement sur les images de signes à niveaux de gris. Nous pouvons citer Freeman [12] qui a pu introduire le calcul de l'histogramme d'orientation lié aux images de niveaux de gris des signes pour construire des vecteurs de caractéristiques pour des fins de reconnaissance de quelques dizaines de postures de commandes spécifiques.

Dans [27], plusieurs essais de reconnaissance de gestures dynamiques ont été effectués sur des caractéristiques différentes, telles que la distance totale et l'énergie de flexion/extension des doigts, la taille de la boîte englobante, ainsi que le calcul d'histogrammes sur les caractéristiques précédentes. Mais cette étude ne fournit pas d'analyse ni de justification sur le choix des caractéristiques. Celles qui ont été gardées sont celles pour lesquelles le taux de reconnaissance est le meilleur (80% de taux de reconnaissance sur 95 gestes de l'AusLan, le langage des signes australien).

Nous pouvons conclure que les prototypes sont simples à mettre en oeuvre mais ils doivent être associés à une étape d'apprentissage portant suffisamment d'exemples si l'on veut bénéficier des connaissances statistiques représentatives des données. La discrétisation de l'espace en zones est simple à mettre en oeuvre, mais elle ne respecte pas le critère de continuité exposé dans le chapitre 1. Ce type de représentation n'est pas suffisamment robuste par rapport à la variabilité du signal.

La représentation à base de caractéristiques cinématiques et géométriques ou encore topologiques paraît plus complexe à mettre en oeuvre que les deux autres types car il faut choisir les caractéristiques les plus adaptées au signal à traiter et à une taille de vocabu-

laire largement suffisante. Dans toutes les études utilisant ce type de représentation, le choix d'un type de caractéristiques se fait d'une manière empirique ou en fonction de la composition du vocabulaire. Si nous disposons d'une bonne connaissance du vocabulaire à analyser comme dans le cas de l'alphabet du langage des signes italien, ce type de représentation est très souhaitable. Nous avons donc choisi ce type de représentation à base de caractéristiques géométriques et topologiques (moments, fonctions de taille) (voir le chapite 5).

Une fois le signal brut acquis et représenté sous forme de vecteur de caractéristiques, la seconde étape consiste à donner ce vecteur comme entrée au module de décision. Les différentes techniques de décision sont présentées dans le paragraphe suivant.

2.2.2 La décision

Parmi les différentes techniques existantes, certaines ne fonctionnent qu'avec un type de représentation, d'autres sont plus générales. Les processus de décision les plus couramment utilisés en reconnaissance de gestes statiques et dynamiques, sont l'approche linguistique, la comparaison de prototypes, les réseaux connexionnistes et les modèles de Markov cachés.

Approche linguistique

Les approches linguistiques consistent à appliquer la théorie des automates et des langages formels au domaine de la reconnaissance des formes. Elles sont généralement utilisées dans le cas de gestures dynamiques avec la technologie des gants numériques comme dans le cas de l'application effectuée par C. Hand [21]. Le processus de représentation fournit une séquence de lexèmes représentant les postures. Le processus de décision est constitué d'un ensemble de règles qui permet d'associer des suites de lexèmes.

C. Hand [21] a utilisé cette approche. Les lexèmes sont constitués de huit configurations statiques, et les règles définissent six gestes qui sont des séquences de lexèmes, parfois

associés à un mouvement rectiligne simple. Le taux de reconnaissance obtenu est médiocre (de 15% à 80% selon le geste).

L'approche linguistique a été abandonnée dûe au fait de ces mauvaises performances (50 % de taux de reconnaissance contre 80% pour la deuxième approche). Cette méthode très rigide ne semble pas adaptée pour le traitement de données très variables telles que celles issues des gestes.

Comparaison de prototypes

La méthode par comparaison de prototypes est sans doute la méthode la plus simple et la plus naturelle. Dans cette technique, une image de signe est représentée par un vecteur de caractéristiques. Un certain nombre de vecteurs types est stocké et le prototype présenté est alors supposé appartenir à la classe à laquelle il est le plus proche. En général un seuil de reconnaissance est défini, en-dessous duquel l'entrée est rejetée car elle n'est pas assez proche d'une des classes types. Les algorithmes qui mettent en oeuvre cette méthode diffèrent essentiellement par la distance qu'ils utilisent : la distance entre les listes, la distance entre les chaînes [51], la somme de la valeur absolue des différences [42] et finalement, le produit scalaire entre deux vecteurs. [37].

Nous pouvons noter que les taux de reconnaissance obtenus avec ce type de processus de décision varient entre 65% et 80% et sont parmis les moins bons, comme soulignés dans le tableau récapitulatif au paragraphe 2.4.

Les méthodes de type comparaison de prototypes sont faciles à développer et économiques du point de vue informatique (temps de calcul, mémoire). Cependant, elles nécessitent une segmentation préalable du signal suivie de l'envoie de chaque geste segmenté au système de reconnaissance.

L'approche statistique et structurelle

La plus couramment utilisée est l'approche bayesienne. Elle a été formalisée par Chow en 1965 [11] et consiste à représenter le problème sous la forme d'un modèle probabiliste.

Elle estime la classe d'appartenance d'un prototype avec un minimum d'incertitude et en plus, estime le risque de cette décision. Cette approche est basée sur le théorème de Bayes appliqué aux distributions de probabilité.

La méthode des k-plus proches voisins est utilisée dans [12, 47, 48]. Son principe est simple, il consiste à considérer les k-plus proches voisins du prototype présenté. La classe du prototype sera celle qui est la plus représentée dans ce voisinage. Parmi les systèmes basés sur cette approche, nous pouvons nous référer aux travaux de C. Uras [48, 47] dont le type de représentation consiste en des vecteurs de caractéristiques obtenus à base de fonctions de taille (ce concept est utilisé dans ce mémoire et sera détaillé plus tard) et la règle des k-plus proches voisins est utilisée pour classifier les signes de l'alphabet du langage des signes italien. Le taux de reconnaissance obtenu varie entre 74% et 90%. De même, W. Freeman [12] a utilisé la technique des k-plus proches voisins pour classifier une dizaine de postures qui sont représentées par des vecteurs de caractéristiques à base d'histogrammes d'orientation et il a obtenu un taux de reconnaissance de 80%.

Une autre méthode est celle des modèles de Markov cachés qui sont surtout utilisés dans le domaine de reconnaissance de gestes enchaînés. Parmi les systèmes déjà conçus à l'aide de modèles de Markov cachés, nous pouvons citer Thad Starner [41]. Le capteur utilisé est une caméra et le vocabulaire est constitué de gestes enchaînés. De bons résultats sont obtenus avec l'aide d'une grammaire sur des gestes pour lesquels la configuration n'est pas discriminante.

Les modèles de Markov cachés permettent une double approche de reconnaissance, à la fois statistique et structurelle. Ils sont capables de prendre en compte l'ensemble des variabilités du signal sans imposer un traitement arbitraire préalable par seuillage ou équivalent mais des problèmes de temps de calcul limitent les performances des applications [41, 29].

L'approche connexionniste

C'est la technique la plus récente, elle utilise les réseaux de neurone [15, 24, 19, 32]. Ceux-ci sont des classifieurs qui établissent des frontières, linéaires ou non, dans des espaces de dimension finie [26]. L'apport des techniques neuronales en classification automatique est important du fait qu'elles permettent un apprentissage avec peu de spécifications des règles de décisions et qu'elles sont relativement simples à mettre en oeuvre. Un réseau de neurone est une structure constituée d'éléments de base simples, connectés entre eux : les neurones formels. Chaque neurone réalise une fonction de discrimination linéaire. L'utilisation du réseau nécessite l'emploi d'un algorithme d'optimisation itératif. Nous ne connaissons pas de moyens pratiques pour assurer la convergence de cet algorithme, cependant, il existe simplement pour chaque cas une configuration (nombre de couches et nombre de neurones dans chaque couche) qui mène à la convergence. Il existe plusieurs types de réseaux de neurone qui diffèrent essentiellement par la façon dont les neurones sont connectés et le type d'apprentissage qu'on leurs applique. La plupart des réseaux connexionnistes utilisés pour reconnaître des postures sont de type Perceptron Multicouches (PMC).

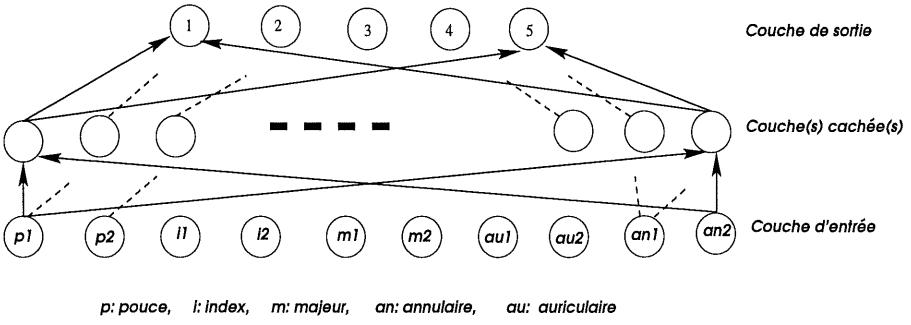


Figure 2.2: *L'exemple de PMC sur un vocabulaire de 5 postures utilisées par Gourley.*

Trois couches successives sont généralement utilisées, la couche d'entrée, la couche cachée et la couche de sortie. Chaque neurone d'une couche est relié à tous les neurones de la couche suivante comme dans la figure 2.2 qui correspond à l'architecture du réseau de

neurone utilisé par Gourley [19]. C'est à la couche d'entrée que sont fournies les données issues du processus de représentation, après la normalisation. Elle possède un nombre de cellules égales à la taille du vecteur de représentation. Chaque unité est associée à une valeur du vecteur. La couche cachée est de taille variable (choisie par expérimentation). En général, des tests sont faits avec différentes tailles afin de déterminer la taille optimale à utiliser. Le nombre d'unités de la couche de sortie correspond à la taille du vocabulaire (5 dans le cas de l'étude effectuée par Gourley [19]). Chaque unité est alors associée à une posture du vocabulaire.

Dans [15], une approche originale a été proposée. À partir d'une analyse des images des signes, il a obtenu un vecteur de caractéristiques de dimension 80. Des groupes de ressemblance entre postures ont été définis. Une cascade de 7 sous-réseaux rétro-propagation (BP) est utilisée (fig. 2.3). À chaque niveau de la cascade, des hypothèses sont éliminées. À la dernière étape, une unique solution est choisie.

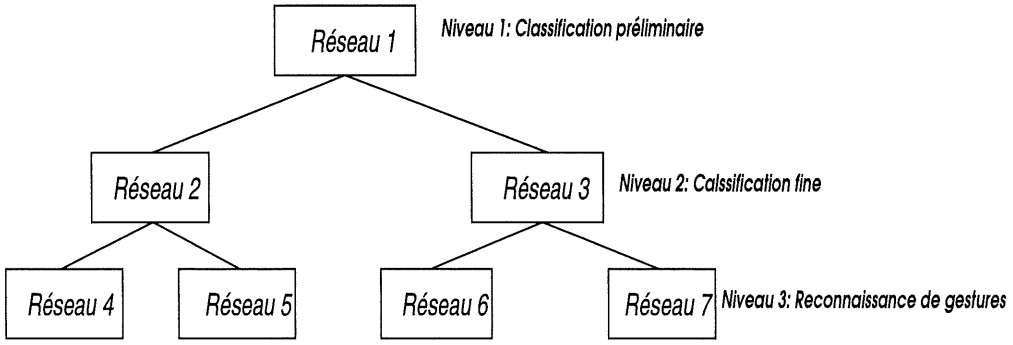


Figure 2.3: *L'architecture de sous-réseaux BP multi-couches en cascade utilisée par Wen.*

L'intérêt de cette approche réside au fait que l'apprentissage est rapide grâce à la spécialisation des différents réseaux. Wen a obtenu un taux de reconnaissance de 80% à 97% sur 13 postures effectuées par une vingtaine de personnes.

Parmi toutes les études basées de PMC, les taux de reconnaissance obtenus varient entre 80% et 95% pour des corpus composés de postures isolées dont la taille varie de 5 à

42 postures. Les meilleurs résultats sont obtenus par K. Murkami [32] qui a utilisé la technique des prototypes au niveau de la représentation et un réseau connexionniste pour la reconnaissance. Il a obtenu un taux de reconnaissance de 98% sur un vocabulaire de 42 postures représentant l'alphabet manuel japonais, mais un temps d'apprentissage de plusieures heures est nécessaire sur une station Sun4.

Du fait de l'étape d'apprentissage, les réseaux connextionnistes sont capables de reconnaître une forme complète bruitée ou incomplète et ils sont capables de généraliser. Il faut noter que l'approche connexionniste est majoritairement utilisée dans les différentes applications de reconnaissance des postures statiques et dynamiques.

2.3 Débat : Dispositifs électroniques VS vision

À la lumière de ce que nous avons exposé précédemment, nous pouvons distinguer dans la littérature deux catégories d'approches utilisées dépendamment de la technologie utilisée pour l'acquisition et l'obtention des informations décrivant le geste : Les approches à base de détecteurs électroniques et les approches à base de caméras.

2.3.1 Approches basées sur des dispositifs électroniques

Ces techniques nous permettent de mesurer les gestes à l'aide d'un dispositif électronique (gant numérique, styli, position-tracker, etc.) [24, 32, 27, 42]. À l'aide de ces valeurs numériques, nous pouvons obtenir une variété de détails sur la position, l'orientation et le mouvement de la main.

Les problèmes techniques de capture de gestes semblent, à ce jour, encore trop importants pour pouvoir élaborer de bons résultats dans des conditions normales. D'une manière générale, les capteurs de gestes ne permettent pas encore une saisie suffisamment fiable et riche comme c'est le cas des microphones en parole. Même si nous disposions d'un système performant pour mesurer les gestes des deux mains, les langues des signes sont

aussi constituées de gestes du corps, du visage et même de quelques émissions sinon sonores du moins buccales. De plus, des contacts entre les mains et différentes parties du corps sont souvent effectués. Nous ne disposons pas, à ce jour, de capteurs permettant de mesurer toutes les informations utiles.

2.3.2 Approches visuelles

Ces techniques utilisent des caméras qui permettent de capter une personne exécutant un signe. Divers auteurs ont utilisé cette approche dans la littérature dont on peut citer Starner [40, 41] qui s'est servi à la fois d'une caméra et de deux gants simples colorés (jaune et orange). C. Uras et A. Verri quant à eux, ont utilisé seulement une caméra pour l'analyse de l'alphabet du langage des signes italien afin d'obtenir des images à niveaux de gris sur lesquelles on a procédé par l'extraction de caractéristiques sur les contours des images sans utiliser aucun autre dispositif additionnel. De même, W. Gao [15] et W. T. Freeman [12] ont utilisé les images à niveaux de gris en choisissant d'autres types des caractéristiques.

Pour bénéficier des techniques de traitement d'image, nous avons opté pour une approche visuelle. Les approches visuelles sont avantageuses puisque les utilisateurs ne sont pas encombrés par des périphériques externes complexes. Cependant, ces approches requièrent d'immenses calculs pour les traitements de l'image et l'extraction des caractéristiques avant de passer à la représentation des données, donc, il suffit de trouver des techniques de prétraitement efficaces pour remédier à ce problème.

2.4 Applications et corpus

Ce paragraphe synthétise sous la forme d'un tableau (voir tableau 2.1), les principales études portant sur la reconnaissance de postures et de gestures présentées précédemment et pour lesquelles sont indiqués la spécification du capteur utilisé (gant ou caméra), des processus de représentation, de décision, la taille et le type de vocabulaire (pos-

tures/gestes dynamiques, isolés/enchaînés), et enfin le taux de reconnaissance.

Tableau 2.1: *Le tableau récapitulatif des performances des systèmes existants.*

Nom caract.	Taille Voc.	Type Voc.	Capt.	Rep.	Déc.	Post. Gest.	Taux %
Gourley	26	ASL[1]	Elect.[0]	Prot.[12]	PMC[4]	P	95
Harling	5	ASL[1]	Elect.[0]	Prot.[12]	PMC[4]	P	96
Harling	3	ASL[1]	Elect.[0]	Prot.[12]	PMC[4]	G	90
Murkami	42	JSL[2]	Elect.[0]	Prot.[12]	PMC[4]	P	98
Murkami	10	JSL[2]	Elect.[0]	Prot[12]/carac.[6]	NN Récurrent	G	96
Takahashi	46	JSL[2]	Elect.[0]	Zones	Comp. Prot.[5]	P	65
W. Gao	13	Prédéfini	Caméra	C.C[7]	NN BP	P	80
C. Uras (1)	25	ISL[16]	Caméra	F.T[8]	k-voisins	P	85
C. Uras (2)	25	ISL[16]	Caméra	F.T[9]	k-voisins	P	92
Freeman	15	Prédéfini	Caméra	H.O[1]	k-voisins	P	75
Grimson	25	ASL[1]	Caméra	Géom.[14]	k-voisins	P	75
Starner	40	ASL[1]	Caméra	Prot[12]/Carac.[6]	HMM[11]	G	95
Kadous	95	AUSLAN[15]	Elect.[0]	Carac.[1]	Linguistique	G	50
Kadous	95	AUSLAN[15]	Elect.[0]	Carac.[1]	Comp. Prot.[5]	G	80
Tamura	20	JSL[2]	Caméra	Carac.[1]	ABD[3]	G	45
Hand	14	Prédéfini	Elect.[0]	Prot.[1]	Linguistique	G	45

0 : Dispositifs électroniques (PowerGlove, DataGlove et CyberGlove)

1 : American Sign Language

2 : Japanese Sign Language

3 : Arbre binaire de décision

4 : Réseau connexionniste de type perceptron

5 : Comparaison de prototypes

6 : Caractéristiques cinématiques et géométriques

7 : Chaîne Code de Freeman

8 : Première famille des fonctions de taille

11 : Modèles de Markov cachés

12 : Représentation par la méthode de prototype

14 : Caractéristiques géométriques

15 : Australian Sign Language

16 : Italian Sign Language

10 : Histogramme d'orientation

9 : Deuxième famille des fonctions de taille

D'autres études ont porté sur des gestes artificiels, formant un vocabulaire de petite taille dédié à une application bien spécifique [12, 15]. Ces applications portent en général sur l'utilisation du geste dans des applications multimédias et la réalité virtuelle. Nous pouvons constater que la reconnaissance des postures peut être basée sur un vocabulaire de configuration quelconque (Freeman [12], Gao[15]), mais la plupart du temps, elles sont basées sur l'alphabet manuel.

Toutes les études indiquées dans ce tableau obtiennent des taux de reconnaissance qui varient de 65% à 95% lorsque les postures sont isolées. Nous pouvons constater aussi que les réseaux connexionnistes sont les plus utilisés dans le cas des postures. Les approches linguistiques et les arbres binaires de décision utilisés par Tamura et hand sont ceux qui donnent les moins bons résultats, sans doute parce que l'étude porte sur des lettres enchaînées dans le cas de Tamura et Hand. Par ailleurs, le vocabulaire n'est pas constitué de lettres de l'alphabet qui peuvent être ambigües, mais de configurations statiques toutes distinctes, ce qui relativise les bons résultats obtenus.

Les études qui fournissent les meilleurs résultats dans le cas des postures, sont celles qui utilisent les réseaux connexionnistes (Gourely, Harling et Murkami) avec un capteur électronique. Par contre, dans le cas des gestures, Starner a obtenu de bons résultats en utilisant les modèles de markov mais en dépit du temps de calcul. Notons que pour la reconnaissance des postures et dans le cas d'utilisation d'une caméra, la meilleure performance est obtenue par C.Uras et A.Verri qui ont utilisé des caractéristiques géométriques et topologiques au niveau de la représentation et la méthode de k-plus proches voisins au niveau du processus de décision sur une taille de vocabulaire significative.

Pour les gestes, les méthodes les plus prometteuses sont les réseaux connexionnistes de type récurrent, la comparaison dynamique et les HMM. Les HMM permettent simultanément de segmenter et de reconnaître des phrases gestuelles sans imposer un prétraitement. Si nous travaillons avec un vocabulaire de postures, la méthode par comparaison de prototypes et les réseaux connexionnistes conviennent.

2.5 Conclusion

Dans le cas où l'application utilise des gestes isolés (postures), en nombre peu important, il est possible de se contenter d'un système tel que celui de W. Freeman ou W.Gao, en prenant bien soin de choisir des caractéristiques cinématiques et géométriques qui

ne nécessitent pas d'utilisation de seuils. L'apprentissage est beaucoup moins fastidieux que pour les réseaux connexionnistes et les HMM, car le nombre d'exemples nécessaires par classe est faible. Dans le cas des tailles de vocabulaires considérables comme pour l'alphabet manuel d'un langage de signes, il est préférable d'utiliser des caractéristiques pertinentes comme celles utilisées par C. Uras et A. Verri, ainsi qu'un système de reconnaissance basé sur un réseau connexionniste qui permet de prendre en compte plusieurs prototypes de postures. Si le type de vocabulaire envisagé comporte des gestes enchaînés ou connectés, il est préférable de choisir un outil tel que les HMM qui permet de représenter les aspects de variabilités temporelles et spaciales.

Comme nous nous intéressons en priorité à l'alphabet du langage des signes italien, nous avons constitué une base de données de signes statiques correspondant aux 25 lettres de l'alphabet. Cela a nécessité de construire au préalable différents outils permettant d'une part, de tester les paramètres de représentation adéquats et d'autre part, de préparer les corpus d'apprentissage afin de tester notre système de reconnaissance. Ces outils sont présentés dans les chapitres 3 et 5.

L'approche connexionniste étant celle que nous avons choisie, elle sera donc exposée plus en détail dans le chapitre 5, parce qu'avec cette technique de décision, plusieurs types de représentation peuvent être utilisés. Dans le cas de notre application, nous avons choisi d'utiliser le type de représentation à base de caractéristiques géométriques et topologiques. Cette approche est prometteuse car elle permet de prendre en compte l'ensemble des variabilités du signal exprimées sous forme de vecteurs de caractéristiques obtenus dans l'étape de représentation.

Pour toutes ces raisons, nous avons choisi de baser notre système de reconnaissance sur les réseaux connexionnistes. Il reste le problème de représentation évoqué au début de ce chapitre, notre modèle de représentation à base de caractéristiques géométriques et topologiques devra tenir compte des aspects présentés dans le chapitre 1.

CHAPITRE 3

Évaluation des caractéristiques

3.1 Introduction

Ce chapitre expose les moyens de décrire ou de modéliser les postures qui représentent les lettres de l'alphabet de la langue des signes (voir fig. 3.2 et 3.3). Comme nous l'avons déjà précisé dans les sections précédentes, il n'y a pas de théorie formelle ni de technique générale d'extraction de caractéristiques qui permettent d'identifier tout type d'objet. Dans les chapitres précédents, nous avons décrit les paramètres essentiels du langage des signes italien ainsi que les différentes techniques utilisées jusqu'à maintenant pour la reconnaissance des gestures en général. Dans ce qui suit, nous allons proposer et étudier plusieurs types de caractéristiques pour la représentation des postures de l'alphabet du langage des signes. Nous allons discuter de l'invariance de ces caractéristiques et de leurs pertinences dans le cas particulier de la reconnaissance des postures de l'alphabet du langage des signes italien. En général, un système de reconnaissance gestuelle se divise en deux modules, la détection de la main dans l'image sur un fond de scène parfois aléatoire et la reconnaissance proprement-dite de la gesture de la main. Le premier module de détection de la main, appelé prétraitement, reste un problème important dans la vision

artificielle. Il permet d'extraire les formes ou les silhouettes des images des signes. Nous ne traitons pas ce problème dans notre étude, nous considérons que les données de notre système sont des images contenant seulement une main blanche sur un fond de scène noir. Il n'y a pas une seule technique de représentation qui sera générale et adaptée à tous les problèmes de reconnaissance. La similarité entre les vecteurs de caractérisations des formes des objets est interprétée comme une similarité entre les objets eux-mêmes. Par conséquent, la capacité d'une caractéristique donnée de représenter d'une façon unique l'objet à partir de l'information disponible, détermine la pertinence de cette caractéristique. Pour ce faire, nous allons d'abord fixer les critères qui nous permettent de procéder à l'évaluation de ces caractéristiques pour la représentation des gestures du langage des signes :

- l'invariance par rapport à l'échelle puisqu'il existe des variations entre les tailles des mains des personnes dans l'exécution des différents signes (variations inter-personnes).

- L'invariance par rapport à la translation pour pouvoir produire le même vecteur de caractéristiques pour deux signes identiques qui se situent dans des positions différentes dans l'image.

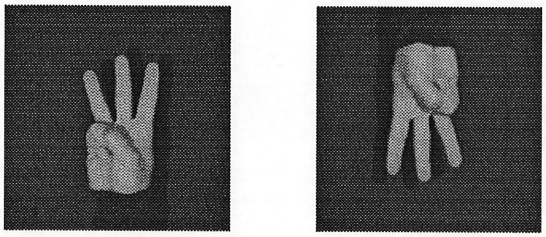
- L'invariance aux changements de l'illumination car les images des signes peuvent être acquises dans des conditions d'illumination différentes. Une bonne caractéristique doit être robuste si nous multiplions l'image par une constante en y ajoutant une autre constante $(aI + b)$.

- La pertinence : Une caractéristique doit pouvoir représenter d'une façon unique un objet et elle doit être largement suffisante pour distinguer entre les différents signes.

- La complexité de calcul pour produire un système efficace.

- La non-invariance par rapport aux nombres de doigts et leur ordre d'apparition, la robustesse contre les changements minimes dans les mains, les poses et le style des signes car il existe des variations intra-personnes dans l'exécution des signes.

- La non-invariance par rapport à l'orientation des gestures puisqu'il existe plusieurs signes qui ont des formes relativement similaires mais avec des orientations différentes (voir fig 3.2). De plus, nous ne devons pas confondre un signe inconnu similaire à un signe existant mais avec une orientation différente (voir fig. 3.1). Nous devons noter aussi qu'une bonne caractéristique doit être tolérante pour des changements minimes dans l'orientation apparente des signes.



(a) *(b)*

Figure 3.1: *La non-invariance à l'orientation des postures : (a) Le signe « W» et (b) le signe inconnu, sont relativement similaires mais avec une différence d'orientation de 180°.*

Dans ce qui suit, nous allons évaluer les caractéristiques suivantes pour l'analyse de l'alphabet du langage des signes selon les critères cités précédemment :

- les chaînes numériques;

- Le descripteur de Fourier;

- Les moments d'inertie;

- Les histogrammes de niveaux de gris et les histogrammes d'orientation.

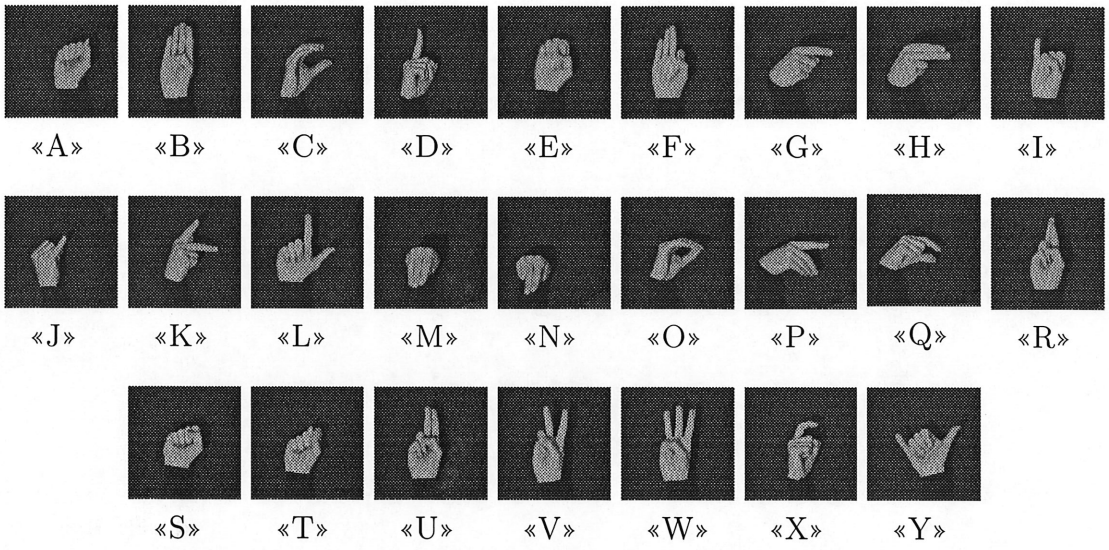


Figure 3.2: *La série des signes réalisée par A. Verri.*

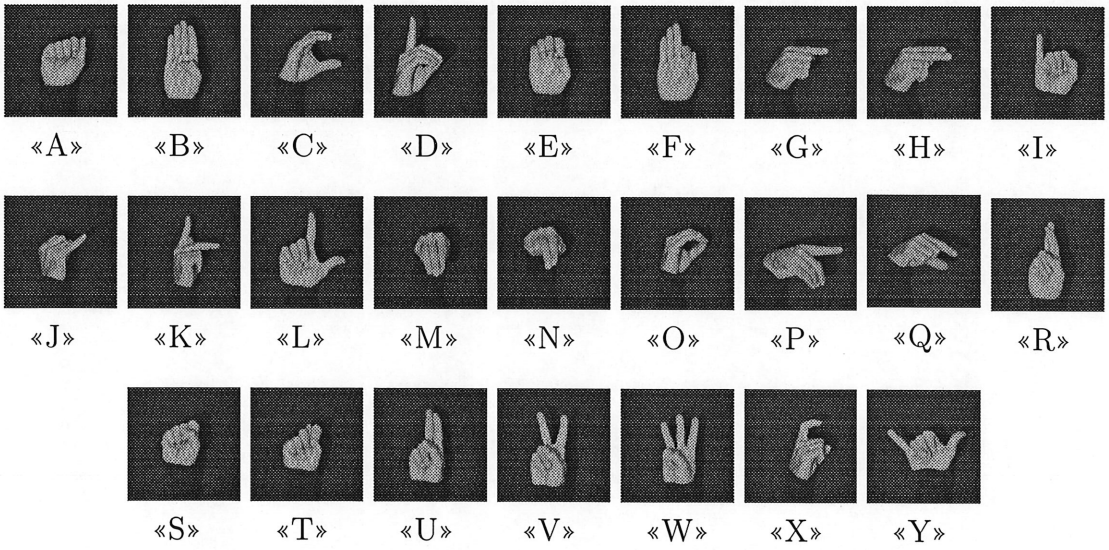


Figure 3.3: *La série des signes réalisée par C. Uras.*

3.2 Les chaînes numériques

Les techniques d'extraction des primitives liées aux contours (chaînes numériques,moments, courbures) sont très utilisées dans plusieurs systèmes de reconnaissance de gestures de la main [15, 16, 2, 52]. Supposons que le contour d'une image de gesture est fermé. Ce descripteur est basé sur le codage de Freeman. Il consiste à coder chaque déplacement entre deux points successifs par un chiffre allant de 0 à 7, si nous considérons les 8-voisins ou par un chiffre allant de 0 à 3 dans le cas des 4-voisins (voir fig 3.4(a)). Ce codage est la méthode d'approximation angulaire la plus simple. La figure 3.4 décrit la définition de la chaîne code d'un contour dans le cas des 4-voisins. Un exemple de codification en chaîne de base d'une image planaire quelconque est donné dans la figure 3.4(c), la séquence de code commence à partir d'un point de départ signalé dans la figure 3.4(b). Bien qu'elle

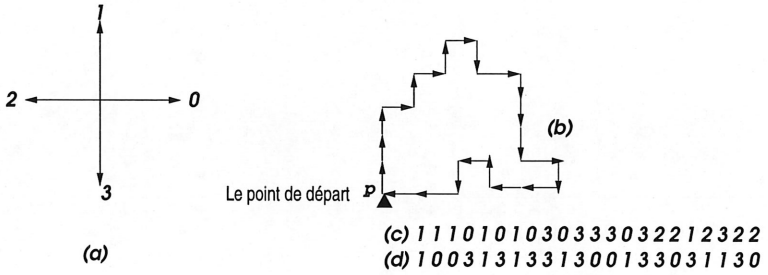


Figure 3.4: *La chaîne numérique et sa dérivée: (a) représente les nombres de direction avec la règle des 4-voisins, (b) représente un contour quelconque, (c) représente la chaîne numérique du contour et (d) représente la dérivée de (c).*

est invariante à la translation, l'inconvénient essentiel de cette représentation est qu'elle est étroitement liée à un repère de référence (le point de départ) et cette représentation est très sensible à l'orientation de l'objet codé. En effet, deux contours peuvent être semblables et avoir deux représentations différentes. Cette représentation est également sensible à la taille de la main puisque le code de Freeman dépend aussi de la longueur du contour des gestures. Il est clair que les chaînes numériques simples décrites précédem-

ment sont sensibles à la direction du suivi de contour. Pour obtenir une représentation non sensible à l'orientation de l'objet et à la direction de la chaîne, il suffit de considérer la dérivée de la chaîne numérique [15]. Cette dérivée fournit une autre séquence de nombres indiquant la direction relative des segments de la chaîne numérique (voir fig. 3.4(d)). Il est clair que la dérivée de la chaîne numérique est invariante à la translation puique la dérivée ne dépend pas de l'emplacement de la forme dans l'image. Par contre, la dérivée de la chaîne numérique dépend du point initial choisi. Par conséquent, ce type de caractéristique n'est pas invariant à l'échelle mais invariant à la rotation de l'objet. Dans le cas de l'analyse des postures, pour obtenir une représentation pertinente, W. Gao [15] a utilisé des filtres passe-bas permettant de distinguer les signes en fonction de l'état et de l'ordre d'apparition des doigts et des vallées.

Ce type de caractéristique appelé aussi « le code de convexeté et de concavité» est invariant à la translation, à l'échelle et à la rotation. Pour mieux comprendre, considérons l'exemple de la figure 3.5. La figure 3.5(a) montre le contour de la gesture «L» et sa repré-

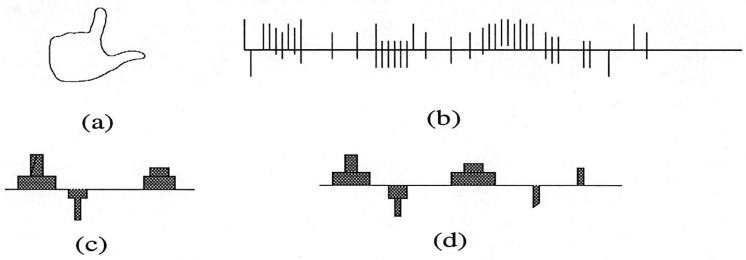


Figure 3.5: *La chaîne code d'une gesture : (a) l'image de contour de la gesture «L», (b) la dérivée de la chaîne numérique, (c) le code de convexeté et de concavité en utilisant un filtre passe-bas et (d) le code de convexeté et de concavité en utilisant le filtre passe-bas à une autre échelle.*

sentation en code de convexeté et de concavité qui est montré par la figure 3.5(b). Comme nous pouvons le constater, la représentation par la dérivée de la chaîne numérique, après avoir appliqué le filtre passe-bas dans la figure 3.5(d), illustre mieux le nombre de doigts

et le nombre de vallées perçus de la gesture, la netteté de chaque doigt, la profondeur et la largeur de chaque vallée des doigts. Il est facile à comprendre que chaque convexe dans la chaîne code représente un pique (doigt) et chaque concave représente une vallée (espace entre deux doigts). Par contre, ce code n'est pas sensible à l'orientation des gestures, il est appliqué particulièrement dans un système de reconnaissance pour permettre seulement d'obtenir des informations sur le nombre de doigts et de vallées. Autrement-dit, deux formes similaires avec des orientations différentes peuvent produire une même chaîne de concavité et de convexeté. En plus, cette caractéristique ne différencie pas entre les doigts (index, majeur, etc.) et elle n'est pas sensible à l'ordre d'apparition de doigts et de vallées. Nous pouvons noter que cette caractéristique est pertinente seulement dans le cas où l'on traite des gestures indiquant les chiffres de «0» à «9» puisque ces signes se distinguent uniquement par le nombre de doigts et de vallées. Ce type de caractéristique est étroitement lié au contour des images des signes et il est moins sensible à l'illumination.

3.3 Descripteur de Fourier

C'est une caractéristique liée au contour des images. Puisque la position de tout point du contour fermé d'un objet est une fonction périodique, les séries de Fourier peuvent être utilisées pour l'approximation du contour. La précision de l'approximation de contour est déterminée par le nombre de termes dans les séries de Fourier [4, 23]. Supposons qu'un contour d'un objet est exprimé par une séquence de coordonnées :

$$z(t) = [x(t), y(t)], \quad pour \quad t = 0, 1, 2, ..., N-1$$

L'angle de courbure ψ du contour est calculé comme suit :

$$\psi = \tan^{-1} \frac{(dy/dt)}{(dx/dt)}$$

La courbure est une fonction réelle $Z(t)$ qui s'écrit comme suit : $Z(t) = \frac{d\psi(t)}{dt}$. La courbure $Z(t)$ étant continue, bornée et périodique par rapport à la longueur du périmètre P du contour, nous pouvons la développer en série de Fourier sous la forme monodimentionnelle suivante :

$$Z(t) = \sum_{n=0}^{N-1} C_n \exp \frac{(-2\pi i n)t}{N} \quad , \quad 0 \leq t \leq N-1$$

Le coefficient C_n s'écrit en fonction de $Z(t)$:

$$C_n = 1/N \sum_{t=0}^{N-1} Z(t) \exp \frac{(2\pi i n)t}{N} \quad , \quad 0 \leq n \leq N-1$$

Les coefficients complexes C_n sont appelés *les descripteurs de Fourier* du contour fermé. Ces résultats servent à définir la série de Fourier tronquée aux premiers M coefficients, ce qui est équivalent à mettre $C_n = 0$ pour $n > M-1$. Ces coefficients sont suffisants pour décrire le contour (M représente le nombre de points du contour). Il a été démontré [4, 23] que la série de Fourier est facilement normalisable par rapport à l'échelle du contour. Nous multiplions simplement les termes de $Z(t)$ par une constante, ce qui est équivalent, compte tenu de la linéarité du descripteur de Fourier, à multiplier en taille le contour par le même coefficient. La rotation d'un angle θ du contour s'obtient par la multiplication des termes de $Z(t)$ par $e^{i\theta}$ et la position du point d'origine est déplacée en multipliant le k^{eme} terme de $Z(t)$ par e^{ikt}. Donc, les descripteurs de Fourier sont invariants à la translation et à l'échelle mais, ne sont pas invariants à la rotation. Ces propriétés d'invariance rendent les descripteurs de Fourier attractifs pour la représentation des formes à base de contour. Cependant, leur utilisation pour l'analyse des postures du langage des signes n'est pas très appropriée puisqu'ils ne sont pas suffisants et pertinents pour représenter, de façon unique, les postures du langage des signes. D'autant plus, que deux signes différents peuvent produire deux vecteurs de caractéristiques identiques. Ils sont peu sensibles à l'illumination et le coût de calcul est considérable.

3.4 Les moments invariants

L'utilisation des moments pour l'analyse d'image et la représentation des objets sont inspirées par Hu [25] qui a montré que l'ensemble des moments $\{m_{pq}\}$ est uniquement déterminé par l'image $f(x, y)$ et inversement, l'image $f(x, y)$ est uniquement déterminée par $\{m_{pq}\}$. Puisque le support de l'image est fini, les moments de tout ordre existent et peuvent être calculés et décrivent, d'une façon unique, l'information contenue dans l'image. Pour caractériser toute l'information contenue dans l'image, un nombre infini de moments est nécessaire. L'objectif est de sélectionner un sous-ensemble significatif de moments suffisants pour une application bien spécifique. Dans ce qui suit, nous allons présenter l'ensemble de moments invariants de Hu [25] et l'ensemble de moments invariants de Alt[2]. Par la suite, nous allons étudier l'intérêt et les propriétés des moments d'inertie pour leur utilisation pour la représentation des signes.

3.4.1 Introduction des moments invariants

Avant de présenter les moments invariants, nous devons donner la définition de base des moments d'inertie ainsi que les notations que nous allons utiliser dans les prochaines sections. Nous allons utiliser dans ce chapitre $f(x, y)$ pour dénoter l'image continue et $g(x, y)$ pour dénoter l'image discrète. Le moment cartésien bi-dimentionnel d'ordre $(p+q)$ d'une fonction continue $f(x, y)$ s'écrit :

$$m_{pq} = \int_{-\infty}^{+\infty} x^p y^q f(x, y) dx dy \qquad pour \quad p, q = 0, 1, 2, ...$$

Le moment bi-dimentionnel d'une image discrète $g(x, y)$ est défini par :

$$m_{pq} = \sum_x \sum_y x^p y^q g(x, y)$$

Les propriétés d'invariance des moments ont reçu une attention considérable [22, 23]. Hu [25] en 1961, a été l'un des premiers à introduire les moments invariants. Il a pu obte-

nir un ensemble de moments invariants sous un changement d'échelle, de translation, de rotation et/ou de réflexion, basé sur une combinaison des moments réguliers en utilisant des invariants algébriques. Les moments invariants de Hu d'ordre ≤ 3 sont utilisés pour la première fois pour la reconnaissance des images de différents types d'avions [38], ces mêmes moments invariants sont aussi utilisés pour la reconnaissance des bateaux [6] et les moments d'ordre ≤ 2 sont aussi utilisés dans la reconnaissance de caractères. Il a été montré que l'utilisation des moments d'ordre ≤ 2 réduit l'erreur de classification [52]. Un survol des recherches récentes [34], révèlent les besoins d'une étude des propriétés des différents moments invariants quand ils sont utilisés dans les applications de reconnaissance des objets ou des formes en général. Pour la première fois, nous allons étudier leurs performances dans la reconnaissance des postures de l'alphabet du langage des signes en particulier. Dans ce qui suit, nous allons d'abord rappeler les moments invariants proposés par Hu [25] et ceux proposés par Alt [2] pour la reconnaissance des formes en général. Ensuite, nous étudierons les caractéristiques des moments d'inertie afin de les intégrer dans notre système de reconnaissance des postures de l'alphabet du langage des signes. Ces moments invariants sont décrits comme suit :

- les moments invariants de Hu :

  les moments centrés d'ordre (p, q) sont donnés par :

$$\mu_{pq} = \sum_x \sum_y (x - \overline{x})^p (y - \overline{y})^q g(x, y)$$

  où $\overline{x}$ et $\overline{y}$ représentent les coordonnées du centre de gravité de l'image. Il a été montré que l'ensemble des moments μ_{pq} sont invariants à la translation des objets dans l'image [2, 34]. Hu a ensuite défini les moments centrés normalisés comme suit :

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \qquad avec \quad \gamma = \frac{p+q}{2} + 1 \quad et \qquad p + q = 2, 3, ...$$

À partir de ces moments centraux normalisés, Hu a pu extraire un ensemble de caractéristiques qui sont invariantes à l'échelle, à la position et à l'orientation de l'objet [34]. En termes de moments, ces six caractéristiques sont données ci-dessous :

$$M_1 = \eta_{20} + \eta_{02}$$

$$M_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$M_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$M_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$M_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2]$$
$$+(3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

$$M_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

Hu a montré l'utilité des moments invariants à travers de simples expériences de reconnaissance de formes. Les moments invariants de Hu sont invariants à l'orientation des signes. Pour obtenir la non-invariance à la rotation, nous nous sommes intéressés dans ce qui suit, à une série de moments invariants qui a été proposée par Alt [2].

- Les moments invariants de Alt :

dans le but d'obtenir un ensemble de moments invariants par rapport aux changements d'échelle et à la translation, Alt [23] a proposé de normaliser les coordonnées (x, y) comme suit :

$$x' = \frac{(x - \overline{x})}{\sigma_x} \qquad y' = \frac{(y - \overline{y})}{\sigma_y}$$

où $\sigma_x = \sqrt{\frac{m_{20}}{m_{00}}}$ et $\sigma_y = \sqrt{\frac{m_{02}}{m_{00}}}$. Les valeurs moyennes des variables x' et y' sont nulles et leurs variances sont égales à 1. Pour forcer la non-invariance à la rotation,

Alt a proposé une autre normalisation des variables $x^{'}$ et $y^{'}$ par :

$$x^* = \frac{x^{'} - \rho y^{'}}{\sqrt{1-\rho^2}} \quad et \quad y^* = y^{'} \qquad \rho = \frac{\sum\sum x^{'} y^{'}}{\sum y^{'2}}$$

Finalement, il a proposé l'ensemble des moments suivants :

$$m_{pq} = \frac{1}{M_{00}} \sum_{x^*} \sum_{y^*} (x^*)^p (y^*)^q g(x,y)$$

La quantité ρ représente le coefficient de régression des variables $x^{'}$ et $y^{'}$. En particulier :

$$m_{00} = m_{20} = m_{02} = 1 \quad et \quad m_{10} = m_{01} = m_{11} = 0$$

Alt a montré que l'ensemble des moments m_{pq} décrits précédemment sont invariants à la translation et aux changements d'échelle et non invariants à la rotation, ce qui correspond aux critères fixés pour la représentation des postures du langage des signes. Cependant, pour représenter d'une façon unique les différentes postures, nous devons utiliser les moments au-delà de l'ordre 3, ceci diminue la performance du système. Un ensemble réduit et optimal de moments invariants de Alt n'est pas pertinent pour l'analyse du langage des signes.

3.4.2 Intérêt des moments dans la représentation des signes

Pour illustrer l'applicabilité des moments pour la représentation des postures de l'alphabet du langage des signes, nous allons associer à ces moments une signification en terme de caractéristiques des signes. Dans le cadre de notre étude, nous nous limiterons aux moments d'ordre < 4 puisque la signification des moments d'ordre supérieur ou égal à 4 n'est pas évidente. Parmi les caractéristiques induites par les moments d'ordre ≤ 4, nous pouvons citer la surface, le centre de gravité, la variance, la dissymétrie et finalement l'aplatissement.

Les moments d'ordre 0

La définition des moments d'ordre 0 $\{m_{00}\}$ de l'image discrète $g(x, y)$,

$$m_{00} = \sum_x \sum_y g(x, y) \quad = \quad M$$

représente la surface totale d'une image donnée mesurée en nombre de pixels. Il est clair que la surface des signes est invariante par rapport à la translation et non-invariante aux changements dans l'illumination mais, elle est aussi non-invariante à l échelle et à la rotation. Nous pouvons noter ainsi que la surface n'est pas une caractéristique pertinente puisqu'elle n'est pas suffisante pour distinguer entre tous les signes.

Les moments d'ordre 1

Les moments d'ordre 1 $\{m_{10}, m_{01}\}$ sont utilisés pour localiser le centre de gravité de la main. Le centre de gravité est invariant par rapport à l'échelle mais il est invariant à l'orientation des postures de la main. En terme de valeurs de moments, les coordonnées du centre de gravité sont déduites à partir des moments d'ordre 1 comme suit :

$$\overline{x} = \frac{m_{10}}{m_{00}} \qquad et \qquad \overline{y} = \frac{m_{01}}{m_{00}}$$

L'invariance de cette caractéristique par rapport aux changements des conditions d'illumination dépend de la définition de la fonction de l'illumination et du bruit, lorsqu'elle est calculée à partir des contours des signes. Cependant, si nous calculons le centre de gravité à partir des images de niveaux de gris, les conditions d'illumination de la scène deviennent un facteur important pour le calcul du centre de gravité. Nous ne devrons pas nous limiter à cette caractéristique car elle n'est pas pertinente.

Les moments d'ordre 2

Les moments d'ordre 2 $\{m_{02}, m_{11}, m_{20}\}$ représentent les moments d'inertie, ils peuvent être utilisés pour déterminer les axes principaux de l'allongement minimal et maximal de l'objet (voir fig. 3.6).

Les axes principaux d'inertie des postures

Les moments de second ordre peuvent être utilisés pour déterminer les axes principaux d'inertie des postures de l'alphabet du langage des signes. Les axes principaux majeur et mineur peuvent être décrits comme une paire d'axes associés aux moments de second ordre minimal et maximal respectivement (voir fig. 3.6).

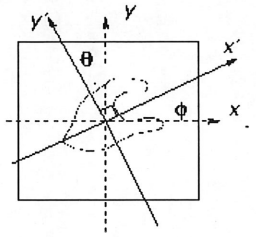


Figure 3.6: *L'estimation de l'orientation du signe «C».*

En terme de moments, l'orientation de l'un des axes principaux est donnée par l'équation suivante :

$$\phi = \frac{1}{2}\tan^{-1}\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}\right) \tag{3.1}$$

Notons que l'angle ϕ est l'angle de l'axe principal mineur le plus proche de l'axe des x et il est compris dans l'interval $[\frac{-\pi}{4}, \frac{\pi}{4}]$. L'orientation de l'axe principal majeur peut être déduite à partir de l'angle ϕ, et μ et $\mu_{20} - \mu_{02}$. Le tableau 4.2 illustre comment déterminer l'angle d'orientation de l'axe majeur principal θ en fonction des moments d'ordre 2 et l'angle ϕ. Cette orientation n'a du sens que si la région ne présente pas de symétrie de révolution, c'est-à-dire $\mu_{20} \neq \mu_{02}$. Les axes principaux peuvent être utilisés comme axes de référence uniques pour décrire l'orientation de postures de l'alphabet du langage des signes selon la position de la main dans le plan de l'image (rotation dans le plan de l'image). Les axes principaux répondent aux critères cités précédemment plus particulièrement, la non-invariance par rapport à l'orientation des postures de la main,

d'autant plus que le calcul des axes principaux ne nécessite pas le calcul des moments au-delà de l'ordre 2. Les axes principaux ne sont sensibles aux changements dans les conditions d'illumination que lorsqu'il s'agit des images de niveaux de gris. Cependant, l'obtention des axes principaux à partir des contours des postures reste relativement invariants par rapport à l'illumination de la scène. Les axes principaux d'inertie sont évidemment invariants à l'échelle et à la translation des objets dans le plan de l'image mais ils sont sensibles à l'orientation de l'objet dans le plan de l'image. Dans le cas des postures de l'alphabet du langage des signes, l'orientation des doigts et de la paume est un paramètre important pour distinguer plusieures postures. Cependant, les axes principaux ne sont pas pertinents puisqu'ils ne peuvent pas extraire les informations liées à la forme des signes et ils ne sont pas invariants aux nombres de doigts et aux nombres de vallées qui apparaîssent dans l'exécution des signes.

Nous détaillerons les solutions apportées à ces problèmes dans le chapitre 4, quand ceux-ci sont intégrés dans un nouveau modèle de représentation basé sur les fonctions de taille.

Les moments d'ordre 3

Les deux moments centrés d'ordre 3 $\{\mu_{30}, \mu_{03}\}$ décrivent la dissymétrie des projections de l'image sur les axes x et y. La dissymétrie est une mesure statistique du degré de distribution de la déviation de la symétrie autour du centre de gravité [34]. Les coefficients de la dissymétrie pour les projections de l'image sur les axes x et y sont donnés par :

$$DIS_x = \frac{\mu_{30}}{\sqrt{\mu_{20}^3}} \qquad DIS_y = \frac{\mu_{30}}{\sqrt{\mu_{20}^3}}$$

Comme pour les propriétés des moments d'ordre 2, la dissymétrie des formes est forcément invariante à l'échelle et à la translation. Cependant, elle n'est pas invariante à la rotation, puisqu'en tournant de 90^o un objet assymétrique par rapport à l'axe des x et symétrique par rapport à l'axe des y, produira un objet symétrique par rapport à l'axe des x et assymétrique par rapport à l'axe des y. Les signes des coefficients est une indication sur

quel coté de l'axe de projection que l'objet est dissymétrique. Dans notre cas, il n'est pas nécessaire de connaître la direction des axes principaux d'inertie comme nous pourrons le voir dans le chapitre 4.

Les moments d'ordre 4

Les deux moments centrés d'ordre 4 $\{\mu_{40}, \mu_{04}\}$ décrivent la platitude des projections de l'image sur les axes x et y. Le coefficient de l'aplatissement pour les projections de l'image sur les axes x et y est donné par :

$$PLT_x = \frac{\mu_{40}}{\mu_{20}^2} - 3 \qquad PLT_y = \frac{\mu_{04}}{\mu_{02}^2} - 3$$

Les valeurs négatives indiquent une distribution plate de pixels tandis que les valeurs positives indiquent une distribution de pixels plus aiguëe. Leurs invariances par rapport aux changements dans l'échelle et à la translation, sont justifiées même par leur définition, mais ne sont pas invariants à la rotation et restent toujours moins pertinents que les moments d'ordre 2 puisque deux formes identiques avec deux orientations différentes peuvent produire deux représentations similaires.

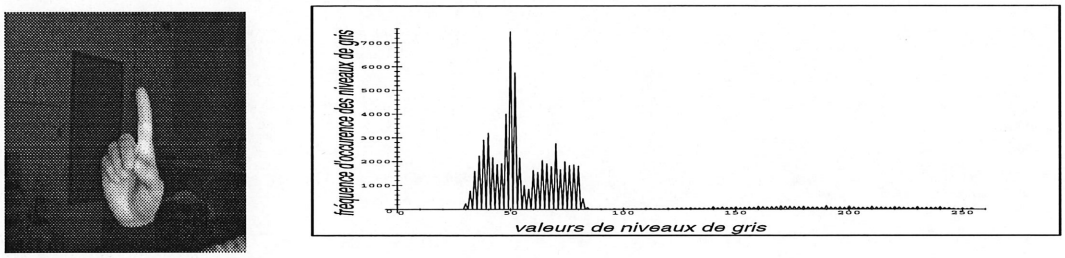
L'inconvénient majeur lié à l'utilisation des moments d'ordre ≥ 4 est le temps de calcul. Récemment, il a été démontré, dans une expérience d'identification des avions militaires [35] avec des images bruitées, que les moments réguliers et les moments invariants de Hu sont plus performants que les descripteurs de Fourier.

Les moments peuvent être utilisés soit pour l'image originale à niveaux de gris ou sur l'image binaire [1]. Pour des fins de pertinence et de complexité de calcul, nous avons opté pour l'utilisation des propriétés des moments réguliers sur les images binaires (après détection de contours) des postures. La performance et le coût d'utilisation des moments dans un système de reconnaissance des formes sont proportionnels à l'ordre des moments et du type d'invariant. Il serait préférable d'utiliser les moments d'ordre moins élevés.

3.5 L'histogramme d'orientation

3.5.1 Les histogrammes des niveaux de gris

Un histogramme d'une image à niveaux de gris est une fonction qui donne la fréquence d'occurence de chaque niveau de gris. En d'autres termes, la valeur de l'histogramme au point p dénotée par $h(p)$ est le nombre de pixels dans l'image avec les niveaux de gris égal à p. L'histogramme résume mieux l'information fréquentielle que contient l'image et



(a) *(b)*

Figure 3.7: *La représentation graphique de l'histogramme de niveaux de gris : (a) Le signe «1» et (b) l'histogramme de valeurs.*

sa structure statistique [5]. Par exemple, l'histogramme de la figure (fig. 3.7) informe sur le contraste de l'image et permet de calculer les caractéristiques de l'image comme les régions et les contours [10, 22]. Les autres caractéristiques pouvant être calculées à partir des histogrammes des niveaux de gris, sont indiquées ci-après :

- dynamique : la différence entre la plus grande valeur et la plus petite valeur des niveaux de gris.

- Médiane : valeurs du niveau de gris séparant en deux parties égales la surface de l'histogramme.

- Moyenne : la somme des valeurs des pixels de l'image divisée par le nombre de pixels de l'image.

- Variance : cette grandeur caractérise la dispersion des niveaux de gris autour de la valeur moyenne. La variance se calcule en effectuant la somme des carrés de la différence entre chaque valeur et la moyenne, divisée par le nombre de pixels de l'image. L'écart type est la racine carrée de la variance.

Les propriétés de cette représentation

Dans ce qui suit, nous allons citer quelques propriétés des histogrammes des niveaux de gris relativement aux critères cités au début de ce chapitre :

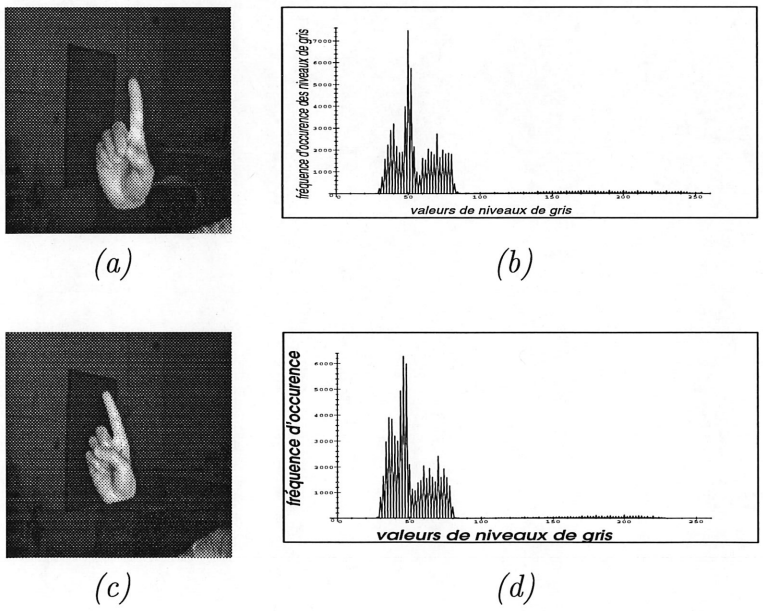
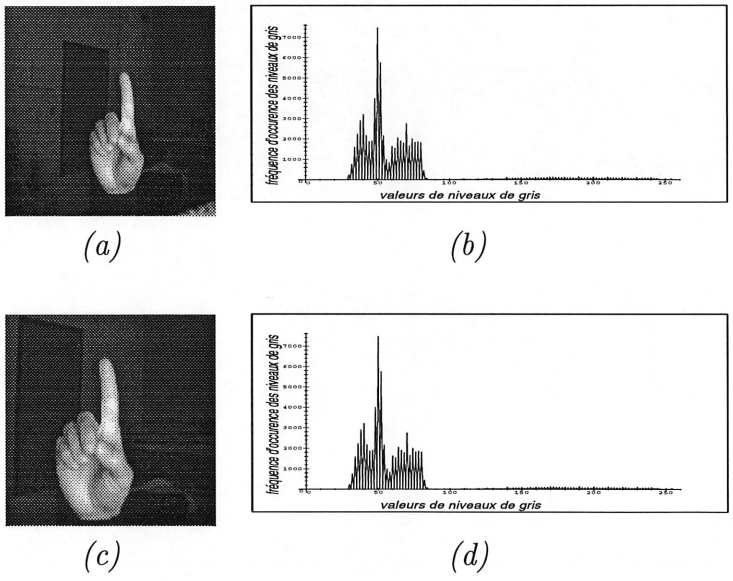


Figure 3.8: *L'invariance à la translation et à la rotation : (a) le signe «1», (c) le signe «1» après une translation et une rotation, (b) et (d) représentent les histogrammes de valeurs correspondants aux images (a) et (b) respectivement.*

- rapidité de calcul : supposons que nous nous contentons tout simplement d'utiliser l'histogramme des niveaux de gris de l'image comme vecteur de caractéristiques. L'avantage le plus important ici est la simplicité et la rapidité de calcul de l'histogramme de niveaux de gris relativement aux autres caractéristiques citées précédemment.

- Invariance par rapport à la translation et à la rotation : une autre propriété des histogrammes des niveaux de gris est celle d'être invariants à la translation et à la rotation de la gesture dans la scène. Ceci est dû principalement au fait que la fréquence d'occurence des niveaux de gris d'une même gesture, sous différentes positions ou sous différentes orientations dans la scène, reste la même. Nous devons noter que dans la pratique, l'influence d'une légère différence dans les histogrammes de la figure 3.8 est faible.



(a) *(b)*

(c) *(d)*

Figure 3.9: *La non-invariance à l'illumination et à l'échelle : (a) le signe «1», (c) le signe «1» avec une variation dans l'illumination et dans l'échelle, (b) l'histogramme des niveaux de gris du signe «1» et (d) l'histogramme des niveaux de gris de l'image (c).*

- Invariance par rapport aux conditions d'illumination et à l'échelle : l'efficacité des histogrammes des niveaux de gris est illustrée par la figure 3.9. Les figures 3.9(a) et 3.9(c) montrent la même gesture de la main sous différentes conditions d'illumination. Capter ou recenser seulement l'intensité de pixels, peut être sensible aux changements dans l'illumination de la scène. La différence entre les deux histogrammes des figures 3.9(b) et 3.9(d), représentant les deux images du même signe «1» sous différentes illuminations, est significative. Pour obtenir l'invariance par rapport aux conditions d'illumination et la non-invariance à la rotation, Bichsel [9] a introduit la mesure de la direction locale de l'orientation en chaque pixel de l'image. Finalement, W. T. Freeman [12] a utilisé cette technique pour le calcul des histogrammes des orientations locales.

3.5.2 Les histogrammes d'orientation

Plusieurs chercheurs ont utilisé l'histogramme d'orientation des images à niveaux de gris pour l'analyse des images. W.T.Freeman [12] a utilisé cette technique pour la reconnaissance d'une dizaine de gestures prédéfinies pour le contrôle d'une scène dans les jeux interactifs. Gokrani et Picard [18] ont utilisé les histogrammes d'orientation pour calculer les orientations dominantes de la texture. Un histogramme des orientations locales peut être utilisé comme vecteur de caractéristiques des postures de la main [12] pour les raisons suivantes :

- la considération et l'analyse des orientations nous permet d'obtenir l'invariance par rapport aux changements des conditions d'illumination.

- Le recensement des orientations locales à l'aide d'un histogramme permet d'obtenir une représentation invariante à la translation et par rapport à l'échelle.

– L'histogramme des orientations locales est non-invariant à la rotation.

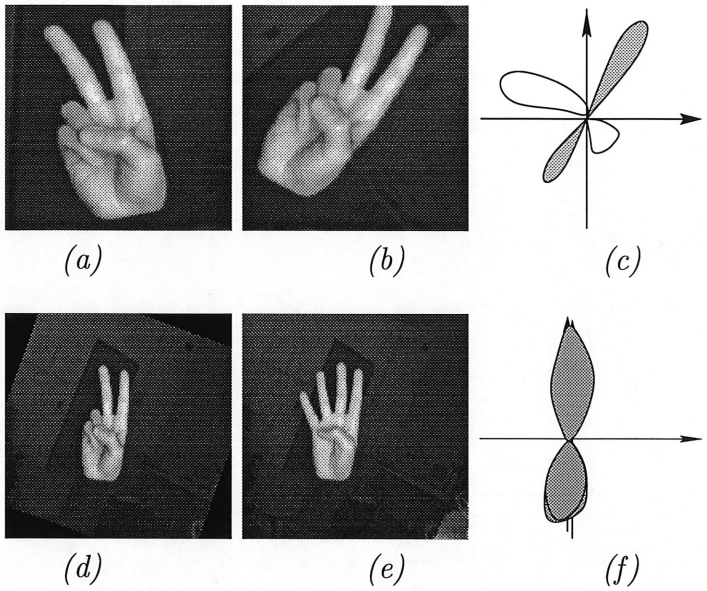


Figure 3.10: *Les histogrammes d'orientation : (a) et (b) représentent les images du signe «2», orientées vers la gauche et vers la droite respectivement, (d) et (e) représentent les images des signes «2» et «W» respectivement et (c) et (f) représentent la différence entre les histogrammes d'orientation des images [(a) et (b)] et [(d) et (e)] respectivement.*

Rappelons que le nombre de doigts perçu ou visualisé dans l'alphabet du langage des signes est un paramètre pertinent et discriminant entre plusieurs signes. Comme le montre la figure 3.10, les histogrammes d'orientation ne permettent pas de distinguer les signes de même orientation avec un nombre de doigts différent. Les figures 3.10(a) et 3.10(b) montrent deux images avec différentes orientations ayant le même nombre de doigts. Dans ce cas, les histogrammes d'orientation associés aux gestures sont différents (fig. 3.10(c)). Dans le cas des figures 3.10(d) et 3.10(e), les deux gestures sont différentes, leurs histogrammes d'orientation sont presque identiques (fig. 3.10(f)). Nous concluons que l'histogramme d'orientation peut être considéré comme un vecteur de caractéristiques invariant par rapport à la taille, à la translation et aux changements d'illumination des

gestures dans l'image. Il est non-invariant par rapport à l'orientation des gestures. L'histogramme d'orientation est aussi relativement insensible par rapport aux nombres de doigts utilisés pour effectuer la gesture. Ce dernier est considéré comme un paramètre pertinent dans l'analyse de l'alphabet du langage des signes.

3.6 Conclusion

Dans ce chapitre, nous avons présenté un étude d'évaluation détaillée des différents types de caractéristiques géométriques se rapportant aux contours et aux régions. Cette évaluation ainsi que les résultats expérimentaux obtenus des différents types de caractéristiques vont nous permettre de choisir un corpus approprié afin de tester un nouveau modèle de représentation qui sera décrit dans le chapitre 5. Les principaux résultats de cette évaluation selon plusieurs critères sont résumés dans le tableau 3.1. Nous pouvons constater qu'aucune des caractéristiques ne répondent aux critères de selection fixés au début de ce chapitre. Lors de cette étude d'évaluation, nous avons tenté de trouver un type de

Tableau 3.1: *Le tableau récapitulatif d'évaluation des caractéristiques.*

Type de caractéristiques	Type de primitive	Invariance à			Pertinence	Invariance aux doigts
		l'éch.	la trans.	la rot.		
Chaînes Numériques	Contour	Non	Oui	Non	Non	Non
Chaînes Convexeté/Concavité	Contour	Oui	Oui	Oui	Non	Oui
Desc. Fourier	Contour	Non	Non	Non	Non	Non
Moments de Hu	Cont/Rég	Oui	Oui	Oui	Non	Non
Moments de Alt	Cont/Rég	Oui	Oui	Non	Non	Non
Axes principaux	Cont/Rég	Oui	Oui	Non	Non	Non
Hist. de niveaux de gris	Région	Non	Oui	Oui	Non	Non
Hist. d'orientation	Cont/Rég	Oui	Oui	Non	Non	Non

caractéristiques qui respecte les critères fixés au début de ce chapitre même si celles-ci

ne sont pas parfaites. Nous avons choisi les moments d'inertie qui serviront de type de caractéristiques de base dans la construction d'un nouveau modèle de représentation autour d'un outil mathématique appelé *fonctions de taille.* Avant de décrire ce nouveau modèle de représentation, nous étudierons d'abord dans le chapitre suivant, la notion et les propriétés des fonctions de taille et ensuite nous proposons un algorithme pour le calcul des fonctions de taille dans le cas discret.

CHAPITRE 4

Représentation des signes

Dans le chapitre 3, nous avons présenté plusieures caractéristiques utilisées dans les approches traditionnelles. Ces caractéristiques paraîssent être convenables pour les objets rigides pour lesquels le modèle mathématique est facile à obtenir, mais elles ne répondent pas aux critères cités au début du chapitre 3. De plus, ces caractéristiques ne sont pas appropriées pour les objets naturels (comme dans le cas du langage des signes) pour lesquels nous devons tenir compte d'autres facteurs non seulement métriques (comme par exemple la distance en chaque point à partir du centre de gravité), mais aussi topologiques pour pouvoir extraire le comportement des mesures effectuées sur la forme.

Le but de ce chapitre est de présenter les fonctions de taille dans un niveau de détails suffisant pour pouvoir les utiliser dans notre modèle de représentation. Ensuite, nous allons démontrer le potentiel de ces fonctions sous les différents aspects de la représentation des formes. Pour faire suite, nous allons identifier les limites de cette théorie et les problèmes soulevés lors des applications déjà effectuées pour la reconnaissance du langage des signes. Finalement, nous prélèverons les principales issues afin de proposer un nouveau schéma de représentation basé sur une nouvelle classe de fonctions de taille dictée par le besoin de répondre aux spécifications citées dans le chapitre 3. Avant de proposer un algorithme de calcul des fonctions de taille, nous allons d'abord présenter dans ce qui

suit, les fonctions de mesure et les fonctions de taille ainsi que leurs propriétés.

4.1 Présentation des fonctions de taille

Dans une série d'articles mathématiques [14, 13, 45], l'analyse des formes par des fonctions à valeurs entières appelées *fonctions de taille*, a été proposée. Contrairement aux approches traditionnelles pour la représentation et la reconnaissance des formes, les fonctions de taille décodent de l'information des deux propriétés métriques et topologiques des objets.

Le potentiel de cette représentation des formes par les fonctions de taille dans le domaine de la vision artificielle, est décrit à travers plusieurs exemples d'application dont on peut citer [48, 47]. Ce sont des concepts modulaires dans le sens ou elles dépendent d'une certaine fonction, appelée *fonction de mesure*, qui sera choisie dans le but d'obtenir les propriétés souhaitables des objets de la scène à analyser [49, 50].

4.1.1 Introduction intuitive des fonctions de taille

Avant de donner une description détaillée de cette représentation à l'aide des fonctions de taille, nous discuterons des idées clées de cette représentation à l'aide d'un exemple simple. Comme nous l'avons déjà expliqué précédemment, le but principal des fonctions de taille est celui de représenter l'aspect métrique et l'aspect topologique du comportement de la fonction de mesure φ [46, 49]. Ceci s'explique par le fait que cette représentation est basée sur deux fonctions :

- *la fonction de mesure :* c'est une fonction à valeurs réelles définie sur la forme.

- *La fonction de taille :* c'est une fonction de deux variables réelles à valeurs entières, résultantes de la fonction de mesure effectuée sur la forme.

Pour une meilleure compréhension, nous allons expliquer les fonctions de taille et de mesure à travers un exemple.

Soit φ la distance entre un point de la courbe planaire γ et un point a (fig. 4.1).

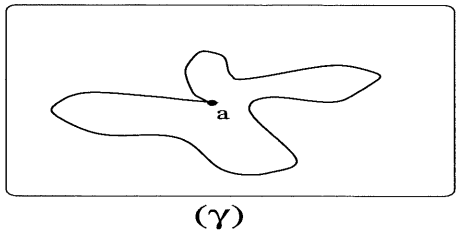


Figure 4.1: *Une courbe planaire quelconque* γ.

Le graphe de $\varphi = \varphi(t)$ représente une paramétrisation particulière de la courbe γ et est montré dans la figure 4.2(a). Puisque la courbe est fermée, le premier et le dernier point du graphe coïncident. La fonction φ est un exemple de *fonction de mesure*, ce qui explique l'aspect métrique des fonctions de taille. Cette représentation ne spécifie pas la mesure la plus souhaitable, puisque cela dépend de l'application. Dans les applications de vision artificielle, les mesures de représentation des formes sont généralement calculées sur les contours de l'image.

L'aspect topologique s'explique par le calcul de la fonction de taille $l_\varphi = l_\varphi(x, y)$ avec l'introduction de deux paramètres réels x, y tels que $x \leq y$. Les deux paramètres sont utilisés pour identifier les primitives qui se situent à une distance entre x et y du point a. Par exemple, les régions ombrées des figures 4.2(b) et 4.2(c) identifient les parties du graphe avec $\varphi \leq x$ et $\varphi \leq y$ respectivement. La superposition des figures 4.2(b) et 4.2(c) est dans la figure 4.2(d). Les régions sombres de la figure 4.2(d) correspondent à l'intersection entre les parties du graphe avec $\varphi \leq x$ et $\varphi \leq y$. La fonction de taille l_φ (fig. 4.2(d)) est définie par le nombre de régions en gris clair qui ont un segment en commun avec au moins une région en gris foncé. À partir de ces définitions, nous pouvons

déduire que dans l'exemple de la figure 4.2(d), $l_\varphi = 2$ puisqu'il n'existe que deux régions marquées en gris clair générées par l_φ. La figure 4.2(e) montre la même construction que la figure 4.2(d) pour un choix différent de (x, y).

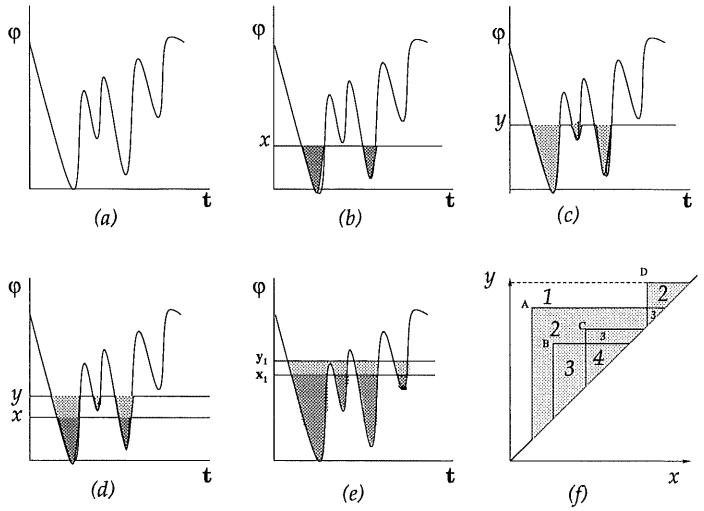


Figure 4.2: *Le calcul des fonctions de taille : (a) représente une paramétrisation particulière $\varphi = \varphi(t)$ de la courbe γ, (b) et (c) identifient les parties du graphe avec $\varphi \leq x$ et $\varphi \leq y$ respectivement, (d) représente l'intersection des graphes (b) et (c), (e) la même construction que (d) avec $x = x_1$ et $y = y_1$ et (f) représente le graphe de la fonction de taille l_φ.*

Pour chaque valeur (x, y) dans le plan réel avec $x \leq y$ une valeur entière $l_\varphi(x, y)$ est obtenue. Le graphe de l_φ est donné dans la figure 4.2(f).

Avant d'étudier les propriétés des fonctions de taille, nous allons donner dans ce qui suit, les définitions mathématiques de la fonction de mesure et de la fonction de taille.

4.1.2 La fonction de mesure

Nous définissons un objet comme une courbe fermée et continue dans le plan γ sans contours internes. Les informations métriques de la forme de la courbe γ sont décodées par la fonction de mesure à valeurs réelles définie sur γ.

Définition 1 : soit γ une courbe planaire continue du plan réel $\mathbb{R}^2$. Toute fonction de mesure φ est définie par : $\varphi : \gamma \longmapsto \mathbb{R}$

L'analyse de la forme de la courbe γ basée sur la fonction de mesure φ est maintenant possible. En principe, toute fonction réelle continue définie sur la courbe γ peut servir de fonction de mesure. La distance entre deux points, la coubure, les ordonnées des points ainsi que les abcisses des points, sont toutes des fonctions de mesure admissibles par rapport à un système de références bien défini.

4.1.3 La fonction de taille

Soient p et q deux points de la courbe γ (fig. 4.1) . Puisque γ est une courbe fermée, il existe toujours un chemin continu qui joint les points p et q sans se détacher de γ.

Soit x un nombre réel et $\gamma(\varphi \leq x)$ l'ensemble de points de γ avec $\varphi \leq x$. Puisque l'ensemble $\gamma(\varphi \leq x)$ peut être plus qu'une composante connexe, alors l'existance d'un chemin continu qui joint p à q sans se détacher de la courbe $\gamma(\varphi \leq x)$ dépend de la valeur spécifique de x.

Le second nombre réel y peut être utilisé pour établir la relation d'équivalence R_y entre une paire de points de γ. Deux points p et q peuvent être R_y-équivalents s'il existe un chemin continu entre p et q qui soit entièrement à l'intérieur de la courbe $\gamma(\varphi \leq y)$ ou si $p = q$. Dans l'exemple de la figure 4.2(e), nous avons $x = x_1$ et $y = y_1$. Les points de γ dont $\varphi \leq x_1$ et $\varphi \leq y_1$ sont représentés respectivement par les régions en gris foncé et en gris clair. La variable $x = x_1$ détermine quatre composantes connexes distinctes de la courbe γ avec $\varphi \leq x_1$. Deux composantes parmi les quatre peuvent être reliées à travers de $x_1 < \varphi \leq y_1$, donc, elles peuvent être identifées étant une, d'où la valeur de la fonction de taille $l_\varphi(x_1, y_1) = 3$.

Nous pouvons maintenant produire une représentation de la courbe γ, appelée *fonction de taille* qui est effectivement, une fonction de deux variables réelles x et y à valeurs

entières dont la variable x identifie l'ensemble $\gamma(\varphi \leq x)$ et la variable y détermine si deux points de la courbe γ sont R_y-équivalent. Si X est un ensemble de points de γ où la fonction de mesure $\varphi \leq x$ et R_y une relation d'équivalence définie sur l'ensemble X, alors $\aleph(X/R_y)$ dénote le nombre de classes d'équivalence dont l'ensemble X est divisé par la relation d'équivalence R_y. Par conséquent, la fonction de taille peut être définie comme suit [44] :

Définition 2 : pour chaque paire de points $(x, y) \in \mathbb{R}^2$, la fonction de taille l_φ est définie par :

$$l_\varphi(x, y) = \aleph(\gamma(\varphi \leq x)/R_y)$$

Puisque le nombre d'éléments de l'ensemble quotient $\gamma(\varphi \leq x)/R_y$ (ie : le nombre de composantes connexes) peut être infini (s'il s'agit par exemple d'une fonction de mesure $\varphi(t) = \sin(t)$), la fonction de taille doit être considérée comme une fonction définie par :

$$\mathbb{R}^2 \longmapsto \mathbb{N} \cup \{+\infty\} \qquad \text{avec} \qquad l_\varphi(x, y) = +\infty \quad \text{si} \quad \aleph(\gamma(\varphi \leq x)/R_y) = +\infty$$

Dans ce qui suit, nous allons étudier les propriétés des fonctions de taille dans les cas continu et discret afin de proposer un algorithme de calcul. Finalement, nous étudierons leurs propriétés d'invariance.

4.2 Les propriétés des fonctions de taille

Les fonctions de taille ont de nombreuses propriétés intéressantes pour la représentation des formes. Nous nous limiterons ici aux propriétés principales suivantes :

- **finitude :** le résultat fondamental de la théorie des fonctions de taille montre que les valeurs des fonctions de taille l_φ sont toujours finies et strictement positives sur les points de la courbe qui s'allongent sur une région triangulaire T_φ de l'aire définie

par :

$$T_\varphi = \{(x,y): \quad \varphi^{min} \leq y \leq \varphi^{max} \quad , \quad \varphi^{min} \leq x < \quad y\}$$

où φ^{min} et φ^{max} représentent le minimum et le maximum de la fonction de mesure φ à travers la courbe γ. Le graphe de la fonction de taille est toujours défini dans une région triangulaire avec $\varphi^{min} \leq x$ et $y \leq \varphi^{max}$ (l'aire T_φ consiste en des points de la région triangulaire avec $x < y$). De plus, la normalisation $\bar{\varphi} = \dfrac{\varphi - \varphi^{min}}{\varphi^{max} - \varphi^{min}}$ montre que la fonction de taille calculée dans la région triangulaire définie par $T_\varphi = \{(x,y) : 0 \leq x < y \leq 1\}$ est invariante à l'échelle.

- **Monotonicité et continuité :** pour toute courbe γ et pour toutes fonctions de mesure φ admissibles, la fonction de taille $l_\varphi(x,y)$ est non décroissante en x (pour un y fixe) et non croissante en y (pour un x fixe). Autrement-dit, si la fonction de taille prend la même valeur en deux points qui s'allongent sur l'axe vertical ou horizontal, alors elle prend les mêmes valeurs à travers le segment qui joint les deux points.

- **Égalités et inégalités :** les deux inégalités fondamentales de cette théorie $\varphi \leq x$ et $\varphi > y$ sont basées sur deux paramètres indépendants x et y. La courbe dans la figure 4.3(b) est obtenue par l'ajout d'un signal sinusiodal à la courbe de la figure 4.3(a). Les figures 4.3(c) et 4.3(d) montrent le graphe de la fonction de mesure φ, la distance définie entre le point a et tout autre point associé aux courbes des figures 4.3(a) et 4.3(b) respectivement. Les fonctions de taille correspondantes aux courbes 4.3(a) et 4.3(b) sont reproduites dans les figures 4.3(e) et 4.3(f) respectivement.

  À partir de la figure 4.3(f), nous pouvons constater que le composant fréquentiel de la courbe 4.3(b) (et de la fonction de mesure de la figure 4.3(d)) génère des grandes valeurs de la fonction de taille dans le voisinage de la droite $y = x$, mais virtuellement n'a pas d'effets dans la région dont y est suffisamment large que x.

En examinant les graphes des figures 4.3(e) et 4.3(f) seulement à proximité de la droite diagonale $y = x$, la similarité entre les deux courbes des figures 4.3(a) et 4.3(b) ne peut être affirmée. Cependant, si nous considérons toute la région triangulaire $y > x$, la similarité des courbes des figures 4.3(a) et 4.3(b) peut être affirmée.

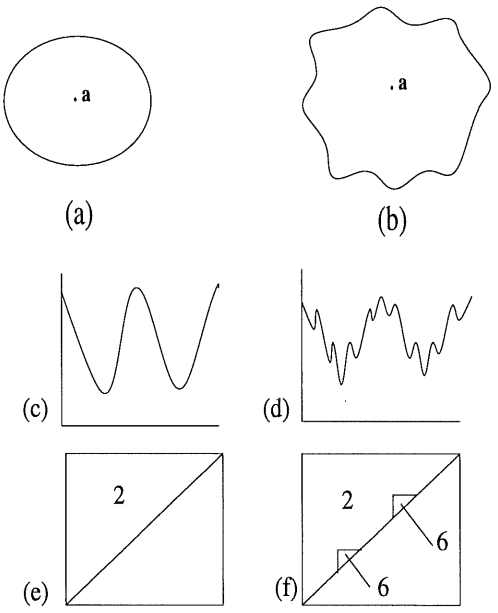


Figure 4.3: *Les propriétés d'inégalité et d'égalité: (a) et (b) représentent une sphère et une sphère bruitée, (c) et (d) représentent la paramétrisation des courbes (a) et (b) respectivement, et (e) et (f) représentent les fonctions de taille correspondantes aux courbes (a) et (b) respectivement.*

Nous déduisons que les fonctions de taille sont souhaitables pour la reconnaissance des objets similaires mais pas nécessairement identiques. Nous pouvons aussi nous intéresser qu'à certaines zones des régions triangulaires qui représentent les fonctions de taille pour effectuer la comparaison des formes selon les besoins de l'application.

- **Héritage :** une des propriétés fondamentales de la représentation proposée dans ce projet est celle de l'héritage des propriétés d'invariance par rapport aux transformations géométriques des fonctions de mesure sur lesquelles est basée la fonction de taille. En général, les propriétés d'invariance souhaitées dans une application spécifique comme dans la reconnaissance du langage des signes, peuvent être utilisées comme critères de base dans la recherche de la fonction de mesure appropriée. Nous allons présenter les propriétés d'invariance plus en détail dans la section 4.4.

4.3 L'algorithme de calcul des fonctions de taille

Une approche formelle de calcul des fonctions de taille consiste à discrétiser la forme de l'objet. Si nous considérons la discrétisation de la lettre «M» tracée sur le plan réel, alors, la lettre «M» peut être vue comme un graphe (fig. 4.4). Nous pouvons maintenant dresser un algorithme pour le calcul de la fonction de taille d'une courbe dans le plan image. Pour des raisons de simplicité, nous allons illustrer l'algorithme de calcul de la fonction de taille dans le cas où la fonction de mesure φ est définie sur un ensemble de points de la courbe γ avec $\varphi \leq 0$ qui représente la discrétisation de la lettre «M».

Pour ce faire, nous choisissons la fonction de mesure φ qui associe à chaque point du graphe correspondant à la lettre «M», un nombre réel. Cette fonction de mesure est définie comme suit :

$$\varphi(p) = \frac{y(p) - \varphi^{min}}{\varphi^{max} - \varphi^{min}}$$

où p est un point du graphe de la lettre «M», $y(p)$ son ordonnée, φ^{min} et φ^{max} sont respectivement le minimum et le maximum de $\varphi(p)$. Dans la figure 4.4(a), $\varphi^{min} = 0$ et $\varphi^{max} = 1$.

Soit $B(p^i)_\epsilon$ une boule de centre p^i et de rayon ϵ et $\bar{l}_\varphi$ la fonction de taille dans le cas

discret, alors l'algorithme de calcul de la fonction de taille est composé de quatres étapes :

- **Étape 1 :**

  discrétiser la courbe γ avec un nombre fini N de points $p^i, i = 1, ..., N$ tel que $\gamma \subset \cup_{i=1}^{N} B(p^i)_\epsilon$ et l'ensemble $B(p^i)_\epsilon \cap \gamma$ est un ensemble non vide et connexe pour $i = 1, ..., N$ (fig. 4.4(a)).

- **Étape 2 :**

  - définir le graphe G dont les sommets sont les points p^i et les arcs joignent les points adjacents dans γ.
  - Calculer $\varphi(p^i)$ à chaque points p^i,$i = 1, ..., N$. (fig. 4.4(a)).

- **Étape 3 :**

  calculer le maximum φ^{max} de $\varphi(p^i), i = 1, ..., N$ et initialiser une marge de déplacement δ.

- **Étape 4 :**

  pour $y = 0$ jusqu'à φ^{max} par pas de δ faire :

  - définir le sous-graphe $G_{\varphi \leq y}$ du graphe G induit par l'ensemble des sommets de G pour lesquels $\varphi \leq y$ (comme par exemple dans la figure 4.4(b) où $y = 0.8$).
  - Pour $x = 0$ jusqu'à y par pas de δ faire :
    - calculer $\bar{l}(\alpha; x, y)$ le nombre de composantes connexes de $G_{\varphi \leq y}$ qui contiennent au moins un sommet p^i tel que $\varphi(p^i) \leq x$ (comme par exemple dans la figure 4.4(b) où pour $y = 0.8$ et $x = 0.5$, $\bar{l}_\varphi(0.5, 0.8) = 3$).

Les deux conditions de la première étape consistent à vérifier que la courbe γ est discrétisée de façon à ce que les cercles contiennent exactement un seul arc connecté à la fois sur la

courbe γ. Le graphe G dans la deuxième étape, représente la version discrète de la courbe γ. La troisième étape détermine la résolution minimale et les points (x, y) pour lesquels $\varphi(p^i)$ est calculée. Dans la quatrième étape, la fonction de taille est calculée pour l'ensemble des points de la région triangulaire $T_\varphi(\gamma) = \{(x, y) : 0 \leq y \leq \varphi^{max}, 0 \leq x < y\}$.

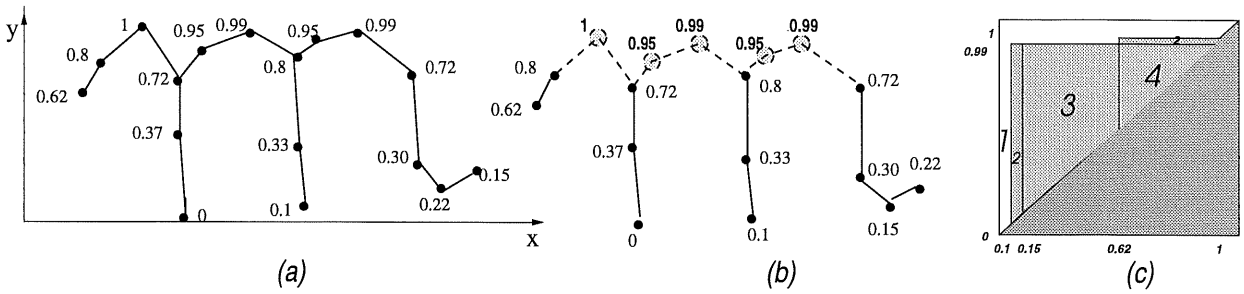


Figure 4.4: *La fonction de taille associée à la lettre «M» : (a) le graphe de la lettre «M» dont les sommets sont étiquetés par un nombre réel (résultats de la fonction de mesure φ), (b) les points ainsi que les composantes connectées dont la fonction de mesure φ dépasse strictement 0.8 sont représentés par des pointillés et le nombre de composantes connexes qui contiennent au moins un point étiqueté par une valeur qui ne dépasse pas 0.5 est égale à 3 (ie : $l_\varphi(0.5, 0.8) = 3$) et (c) la fonction de taille associée à la lettre «M» représentée dans la région triangulaire avec $x > y$.*

4.4 Propriétés d'invariances euclidiennes

Une des propriétés fondamentales des fonctions de taille présentée précédemment dans ce chapitre, est celle de l'héritage des propriétés d'invariance de la fonction de mesure. Par exemple, la fonction de mesure (la distance des points du contour à partir d'un point «a») présentée dans la première section de ce chapitre est invariante seulement à la rotation du plan de l'image autour du point «a». Cependant, cette fonction n'est pas invariante ni à la translation ni à l'échelle puisque le point «a» est fixe dans le plan. Dans le but d'obtenir l'invariance à une plus grande classe de transformations, la meilleure stratégie est celle de remplacer le point «a» par le centre de gravité du contour noté par «c» comme dans l'exemple de la figure 4.5(a). Par conséquent, la fonction de mesure D_c qui mesure

la distance des points du contour à partir du centre de gravité ainsi que les fonctions de taille induites l_{D_c} sont invariantes à l'échelle, à la rotation et à la translation à travers le plan de l'image. Donc, les propriétés d'invariances euclidiennes sont déjà obtenues. Ces propriétés d'invariances sont illustrées dans la figure 4.5.

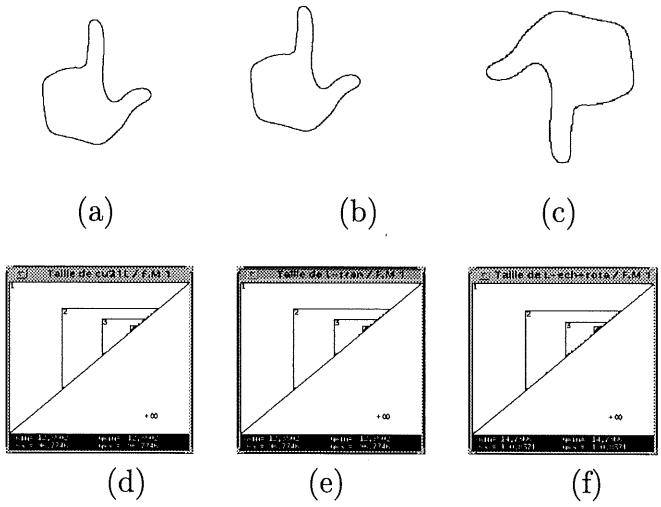


Figure 4.5: *Les propriétés d'invariances euclidiennes: (a) le signe «L», (b) déplacement du signe «L» dans l'image, (c) le signe «L» avec un agrandissement et une rotation de* 180^o*, (d), (e) et (f) représentent les fonctions de taille correspondantes aux contours (a), (b) et (c) respectivement.*

La figure 4.5 montre respectivement le contour du signe «L» (fig. 4.5(a)), le même contour après lui avoir appliqué différentes transformations euclidiennes, une translation (fig. 4.5(b)) et une rotation de 180^o avec un agrandissement (fig. 4.5(c)). Les figures 4.5(d), 4.5(e) et 4.5(f) montrent les fonctions de taille l_{D_c} correspondantes aux différentes transformations du signe «L» induites par la fonction de mesure D_c qui est la distance des points du contour à partir du centre de gravité. Il est clair que la différence entre les fonctions de taille des figures 4.5(d), (e) et (f) qui semble être due aux effets du plan discret est négligeable. Nous ne voulons pas d'une fonction de mesure invariante à la rotation comme nous l'avons déjà précisé dans le chapitre 3, puisque la représentation

du signe de la figure 4.5(a) doit être différente de celle du signe de la figure 4.5(c).

Nous allons maintenant décrire dans les sections suivantes, la construction d'une représentation pour la reconnaissance de l'alphabet du langage des signes basée sur les fonctions de taille. Pour ce faire, nous allons présenter dans un premier temps les différentes fonctions de mesure existantes et ensuite trouver une fonction de mesure admissible et adéquate pour le traitement des postures de l'alphabet du langage des signes.

4.5 Les fonctions de mesure existantes

Dans cette section, nous allons étudier plusieures fonctions de mesure déjà utilisées pour l'analyse de l'alphabet du langage des signes. Le choix de la fonction de mesure adéquate pour la représentation des postures de l'alphabet du langage des signes doit satisfaire les critères cités dans le chapitre 3. À partir des images de postures de l'alphabet du langage des signes des figures 3.2 et 3.3, nous constatons que le problème de déterminer une fonction de mesure capable de décoder les similarités et de distinguer entre différents signes est non trivial. La plus grande difficulté est due aux faits que les poses des mains ne sont pas fixes et le style de faire les signes change avec les individus. Dans ce qui suit, nous allons énumérer plusieures classes de fonctions de mesure qui ont été proposées par C. Uras et A. Verri [44, 48, 47, 49, 46] et nous allons soulever les problèmes de chacunes d'elles, pour finalement proposer une nouvelle fonction de mesure efficace basée sur les moments d'inertie d'ordre 2.

1. La distance à partir du centre de gravité :

la distance d'un point à partir du centre de gravité paraît plausible pour extraire les caractéristiques pertinentes des images de signes, telles que les doigts apparents dans chaque contour de signe. Les figures 4.6(a) et 4.6(d) montrent les images de contour des signes «K» et «V» respectivement, et les figures 4.6(b) et 4.6(e) montrent les fonctions de taille correspondantes. Ce n'est pas surprenant qu'à partir des deux postures qui

ont un même nombre de «doigts» de même taille (fig. 4.6(a)) et (fig. 4.6(d)), les deux représentations correspondantes (fig. 4.6(b)) et (fig. 4.6(e)) sont presque similaires. Cette similarité est due essentiellement aux propriétés d'invariance par rapport à la rotation de la fonction de mesure employée. Dans ce qui suit, nous allons étudier une autre fonction de mesure basée sur l'axe horizontal qui passe par le centre de gravité. Cette nouvelle fonction de mesure est capable de prendre en compte les orientations des doigts puisque nous considérons seulement les points qui existent au-dessus de l'axe horizontal qui passe par le centre de gravité et évidemment est non invariante à la rotation.

2. La distance à partir d'un axe de référence :

pour remédier au problème précédent, C.Uras [48] a donc proposé à ce sujet, $d(p)$ soit la distance d'un point de contour à partir de la ligne horizontale qui passe par le centre de gravité, alors la fonction de mesure φ_0 est définie par :

$$\varphi_0(p) = \begin{cases} d(p) & \text{si } p \quad \text{s'allonge au-dessus de l'axe horizontal} \\ 0 & \text{sinon} \end{cases}$$

Les fonctions de taille des signes «K» et «V» induites par la fonction de mesure φ_0 sont représentées respectivement dans les figures 4.6(c) et 4.6(f). Nous pouvons constater que l'utilisation de la fonction de taille à base de la fonction de mesure φ_0 distingue mieux les signes «K» et «V» puisque la fonction de taille l_{φ_0} dépend seulement de la portion des contours située au-dessus de l'axe horizontal. Cette fonction de taille est sensible à l'index dans le cas du signe «K» et à la fois à l'index et le majeur dans le cas du signe «V».

Après avoir fait plusieures analyses sur la structure de l'alphabet de la langue des signes dans le chapitre 1, il est clair que la fonction de mesure φ_0 est incapable de discriminer tous les signes de l'alphabet. Ceci est dû aux orientations relatives des doigts par rapport à l'axe horizontal comme dans le cas des signes «P» et «Q», «G» et «H» ou «M» et «N»

dans les figures 3.2 et 3.3.

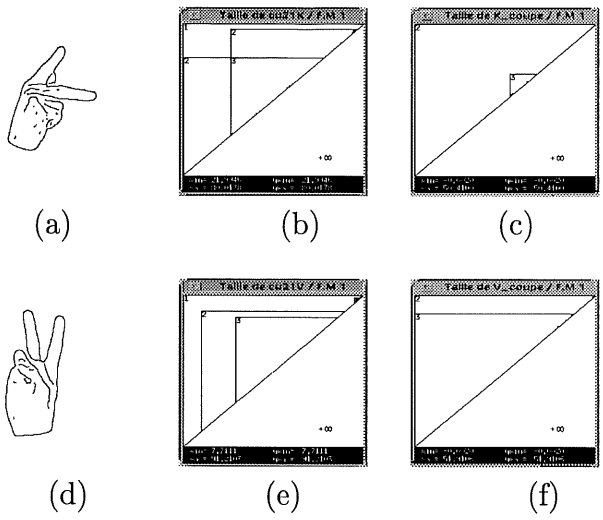


Figure 4.6: *La recherche d'une fonction de mesure adéquate: (a) et (d) les contours des signes «K» et «V», (b),(e) et (c),(f) les fonctions de taille correspondantes aux contours (a) et (b) avec φ = la distance des points à partir du centre de gravité et $\varphi = \varphi_0$ respectivement.*

3. La première famille de fonctions de mesure :

comme il a été mentionné précédemment, étant donnée une fonction de mesure spécifique φ, différentes formes peuvent produire la même fonction de taille. Ceci implique la nécessité de rechercher possiblement une famille de fonctions de mesure, chacune permettra la discrimination d'un sous-ensemble de signes. Nous pouvons considérer la fonction de mesure φ_0 définie précédemment comme un cas particulier (le cas ou $\theta = 0$) de la famille de fonctions de mesure φ_θ indexées par l'angle θ avec $0 \leq \theta \leq 360$.

4. La deuxième famille de fonctions de mesure :

une autre catégorie de fonctions de mesure utilisée par C.Uras et A.Verri [49, 46] consiste à encadrer les contours des signes extraits dans une boîte rectangulaire montrée dans la figure 4.7(a). Pour chaque point p du segment de droite horizontale qui passe par le

centre de la boîte rectangulaire, est associée une distance $h(p)$ qui est la distance entre le point p et le point de contour le plus loin se trouvant sur la droite verticale qui passe par le point p (fig. 4.7(b)).

Cette fonction de mesure est définie par :

$$\varphi_0(p) = \begin{cases} h(p) & \text{si l'intersection de la droite verticale et le contour n'est pas vide} \\ 0 & \text{sinon} \end{cases}$$

Le graphe de φ_0 pour le contour de la figure 4.7(b) est montré dans la figure 4.7(c). La fonction de taille l_{φ_0} induite par une telle fonction de mesure est montrée dans la figure 4.7(d). Il est clair que la fonction de taille l_{φ_0} couvre une certaine direction particulière du plan de l'image. Dans le but de produire des fonctions de taille capables de distinguer entre les différents signes, la fonction de mesure φ_0 peut être considérée comme un cas particulier dont ($\theta = 0$) d'une grande famille de fonctions de mesure φ_θ indexée par un angle θ avec ($0 \leq \theta \leq 360$).

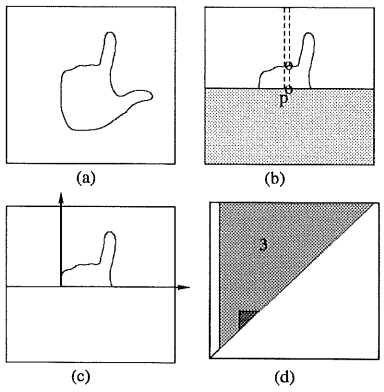


Figure 4.7: *La deuxième famille de fonctions de mesure : (a) le contour du signe «L» à l'intérieur d'une boîte rectangulaire, (b) la distance entre un point p et le point le plus loin qui se trouve sur la droite verticale qui passe par le centre de gravité, (c) le graphe de la fonction de mesure et (d) représente la fonction de taille correspondante.*

4.6 Limite des fonctions de mesure existantes

Avant de proposer notre fonction de mesure, nous allons présenter dans ce qui suit, les limites des fonctions de mesure proposées précédemment et les résultats expérimentaux obtenus en utilisant la troisième famille de fonctions de mesure.

La pertinence de ces familles de fonctions de mesure ainsi que le nombre considérable de rotations nécessaires pour le calcul des fonctions de taille induites reste un problème majeur de cette technique. Les expériences empiriques nous montrent l'impertinence de la deuxième famille de fonction de mesure φ_θ pour un signe particulier «P» (fig. 4.8). Le fait que la portion de contour de la figure 4.8(c) n'apparaît pas être une bonne approximation complète du contour de la figure 4.8(a) malgré les 72 rotations du signe. Le nombre de rotations est un facteur important pour s'assurer de la pertinence de cette famille mais au dépend du coût élevé de calcul des fonctions de taille.

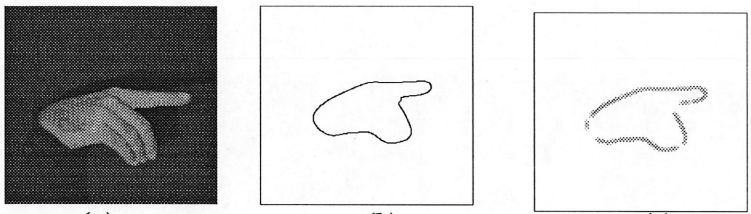


Figure 4.8: *Le problème de la pertinence d'une famille de fonctions de mesure : (a) l'image à niveaux de gris du signe «P». (b) Le contour du signe «P», (c) les points représentatifs du signe «P» en utilisant la deuxième famille de fonctions de mesure.*

Plusieures expériences ont déjà été effectuées avec l'utilisation des différentes familles de fonctions de mesure citées précédemment. Les meilleurs résultats sont obtenus avec l'utilisation de la deuxième famille de fonctions de mesure. Le tableau 4.1 montre les résultats des expériences déjà réalisées pour la reconnaissance de l'alphabet du langage des signes en utilisant la deuxième famille de fonctions de mesure citées précédemment et un système de reconnaissance basé sur la règle des k-plus proches voisins.

Durant ces expériences, trois ensembles d'apprentissage et deux personnes différentes ont été utilisés pour l'exécution des signes. Chaque ensemble d'apprentissage comprend 10 exemplaires de chaque signe (250 exemplaires pour chaque ensemble). L'ensemble d'apprentissage T1 comprend 10 exemplaires effectués par la personne S1, T2 comprend 10 exemplaires effectués par la personne S2 et finalement, T3 comprend 5 exemplaires effectués par S1 et 5 exemplaires effectués par S2. Les différents ensembles d'apprentissage ont été testés avec 20 nouveaux exemplaires effectués par S1 et S2.

Tableau 4.1: *Les résultats obtenus avec la deuxième famille de fonctions de mesure.*

Tests	T1		T2		T3	
Signes	S1	S1	S1	S2	S1	S2
«A»	1(S)	9(S)	——	2(T)	——	2(S)
«B»	——	1(F)	——	——	——	1(F)
«C»	——	——	——	——	——	——
«D»	——	9(F)	4(I)	——	——	2(R)
«E»	——	——	1(J)	——	2(J)	——
«F»	——	——	——	——	——	——
«G»	——	4(H)	——	——	——	1(H)
«H»	——	——	1(G)	——	2(G)	——
«I»	——	——	3(J)	——	3(J)	——
«J»	——	——	——	——	——	——
«K»	——	——	——	——	——	——
«L»	——	8(K)	9(C)	——	——	——
«M»	1(N)	5(N)	9(N)	——	1(N)	——
«N»	1(M)	——	1(M)	2(M)	1(M)	5(M)
«O»	——	——	——	——	——	——
«P»	——	——	——	——	——	——
«Q»	——	1(H)	2(P)	——	1(P)	——
«R»	1(U)	4(U)	1(U)	——	1(U)	4(U)
«S»	——	——	3(A)	——	1(A)	2(A)
«T»	——	——	3(S)	——	4(S)	2(E)
«U»	1(R)	4(B)	3(R)	1(R)	1(R)	1(B)
«V»	——	——	——	——	——	——
«W»	——	——	——	——	——	——
«X»	——	——	3(K)	——	——	——
«Y»	——	——	——	——	——	——
Taux	96%	82%	83%	98%	92%	92%

Les lettres entre parenthèses montrent les erreurs de classification des signes. Les taux de reconnaissance sont montrés à la dernière ligne du tableau 4.1. Notons que si l'ensemble d'apprentissage et les tests de validation sont composés de signes effectués par la même personne (T1-S1 ou T2-S2), alors, dans ce cas, le taux de reconnaissance est situé entre 96% et 98%. Le taux de reconnaissance diminu s'il s'agit des signes effectués par une personne et validés sur un ensemble d'apprentissage constitué de signes effectués par une autre personne (T1-S2 ou T2-S1). Cependant, ce taux de reconnaissance croît vers les 90% s'il s'agit de l'ensemble d'apprentissage constitué d'un mélange de signes effectués par deux différentes personnes (T3-S1 et T3-S2).

Nous pouvons constater que malgré le nombre de rotations effectuées pour chaque signe, la famille des fonctions de mesure est toujours incapable de distinguer plusieurs signes dont on peut citer le signe «D» qui est confus avec le signe «F», «L» et «K», «M» et «N», «S» et «T» et finalement «U» et, «B» et «R».

Après plusieures expériences empiriques et une inspection particulière de la structure des formes des signes, nous constatons que les informations pertinentes de chaque signe sont relativement situées au-dessus ou au-dessous d'un axe de référence spécifique au signe en question. Au lieu de considérer 72 axes de référence, il semble suffisant de prendre les axes principaux d'inertie (l'axe principal majeur et mineur) comme axes de référence pour aboutir à une paire de fonctions de mesure. Ceci permet d'éviter de faire plusieures rotations afin de couvrir l'information contenue dans chaque signe.

4.7 Notre modèle de représentation

La plupart des caractéristiques proposées dans le chapitre 3 pour l'analyse et la représentation des formes, semblent être appropriées pour des cas particuliers comme pour les objets polyhédriques rigides ou la reconnaissance des caractères. Cependant, elles ne sont pas suffisamment flexibles pour être appliquées dans le cas des postures du langage

des signes (voir le tableau 3.1). Pour éviter un nombre élevé de fonctions de taille comme dans le cas des expériences de C. Uras [49, 46], nous proposons l'utilisation des axes principaux d'inertie comme axes de référence.

Comme il a été expliqué dans le chapitre 3, les moments d'inertie d'ordre 2 peuvent être utilisés pour déterminer les axes principaux des contours des signes. Les axes principaux sont décrits par une paire d'axes qui correspondent au moment minimal et maximal d'ordre 2 (l'axe principal majeur et l'axe principal mineur respectivement).

Comme nous l'avons déjà spécifié dans le chapitre 3, l'orientation de l'un des axes principaux est déterminée spécifiquement par les signes de μ_{11} et $\mu_{20} - \mu_{02}$ comme le montre l'équation 3.1.

Le tableau 4.2 montre comment déterminer l'orientation de l'axe principal majeur θ en fonction des moments d'ordre 2 et de l'orientation de l'axe principal mineur ϕ incluant les cas particuliers où $\mu_{11} = 0$ et $\mu_{20} = \mu_{02}$. Notons que dans notre expérience, il n'est pas nécessaire de distinguer l'axe principal majeur et l'axe principal mineur puisque nous faisons la moyenne des deux fonctions de taille induites par les deux axes principaux.

Tableau 4.2: *Les orientations des axes principaux d'inertie (majeur et mineur).*

μ_{11}	$\mu_{20} - \mu_{02}$	ϕ	θ
0	-	0	$+\frac{\pi}{2}$
+	-	$0 \succ \phi \succ -\frac{\pi}{4}$	$+\frac{\pi}{2} \succ \theta \succ +\frac{\pi}{4}$
+	0	0	$+\frac{\pi}{4}$
+	+	$\frac{\pi}{4} \succ \phi \succ 0$	$+\frac{\pi}{4} \succ \theta \succ 0$
0	0	0	0
-	+	$0 \succ \phi \succ -\frac{\pi}{4}$	$0 \succ \theta \succ -\frac{\pi}{4}$
-	0	0	$-\frac{\pi}{4}$
-	-	$+\frac{\pi}{4} \succ \phi \succ 0$	$-\frac{\pi}{4} \succ \theta \succ -\frac{\pi}{2}$

Nous allons utiliser les axes principaux d'inertie comme axes de référence uniques en plus

du centre de gravité pour définir une nouvelle paire de fonctions de mesure.

Une paire de fonctions de mesure :

l'introduction des moments d'inertie est rendue nécessaire pour éviter le problème de calcul des fonctions de taille induites par une famille de fonctions de mesure citée dans les sections précédentes. Au lieu d'une famille de fonctions de taille, nous utilisons seulement une paire de fonctions de taille dont l'une est associée à l'axe principal d'inertie majeur et l'autre à l'axe principal d'inertie mineur. Il est clair qu'un seul axe principal ne garantie pas la discrimination entre tous les signes puisque le problème est semblable à celui rencontré dans le cas de la distance à partir de l'axe horizontal.

Les deux fonctions de mesure sont définies par :

$$\varphi_1(p) = \begin{cases} d(p) & \text{si } p \quad \text{s'allonge au-dessus de l'axe principal majeur} \\ 0 & \text{sinon} \end{cases}$$

et

$$\varphi_2(p) = \begin{cases} d(p) & \text{si } p \quad \text{s'allonge au-dessus de l'axe principal mineur} \\ 0 & \text{sinon} \end{cases}$$

où $d(p)$ est la distance du point du contour p à partir du centre de gravité. Le nombre de fonctions de taille associées à chaque signe est réduit à un binôme de fonctions de taille $\bar{l}_{\varphi_1}$ et $\bar{l}_{\varphi_2}$. Nous aurions pu considérer tous les points du contour avec l'ajout d'une ou plusieures fonctions de mesure afin d'inclure les points du contours non considérés, mais cette hypothèse s'est avérée inutile puisque des expériences empiriques montrent que cette paire de fonctions de taille est suffisante pour extraire les informations pertinentes contenues dans les contours des signes. Les figures 4.9(b) et 4.9(d) représentent les fonctions de taille $\bar{l}_{\varphi_1}$ et $\bar{l}_{\varphi_2}$ induites par l'axe majeur et l'axe mineur du contour du signe «C».

Nous pouvons conclure que ces deux fonctions de mesure sont invariantes à l'échelle et à la translation par le fait même qu'elle sont basées sur les moments d'inertie centraux.

La non-invariance à la rotation est intrinsèque aux axes principaux. En effet, deux signes semblables ayant deux orientation différentes auront des axes principaux différents, mais deux fonctions de taille identiques. Afin d'obtenir une représentation sensible à l'orientation des signes, nous devons tenir compte des orientations des axes principaux (majeur et mineur).

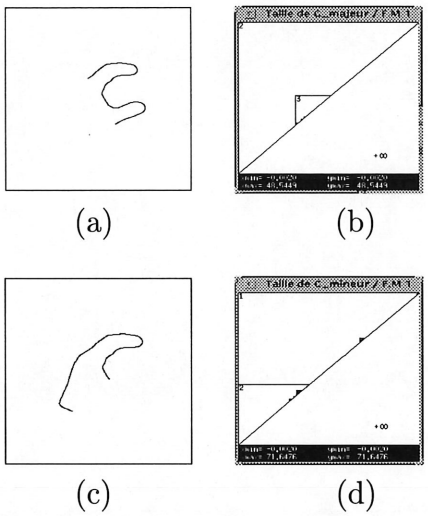


Figure 4.9: *Une paire de fonctions de mesure: (a) et (c) représentent les points considérés du contour du signe «C» situés au-dessus des axes principaux majeur et mineur correspondants au contour du signe «C», (b) et (d) représentent les graphes des fonctions de taille induites par les deux fonctions de mesure φ_1 et φ_2.*

Le critère de robustesse par rapport aux nombres de doigts et de vallées dans les postures est acquis grâce aux propriétés topologiques des fonctions de taille comme nous l'avons spécifié au début de ce chapitre.

4.7.1 Normalisation des fonctions de taille

Les dimensions des fonctions de taille ainsi que les valeures maximales et minimales des fonctions de mesures sont dépendantes des signes. Comme il a été dit précédemment, les fonctions de taille sont représentées dans une région triangulaire limitée par les valeurs minimales et maximales des fonctions de mesure φ_1 et φ_2. La normalisation des fonctions

de taille est indispensable pour nous permettre de fixer une méthode de reconnaissance. Cette normalisation est décrite en deux étapes :

- **L'étape 1 : normalisation des fonctions de mesure φ_1 et φ_2**

  Soit χ l'ensemble des points du contour d'un signe quelconque, $\varphi_\chi(P)$ une des fonctions de mesure et P est un point de χ. Nous considérons que les valeurs $M = \max_{P\in\chi}(\varphi_\chi(P))$ et $m = \min_{P\in\chi}(\varphi_\chi(P))$ sont respectivement, le maximum et le minimum de la fonction φ_χ.

  Nous définissons une nouvelle fonction $\bar{\varphi}_\chi(P) = \dfrac{\varphi_\chi(P) - m}{M - m}$ de telle façon que nous aurons toujours $\min_{P\in\chi}(\bar{\varphi}_\chi(P)) = 0$ et $\max_{P\in\chi}(\bar{\varphi}_\chi(P)) = 1$. Sachant que $\bar{\varphi}_\chi$ est la fonction de mesure normalisée telle que $0 \leq \bar{\varphi}_\chi(P) \leq 1$. Notons que dans notre application, χ contient plus de deux points et $m \neq M$.

- **L'étape 2 : algorithme de normalisation des fonctions de taille**

  Étant donnée une fonction de mesure normalisée, cet algorithme permettra d'obtenir des fonctions de taille représentées par des matrices carrées :

  - pour chaque i, j dans $\{1, 2, ...r\}$:

    calculer la valeur de $u_{ij} = l_{(\chi,\bar{\varphi}_\chi)}(\frac{i}{r}, \frac{j}{r})$ où r est un entier positif.

La matrice carrée u_{ij} est la fonction de taille normalisée discrète. Nous obtenons toujours une matrice carrée $r * r$, pour toutes les formes considérées. Sachant que pour tout $(i, j) \in \{1, 2, ...r\}$ alors, $(\frac{i}{j}, \frac{i}{j}) \in T_{\bar{\varphi}}$ où $T_{\bar{\varphi}}$ est la région triangulaire qui représente la fonction de taille normalisée.

4.7.2 Technique de normalisation en présence du bruit

Il peut arriver que des objets χ_1 et χ_2 bruités, ont des formes similaires mais les valeurs M et m de χ_1 à χ_2 sont différentes. Il s'en suit que les fonctions de taille normalisées basées sur la fonction de mesure normalisée, peuvent être différentes.

Un exemple peut éclaircir la situation. Nous considérons un ensemble χ_1 et un ensemble χ_2 (voir fig. 4.7.2(a) et (b) resp.) tel que $\chi_2 = \chi_1 \cup P$, où P est un point isolé loin de χ_2.

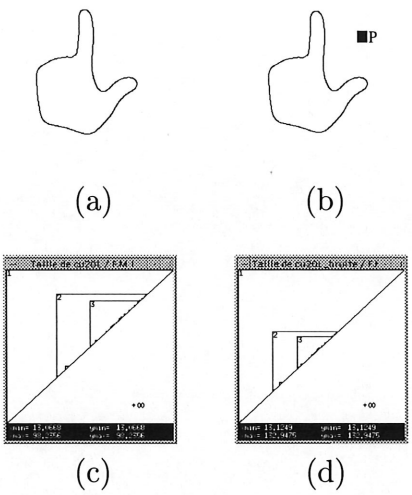


Figure 4.10: *Le problème de normalisation des formes bruitées: (a) le signe «L», (b) le signe «L» bruité, (c) et (d) les fonctions de taille des contours (a) et (b) respectivement.*

Nous pouvons constater que les deux formes sont similaires mais, la valeur $\max_{P \in \chi_2} \varphi_{\chi_2}(P)$ est beaucoup plus grande que $\max_{P \in \chi_1} \varphi_{\chi_1}(P)$. Ce fait implique que les fonctions de taille normalisées associées aux deux ensembles χ_1 et χ_2 sont différentes (fig. 4.7.2(c) et (d)).

Dans le but de faire face au problème de normalisation en présence du bruit, nous pouvons procéder selon l'algorithme suivant :

- nous fixons $\mu = \frac{1}{n} \sum_{i=1}^{n} \varphi_\chi(P_i)$ où les points P_i représente une discrétisation fixe du contour du signe.
- Nous posons ensuite, $d_k^+ = (\sum_{\varphi_\chi(P_i) > \mu} (\varphi_\chi(P_i) - \mu)^k)^{\frac{1}{k}}$ où k est un nombre réel positif.
- Nous posons similairement $d_k^- = (\sum_{\varphi_\chi(P_i) < \mu} (\mu - \varphi_\chi(P_i))^k)^{\frac{1}{k}}$.
- Nous posons finalement $M_k = \mu + d_k^+$ et $m_k = \mu - d_k^-$.

Nous pouvons constater contrairement aux valeurs de M et m que les valeurs M_k et m_k sont plus stables quand nous ajoutons un bruit à l'ensemble χ. On peut montrer que:

$$\lim_{k\to+\infty} M_k = M, \quad \lim_{k\to+\infty} m_k = m, \quad \lim_{k\to 0^+} M_k = +\infty \quad \text{et} \quad \lim_{k\to 0^+} m_k = -\infty$$

pour une discrétisation (P^i) fixe et φ fixe. Si nous choisissons un k plus grand, les valeurs de M_k et m_k seront égales approximativement aux valeurs de M et m qui sont déjà considérées comme inconvénients en présence du bruit. Si nous choisissons un k très petit, nous obtenons M_k et m_k très larges, alors la région carrée $[0,1] \times [0,1]$ où nous représentons la fonction de taille normalisée, va correspondre à des intervales très larges $[m_k, M_k] \times [m_k, M_k]$, par conséquent, beaucoup de détails sont perdus. Nous avons choisi $k = 4$ pour l'implantation de notre système.

4.8 Conclusion

Après avoir discuté des propriétés des fonctions de taille dans le cas continu, nous avons ensuite discrétisé celles-ci pour des fins de calcul et finalement, nous avons proposé une nouvelle approche d'utilisation des fonctions de taille induites par une seule paire de fonctions de mesure à base des axes principaux d'inertie. Cette nouvelle paire paraît être suffisante et pertinente pour distinguer toutes les postures de l'alphabet du langage des signes. Contrairement aux autres familles des fonctions de mesure utilisées déjà par C.Uras et A.Verri [48, 47] dont le vecteur de caractéristique représente la moyenne de 72 fonctions de taille qui correspondent aux 72 rotations des postures. Un vecteur de caractéristiques d'un signe est obtenu par la moyenne et la normalisation de seulement deux vecteurs de base correspondants à l'axe principal majeur et l'axe principal mineur d'inertie de la posture.

Dans le chapitre suivant, nous allons utiliser ce schéma de représentation pour former des prototypes afin de construire notre système de reconnaissance autour d'un réseau connexionniste perceptron multi-couche.

CHAPITRE 5

Reconnaissance des signes

Ce chapitre décrit le système d'analyse et de reconnaissance des postures du langage des signes que nous avons implanté ainsi que les résulats obtenus. Il faut signaler que nous avons validé notre système dans le cas du langage des signes italien.

Il faut se rappeler que le but principal de ce projet est celui d'exploiter les propriétés potentielles des fonctions de taille et de leur intégration comme vecteur de caractéristiques dans un système de reconnaissance basé sur un classifieur neuronal. Nous devons d'abord définir, dans ce qui suit, l'architecture du système et les différents modules qui constituent notre système de reconnaissance.

5.1 Architecture du système

Pour réaliser ce travail, plusieures étapes sont nécessaires (fig. 5.1). Leur objectif est de réduire la quantité des informations manipulées en extrayant les plus pertinentes.

La première étape consiste en l'acquisition des images de signes à l'aide d'une caméra et ensuite à la binarisation, appelée aussi seuillage, dont le but est de transformer l'image numérisée en 256 niveaux de gris en une image noire et blanche où le noir est le fond de l'image et le blanc représente la main. L'étape de seuillage est rendue indispensable

par la quantité d'information véhiculée et la nécessité de réduire cette information. Cela consiste à restituer le plus d'information et le moins de bruit possible.

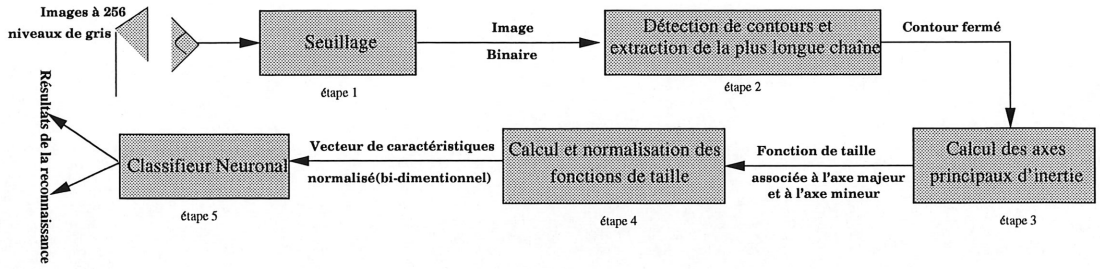


Figure 5.1: *L'architecture du système de reconnaissance.*

La deuxième étape consiste à la détection de la plus longue chaîne dans l'image binaire des différents signes en utilisant un détecteur de contour de Canny. Ainsi, nous avons utilisé un algorithme relativement simple qui consiste à identifier les trous dans le contour et le fermer avec les pixels de ses plus proches voisins.

Les étapes de pré-traitement proprement dites étant effectuées, nous devons maintenant procéder à la reconnaissance des signes. Pour cela, il est nécessaire d'avoir une représentation des contours des signes obtenus dans la deuxième étape. Ainsi, la troisième étape consiste à calculer pour chaque signe l'axe principal majeur et l'axe principal mineur selon les formules données dans le chapitre précédent. Ces axes principaux vont servir comme axes de référence dans le calcul des fonctions de taille dans la quatrième étape.

Ainsi, la quatrième étape nous permet de construire deux fonctions de taille pour chaque signe pour ensuite, passer à la normalisation selon l'algorithme présenté à la fin du chapitre 4 et au calcul de la moyenne, pour finalement obtenir un vecteur bi-dimentionnel de 146 caractéristiques incluant les orientations des axes principaux (majeur et mineur) afin d'avoir une représentation sensible à la rotation des signes. En dernier lieu, un module de reconnaissance neuronale est chargé de la classification des signes à l'aide des fonctions de taille normalisées.

Les deux premières étapes étant évidentes, elles ne seront pas décrites ici. La troisième et la quatrième étape ont été largement présentées dans le chapitre 4. Donc, nous allons présenter dans ce qui suit, la cinquième étape de notre système soit le module de reconnaissance.

5.2 Module de reconnaissance

Après l'étape de représentation, la classification reste à accomplir. Il s'agit d'un classifieur neuronal que nous avons mis en oeuvre pour effectuer cette tâche. Le réseau prend en entrée les fonctions de taille normalisées sous forme de vecteur de caractéristiques. Il doit reconnaître le plus de signes possibles avec un faible taux d'erreur. Il doit surtout éviter les erreurs de classification au risque de faire légèrement baisser le taux de reconnaissance car il doit être fiable.

Après plusieures expériences empiriques, nous avons constaté qu'un seul réseau suffit pour la reconnaissance de plusieures séries de signes. Il s'agit d'un classifieur neuronal M.L.P. (Multi Layer Perceptron) utilisant pour l'apprentissage, l'algorithme du gradient conjugé (SCG). L'utilisation d'un réseau de neurone incrémental (R.C.E. ou Grossberg) aurait pu sembler logique pour obtenir un système adaptif qui aurait évolué en fonction de son utilisation. Mais les performances (taille, temps d'exécution) de ce genre de réseaux nous ont conduit à l'abandonner.

La topologie du réseau utilisée est simple : c'est un réseau à trois couches complètement interconnectées les unes aux autres (fig. 5.2). La première couche comporte 146 neurones, les entrées de ces neurones étant les caractéristiques décrites précédemment dans la section précédente, à savoir la moyenne des fonctions de taille normalisées (12×12) et les deux orientations des axes principaux. Le nombre de neurones de la deuxième couche dépend de la séparabilité des caractéristiques. Plus le problème est linéairement séparable moins nous aurons besoin de neurones dans cette couche. En pratique, un nombre de 81

neurones dans cette couche s'est avéré suffisant. La dernière couche comporte évidemment 25 neurones qui correspondent aux 25 classes possibles de l'alphabet du langage des signes italien («A»... «Y»).

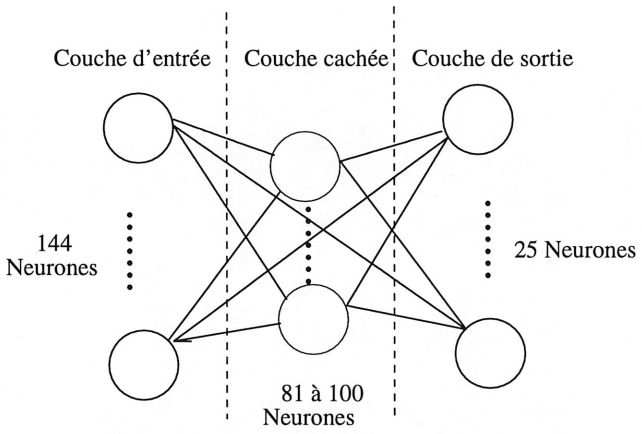


Figure 5.2: *La topologie du réseau utilisé.*

Dans ce qui suit, nous allons présenter l'algorithme d'apprentissage du réseau de neurone qui consiste à modifier les poids des connections inter-neuronales pour que les neurones de sortie aient les valeurs désirées en fonction des entrées que nous lui présentons.

L'algorithme d'apprentissage utilisé ici est la fonction SCG (le gradient conjugé) [30]. C'est un algorithme adapté aux réseaux de neurone de type (Multi Layer Feedward). Cet algorithme est considéré comme un membre de la famille des méthodes du gradient conjugé. Il permet généralement une convergence assez rapide vers un état stable, proche de l'état idéal. L'idée directrice de l'algorithme est la suivante : minimiser le plus possible l'erreur de classification représentée par la différence entre la sortie effective et la sortie désirée. Nous modifions donc les poids des connections inter-neuronales avec un certain pas, que nous appelons gain, dans le sens de la diminution du gradient. Les poids des connections sont modifiés en commençant par ceux de la couche de sortie. Si le gain est faible, l'algorithme a de grandes chances de converger éventuellement dans un minimum

local. Un gain élevé permet d'éviter quelque peu les pièges des minimas locaux mais fragilise la convergence. Au départ les poids sont initialisés aléatoirement.

La fonction d'apprentissage du gradient conjugé (SCG) est basée sur la matrice Hessian. Puisque cette matrice n'est pas toujours définie positive, alors, SCG utilise un scalaire λ qui permet de la rendre positive. Les paramètres de la fonction d'apprentissage SCG sont les suivants :

- σ : un scalaire positif appelé aussi le paramètre d'apprentissage, qui permet de spécifier la valeur de la descente du gradient. Il doit satisfaire la condition : $0 < \sigma \leq 10^{-4}$. Nous l'avons fixé à 10^{-4}.

- λ : un scalaire positif qui permet de rendre la matrice Hussian définie positive. Il doit satisfaire la condition: $0 < \lambda \leq 10^{-6}$. Nous l'avons fixé à 10^{-4}.

- δ : un paramètre de contrôle de l'apprentissage. $\delta = \max_j(t_j - o_j)$ où t_j est la valeur de l'apprentissage et o_j la valeur de la couche de sortie. Les valeurs typiques autorisées sont 0, 0.1, 0.2. Nous l'avons fixé à 0.

- ϵ : un paramètre de contrôle de la précision. Il doit être égal à 10^{-8} pour une simple précision où égal à 10^{-16} pour avoir une double précision. Pour des fins de précision. Nous l'avons fixé à 10^{-16}.

Il a été montré que SCG converge plus rapidement que les autres méthodes de gradients conjugés [15]. De plus, l'ordre de présentation des vecteurs d'apprentissage n'a pas d'effets sur le taux d'apprentissage.

La figure 5.3 présente les résultats obtenus en terme de taux de reconnaissance sur la base d'apprentissage en fonction du nombre d'itérations effectuées pendant l'apprentissage. Ces résultats ont été obtenus avec une couche cachée de 81 neurones et un gain faible

de 0,02. L'apprentissage a été effectué sur 6 prototypes de chaque signe. Nous voyons ici que le taux de reconnaissance cesse de croître fortement dès que l'on atteint un millier d'itérations.

En théorie, il existe une configuration des paramètres du réseau (nombre de couches cachées, nombre de neurones des couches cachées, gain) pour laquelle la convergence est assurée. Cependant aucune méthode ne permet de déterminer tous ces paramètres. Ce n'est qu'empiriquement que nous pouvons fixer les valeurs de ces paramètres.

Dans la pratique, pour le cas qui nous intéresse, le gain de descente du gradient qui s'est avéré suffisant est assez bas (par rapport à d'autres utilisations de ce réseau). Les résultats obtenus avec ce réseau sont assez satisfaisants puisqu'avec 81 neurones dans la couche cachée, et un gain de descente du gradient de 0,02, il converge rapidement sur une base d'apprentissage de seulement six prototypes en peu d'itérations. Selon la figure 5.3, nous pouvons obtenir un taux de reconnaissance d'environ 80% après seulement 400 itérations. Ce résultat est d'autant plus intéressant qu'il semble, dans l'état actuel des travaux, être stable. C'est-à-dire que le taux de reconnaissance continu d'avoisiner 95% avec de faibles variations des paramètres.

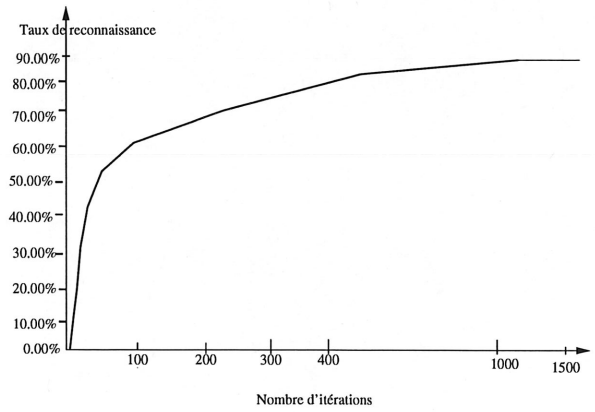


Figure 5.3: *Le taux de reconnaissance en fonction du nombre d'itérations.*

L'expérimentation a été effectuée en grande partie à l'aide du logiciel SNNS version 4.1, un simulateur de réseaux de neurone développé à l'Université de Sttutgart en Allemagne. Notre objectif consiste à créer un environnement flexible et efficace dans le but de reconnaître l'alphabet du langage des signes.

5.3 Présentation des résultats expérimentaux

Nous avons validé notre système en considérant un ensemble d'images de postures de l'alphabet du langage des signes italien qui provient de l'Université de Bologne en Italie, fournit par Claudio Uras et qui comprend plusieurs exemplaires de chaque signe effectué par deux personnes différentes.

Nous disposons de deux séries d'images réalisées par deux personnes différentes S1 et S2. Chacune des deux séries est composée de 10 images par signe. Nous avons en fait 500 images. Un aperçu de quelques-uns de ces signes est présenté dans la figure 3.2 (une série de signes effectués par la première personne S1) et la figure 3.3 (une série de signes effectués par la deuxième personne S2).

Dans le but de réaliser plusieurs tests différents et significatifs, nous avons construit trois réseaux de neurone différents et indépendants les uns des autres correspondants à trois ensembles d'apprentissage :

- T1 : *apprentissage pour une personne (S1)*

  Il s'agit ici de former le réseau à partir des données d'une seule des deux personnes. Nous avons formé un premier réseau constitué d'un ensemble d'apprentissage qui comprend 10 images de signes effectués par la personne S1. Une fois cette tâche accomplie, il est nécessaire de tester le réseau obtenu avec d'autres exemplaires provenant de la même personne et ensuite d'autres exemplaires provenant de l'autre personne S2. Nous n'avons pu effectué la validation des signes provenant de la même

personne (S1) puisque nous ne disposons pas d'autres nouveaux exemplaires de la même personne S1. Cependant, nous avons pu effectuer la validation de 10 exemplaires d'images par signe effectués par la personne S2. Les résultats obtenus sont montrés dans la première colonne du tableau 5.1 où la lettre entre les parenthèses indique la sortie activée et le chiffre avant les parenthèses indique le nombre de fois qu'elle a été activée. Par exemple, le signe «A» est confondu deux fois avec le signe «E».

- T2 : *apprentissage pour une personne (S2)*

  Similairement au réseau précédent, nous avons construit un deuxième réseau à partir des 10 exemplaires de signes provenant de la deuxième personne S2. Une fois cette tâche accomplie, il est nécessaire de tester le réseau obtenu avec d'autres exemplaires provenant de la même personne et ensuite d'autres exemplaires provenant de l'autre personne S1. Les résultats obtenus sont montrés dans la deuxième colonne du tableau 5.1. Nous n'avons pu effectué la validation des signes provenant de la même personne (S2) puisque nous ne disposons pas d'autres nouveaux exemplaires de la même personne (S2).

- T3 : *apprentissage pour plusieures personnes (S1 et S2)*

  Il s'agit ici de former un troisième réseau avec un ensemble d'apprentissage plus significatif pour qu'il reconnaisse avec un taux de succès élevé, les signes de plusieures personnes (2 personnes dans notre expérience).

  Dans un premier temps, nous faisons l'apprentissage du réseau avec 12 exemplaires pour chaque posture de l'alphabet du langage des signes (6 exemplaires sont effectués par la première personne (S1) et 6 exemplaires d'images de postures sont effectués par la deuxième personne (S2)). Dans un deuxième temps, nous avons effectué la validation des 8 exemplaires restant (4 exemplaires de la série de signes

effectués par la personne (S1) et 4 exemplaires de la série de signes effectués par la personne (S2)). Les résultats obtenus sont montrés dans la troisième colonne du tableau 5.1.

Le taux de reconnaissance montré dans la dernière ligne du tableau est obtenu par le rapport du nombre d'erreurs et le nombre d'images utilisées pour la reconnaissance (250 images pour S2/T1, 250 images pour S1/T2, 200 images pour S1/T3 et 200 images pour S2/T3).

Tableau 5.1: *Le tableau récapitulatif des erreurs de classification des signes.*

Tests	T1	T2	T3	
Signes	S2	S1	S1	S2
«A»	2(E)	2(E)	——	——
«B»	1(F)	1(F)	——	1(F)
«C»	——	——	——	——
«D»	——	1(I)	——	——
«E»	——	1(A)	——	——
«F»	——	——	——	——
«G»	——	——	——	——
«H»	1(G)	——	1(G)	——
«I»	——	——	1(O)	——
«J»	1(M)	——	——	——
«K»	——	——	——	——
«L»	——	1(D)	——	——
«M»	2(N)	——	——	——
«N»	1(M)	——	1(M)	——
«O»	——	1(H)	——	1(H)
«P»	1(G)	——	——	——
«Q»	——	1(C)	——	1(C)
«R»	——	——	——	——
«S»	1(A)	1(T)	1(A)	1(T)
«T»	2(S)	1(S)	1(S)	——
«U»	——	2(V)	1(R)	——
«V»	——	——	——	——
«W»	1(Q)	——	——	——
«X»	1(U)	1(G)	1(O)	——
«Y»	——	——	——	——
Taux	85%	89%	93%	96%

Nous avons constaté que si l'ensemble d'apprentissage et l'ensemble de validation sont effectués par deux personnes différentes (voir les deux premières colonnes du tableau 5.1), le taux de reconnaissance varie entre 85% et 89% avec un nombre d'exemplaires de 10 images par personne utilisées au niveau de l'apprentissage du réseau. Ce pourcentage croît jusqu'à 96% s'il s'agit du test T3 puisque l'ensemble d'apprentissage est constitué du mélange d'exemplaires effectués par les deux personnes (S1) et (S2) malgré que l'ensemble d'exemplaires utilisé pour le test T3 est moins élevé que ceux de T1 et T2.

Si nous regardons les différents tests de validation indiqués dans le tableau 5.1, la majorité des erreurs montrent la confusion des signes «A» et «E», «B» et «F», «S» et «T» et finalement «M» et «N». Ce fait n'est pas surprenant et il est dû particulièrement à la similarité de ces signes comme le montre la figure 5.4.

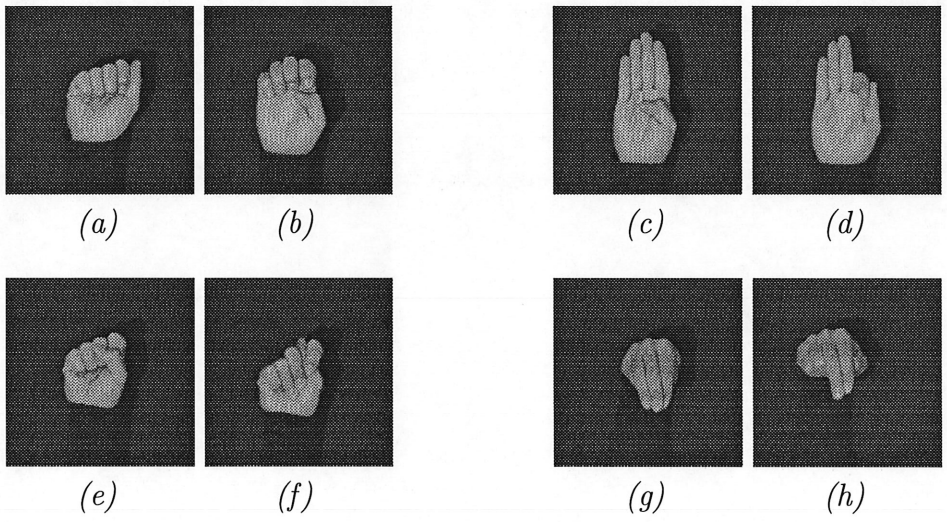


Figure 5.4: *La similarité des signes: (a) et (b) représentent les signes «A» et «E», (c) et (d) les signes «B» et «F», (e) et (f) les signes «S» et «T» et finalement (g) et (h) représentent la similarité des signes «M» et «N» respectivement.*

Cependant, si nous formons un ensemble d'apprentissage constitué d'un mélange d'exemplaires des deux personnes (S1) et (S2) comme dans les tests S1/T3 et S2/T3, les confusions diminues. Nous pouvons conclure que notre système effectue la reconnaissance

des signes avec un taux de succès satisfaisant et un degré de cohérence remarquable. Une meilleure reconnaissance de ces signes requièrt l'obtention des informations sur les contours internes. Le processus entier (détection, suivi des contours, le calcul des axes d'inertie, le calcul des fonctions de taille ainsi que la reconnaissance des signes) prend moins de deux secondes sur une machine SPARC. Les étapes de pré-traitement requièrent plus de 50% du temps total.

Si nous comparons ces résultats avec les résultats obtenus par C. Uras récapitulés dans le tableau 4.1, nous constatons que les taux de reconnaissance que nous avons obtenus dans le cas des tests S2/T1 et S1/T2 sont plus élevés que ceux obtenus par C. Uras qui sont respectivement de 82% et 83% (voir la deuxième et la troisième colonne du tableau 4.1) puisque dans ce cas, nous avons utilisé le même nombre d'exemplaires au niveau de l'apprentissage. D'autre part, dans le cas des tests S1/T3 et S2/T3, nous avons obtenu un taux de reconnaissance un peu plus élevé que celui obtenu par C. Uras (voir les deux dernières colonnes du tableau 4.1). Il s'agit d'une différence minime, ceci est dû aux nombres restreints d'exemplaires que nous avons utilisés pour l'apprentissage du réseau. Nous avons utilisé seulement 4 exemplaires pour chaque signe effectués par la personne (S1) et (S2) relativement aux 10 exemplaires pour chaque signe effectués par (S1) et (S2) dans le cas des tests effectués par C. Uras.

Si nous comparons les résultats obtenus montrés dans le tableau 5.1 relativement aux autres systèmes de reconnaissance de postures montrés dans le tableau 2.1, nous constatons que le taux de reconnaissance que nous avons obtenu est plus élevé que ceux obtenus par Takahashi, Gao, Freeman, Grimson et celui de Tamura.

Nous pouvons conclure que la performance de notre système est justifiée par les résultats obtenus au niveau des différents tests de validation que nous avons effectués. D'autant plus, que notre système est basé seulement sur le calcul d'une paire de fonctions de taille à la place des 72 fonctions de taille dans le cas des expériences de C. Uras [47, 49] .

5.4 Conclusion

Dans ce chapitre, nous avons présenté un schéma de représentation et de reconnaissance des postures de l'alphabet du langage des signes. Ce nouveau schéma est basé sur le concept des fonctions de taille et d'une reconnaissance neuronale. Contrairement aux deux systèmes proposés précédemment par C. Uras et A. Verri [47, 48], les postures sont représentées par un vecteur de caractéristiques qui est la moyenne de seulement deux fonctions de taille incluant l'orientation des axes principaux d'inertie (majeur et mineur). L'autre différence est située au niveau de l'utilisation d'un classifieur neuronal pour la reconnaissance des postures à la place de la règle des k-plus proches voisins. Les résultats obtenus indiquent que notre système est capable d'achever la reconnaissance des signes avec un taux de succès satisfaisant. Nous concluons que l'intégration des fonctions de taille comme vecteur de caractéristique dans un classifieur neuronal permet de construire un système efficace de reconnaissance des postures des signes et d'obtenir des résultats satisfaisants.

Conclusion

Nous avons présenté dans cette étude, l'analyse ainsi qu'un prototype de système de reconnaissance de l'alphabet du langage des signes. Un soin particulièr a été apportée aux traitements de bas niveaux pour permettre au système d'être assez robuste. Les algorithmes proposés et réalisés dans cette partie donnent des résultats satisfaisants. La reconnaissance neuronale utilisée dans notre système est une approche encore peu utilisée avec les fonctions de taille.

Les fonctions de taille sont des fonctions à valeur entière de deux variables réelles qui sont utilisées pour représenter les formes visuelles. Dans ce projet, le potentiel des fonctions de taille dans le domaine de la reconnaissance des formes a été démontré. Plusieurs propriétés attrayantes et essentielles des fonctions de taille, dont la robustesse face à la variabilité du signal et les changements minimes dans les formes, ont été illustrées. Ainsi, la conception d'une paire de fonctions de taille invariante par rapport aux différentes transformations géométriques des images des signes. Un système de reconnaissance basé sur les fonctions de taille induites par une paire de fonctions de mesure et les orientations des axes principaux d'inertie ainsi qu'un un réseau neuronal ont été décrits. Les expériences rapportées montrent la performance de ce système avec des scores moyens de reconnaissance élevés.

Malgré les résultats intéressants obtenus, beaucoup de travail doit être effectué spécia-

lement au niveau de la représentation. Pratiquement, cette représentation par les fonctions de taille nécessite des contours continus; l'utilisation des fonctions de mesure multidimensionnelles doit donc être explorée. En second lieu, l'option d'utiliser une moyenne de plus de deux fonctions de mesure peut augmenter le taux de reconnaissance des postures.

Le système réalisé semble prometteur mais il reste néanmoins à l'affiner davantage, surtout en ce qui concerne la partie classification. L'approche consiste en la spécialisation de réseaux afin de s'attaquer aux ambiguïtés des signes montrés dans le tableau 5.1. Sur la base du travail présenté, un système très performant devrait pouvoir être réalisé.

Bibliographie

[1] Y. Abu-Mustapha and A. Psaltis. *Recognitive Aspects of the Moment Invariants. IEEE Trans. Patt. Anal. and Machine Intell.*, 6:5698–5706, 1984.

[2] Franz L. Alt. *Digital Pattern Recognition by Moments. Journal of the ACM*, 11:240–258, 1962.

[3] N. Ayache and O.D. Faugeras. *HYPER: A New Approach for the Recognition and Positioning of Two-Dimentional objects. IEEE Trans.Patt. Anal. Mach. Intell*, 8:44–54, 1986.

[4] Dana H. Ballard and Christopher M. Brown. *Computer Vision. Prentice-Hall*, 1982.

[5] Abdel Belaid and Yolande Belaid. *Reconnaissance des formes. InterEditions*, 1992.

[6] S.O. Belkasim, M. Shridhar, and M. Ahmadi. *Pattern Recognition with Moment Invariants : A comparative Study and New Results. Pattern Recognition*, 24:1117–1138, 1991.

[7] U. Bellugi and E. Klima. *The Sign Language. Harvard University Press, Cambridge*, 1979.

[8] J. E. Besancon. *Vision par ordinateur en deux et trois dimensions. Eyrolles, Paris*, 1988.

[9] M. Bichsel. *Strategies of Robust Object Recognition for the Automatic Identification of Humain Faces. Ph.D thesis, Institute of Technology, Zurich*, 1988.

[10] Sing-Tze Bow. *Pattern Recognition and Image Preprocessing. Marcel Dekker, Inc.*, 1992.

[11] C. K. Chow. *Statical Independence and Threshold Functions. IEEE Transactions on Electrical Computer*, 14:66–68, 1965.

[12] William T. Freeman and Michal Roth. *Orientation Histograms for Hand Gesture Recognition. IEEE Int. Workshop. on Automatic Face and Gesture Recognition*, Zurich, june, 1995.

[13] P. Frosini. *Measuring Shapes by Size Function.* In *Proc. of SPIE on Intelligent Robotics and Computer Vision X, Boston, Mass.*, volume 1607, pages 3–26, 1991.

[14] P. Frosini. *Discrete Computation of Size Functions. J. Combin. Inform. System Sci.*, 17 3-4:232–250, 1996.

[15] Wen. Gao. *Enhanced User by Hand Gesture Recognition. CHI'95 Wokshop on User Interface by Hand Gesture, Denver*, juin, 1995.

[16] Wen Gao. *Pattern Recognition with Moment Invariants: A comparative Study and New Results. CHI'95 Workshop on User Interface by Hand Gesture*, pages 45–53, Denver 1995.

[17] Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing. Addison-Wesley Publishing Company*, 1977.

[18] Monika M. Gorkani and Rosalind W. Picard. *Texture Orientation for Sorting Photos "at a Glance".* Technical report, *M.I.T. Media Laboratory Perceptual Computing Section,No. 292*, 1994.

[19] C. Gourley. *Neural Networks Utilizing Posture Input for Sign Language Recognition. Rapport technique, Computer Vision and Robotics Research Laboratory, University of Tenessee Knoxville*, novembre, 1994.

[20] W.E.L. Grimson and T. Loranzo-Perez. *Localization Overlapping Parts by Searching the Interpretation Tree. IEEE Trans. Patt. Anal. Mach. Intell*, 9:469–482, 1987.

[21] C. Hand, I. Sexton, and M. Mullan. *A Linguistic Approach to the Recognition of Hand Gestures. Actes de Designing Future Interaction, Ergonomics Society/IEE, University of Warwick, UK*, avril, 1994.

[22] Robert M. Haralick and Linda G. Shapiro. *Computer and Robot Vision*, volume 1. *Addison-Wesley Publishing Company*, 1992.

[23] Robert M. Haralick and Linda G. Shapiro. *Computer and Robot Vision*, volume 2. *Addison-Wesley Publishing Company*, 1993.

[24] Philip A. Harling. *Gesture Input Using Neural Networks.* Technical report, *University of York, UK.*, 1993.

[25] M. Hu. *Visual Pattern Recognition by Moment Invariants. IRE Trans. Inf. Theory*, 8:179–187, 1962.

[26] J.-F. Jodouin. *Les réseaux de neuronnes: Principes et définitions. Édition Hermès*, 1994.

[27] W. Kadous. *GRASP: Recognition of Australian Sign Language using Instrumented Gloves.* Technical report, *Bachelor of Computer Engineering, University of New South Wales*, 1995.

[28] H. Lane, P. Boyes-Braem, and U. Bellugi. *Preliminaries to a Distinctive Feature Analysis of Handshapes in American Sign Language. Cognitive Psychology*, vol.8:263–289, 1976.

[29] Rung-Huei Liang and Ming Ouhyoung. *A Sign Language Recognition System Using Hidden Markov Model and Context Sensitive Search.* Technical report, *Communication and Multimedia Lab., Dep. of Computer Science and Information Engineering, National Taiwan University*, 1995.

[30] M. F. Moller. *A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. Neural Networks.*, 6:525–533, 1993.

[31] B. Moody. *La langue des signes: Histoire et grammaire.* vol.1, Paris, 1983.

[32] K. Murakami and H. Taguchi. *Gesture Recognition using Reccurent Neural Networks. Actes de CHI'91 Workshop on User Interface by Hand Gesture, ACM*, pages 237–242, 1991.

[33] Gregory B. Newby. *Gesture Recognition Based upon Statistical Similarity. M.I.T. Press*, 3(3):236–234, 1994.

[34] Richard J. Prokop and Anthony P. Reeves. *A Survey of Moment-Based Techniques for Unoccluded Object Representation and Recognition. CVGIP: Graphical Models and Image Processing*, 54, no.5:438–460, september 1992.

[35] A. Reeves, R. Prokop, S. Andrews, and F. Kuhl. *Three Dimentional Shape Analysis Using Moments and Fourier Descriptors. IEEE Trans. Patt. Anal. Mach. Intell.*, 10:937–943, 1988.

[36] J. Rondal, F. Henrot, and M. Charlier. *Le Langage des signes. Ed. Pierre Mardaga, Bruxelles*, 1986.

[37] D. Rubine. *The Automatic Recognition of Gestures.* Technical report, *PhD Thesis, Carnegie Mellon University*, avril 1991.

[38] K. Breeding S. Dudani and R. McGhee. *Aircraft Identification by Moment Invariants. IEEE Trans. Comput.*, 26:39–45, 1977.

[39] Robert J. Schalkoff. *Pattern Recognition: Statistical, Structural and Neural Approaches. John Wiley and Sons, Inc.*, 1992.

[40] T. E. Starner. *Whole-Hand Input.* Technical report, *PhD Thesis, MIT.*, 1992.

[41] T. E. Starner. *Visual Recognition of American Sign Language Using Hidden Markov Models.* Technical report, *Master of Science in Media Arts and Sciences, MIT.*, 1995.

[42] T. Takahashi and F. Kishino. *Hand Gesture Coding Based on Experiments Using a Hand Gesture Interface Device. SIGCHI Bulletin*, vol.23:67–74, 1991.

[43] S. Tamura and S. Kawasaki. *Visual Recognition of Sign Language Motion Images. Pattern Recognition*, vol.21:343–353, 1988.

[44] C. Uras and A. Verri. *Invariant Size Functions. Applications of Invariance in Computer Vision, Lecture Notes in Computer Science 825, Springer-Verlag, Berlin Heidelberg*, pages 215–234, 1994.

[45] C. Uras and A. Verri. *Studying Shape Through Size Functions. In: O. Y. Toet, A. Foster, D. Heijmans, H. Meer, P. (eds.), Shape in Picture, NATO, ASI, Series F, Springer-Verlag, Berlin Heidelberg*, 126:81–90, 1994.

[46] C. Uras and A. Verri. *Computing Size Functions from Edge Maps. Internat. J. comput. Vision*, 10, 1995.

[47] C. Uras and A. Verri. *Signe Language Recognition: An Application of the Theory of Size Functions. 6th British Machine Vision Conference*, 2:711–720, 1995.

[48] A. Verri and C. Uras. *On the Recognition of the Alphabet of the Sign Language Through Size Functions. Proc, XII Int. Conf. Patt. Recog., Jerusalem*, II:334–338, 1994.

[49] A. Verri and C. Uras. *A Metric-Topological Approach to Shape Representation and Recognition. Image and Vision Computing*, 14:189–207, 1996.

[50] A. Verri, C. Uras, P. Frosini, and M. Ferri. *On the Use of Size Functions for Shape Analysis. Biol. Cybern*, 70:99–107, 1993.

[51] R.A. Wagner. *Order-n Correction for Regular Languages. Communication of the ACM*, 17:No.5, 1974.

[52] Wai-Hong Wong, Wan-Chi Siu, and Kin-Man Lam. *Generation of Moment Invariants and Their Uses for Caracter Recognition. Pattern Recognition Letters*, 16:115–123, february 1995.