

THESE

présentée à

L'UNIVERSITE DE SAVOIE

pour obtenir

le grade de DOCTEUR

discipline : Electronique - Electrotechnique - Automatique

par

M. Thomas ALLEVARD

le 07 juillet 2005

Représentation symbolique de la configuration
de la main - application à la reconnaissance de signes
et au contrôle d'un robot mobile

Préparée au LISTIC

Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance

COMPOSITION DU JURY

M. Patrick MILLOT	Président
M ^{me} Bernadette BOUCHON-MEUNIER	Rapporteur
M. Patrice DALLE	Rapporteur
M. Michel de MATHELIN	Examineur
M. Laurent FOULLOY	Directeur de thèse
M. Eric BENOIT	Codirecteur de thèse

...



JORGE CHAM ©THE STANFORD DAILY

phd.stanford.edu/comics

Remerciements

Les travaux présentés dans ce mémoire ont été effectués au Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance (LISTIC) de l'Ecole Supérieure d'Ingénieurs d'Annecy (ESIA) à l'Université de Savoie.

Je tiens tout d'abord à remercier les membres du jury de l'intérêt qu'ils ont manifesté pour cette thèse par leurs questions, remarques et autres propositions de perspectives. Merci à Monsieur Patrick Millot, Professeur à l'Université de Valenciennes et du Hainaut-Cambrésis, de m'avoir fait l'honneur de présider ce jury. Merci également aux rapporteurs de cette thèse, Madame Bernadette Bouchon-Meunier, Directrice de Recherche CNRS à l'Université Pierre et Marie Curie (Paris) et Monsieur Patrice Dalle, Professeur à l'Université Paul Sabatier (Toulouse), d'avoir accepté de lire ce manuscrit malgré leur charge de travail. Merci aussi à Monsieur Michel de Mathelin, Professeur à l'Université Louis Pasteur (Strasbourg) pour ses intéressantes suggestions concernant les perspectives de ce travail.

Merci à Monsieur Philippe Bolon, Professeur à l'Université de Savoie de m'avoir accueilli dans son laboratoire. Merci à mes directeurs de thèse, Monsieur Eric Benoit, Maître de Conférences à l'Université de Savoie et Monsieur Laurent Foulloy, Professeur à l'Université de Savoie, pour leurs encouragements et la confiance qu'ils m'ont témoignés. Ils m'ont permis d'atteindre le sommet de cette longue ascension doctorale en me fournissant le matériel dernier cri pour ce genre d'aventure : curiosité scientifique, imagination, rigueur, persévérance.

Merci aux membres du LISTIC pour leur soutien et leur compagnie, à Valérie pour sa bonne humeur et ses compétences orthographiques, aux autres doctorants, compagnons de cordée, à Nacima, Sylvie et Yoann qui ont partagé mon bureau et de ce fait supporté son entropie particulièrement élevée.

Merci à chacun de mes amis, sans qui la vie perdrait de son sens et moi de mon entrain. Un merci tout particulier à ma famille, qui m'a guidé, soutenu, encouragé, entouré et continue à le faire de la plus belle des manières.

Mais si je n'avais qu'un seul merci à offrir, il irait sans hésitation, mais avec beaucoup de tendresse, à Agathe qui partage ma vie et me laisse croquer à la sienne. Je ne peux exprimer en une phrase toute ma gratitude pour son soutien indéfectible. Une vie n'y suffira peut-être pas ; mais on peut toujours essayer...

Table des matières

Remerciements	5
Table des matières	7
Introduction	15
Chapitre 1 - La mesure du geste manuel	23
1. Introduction.....	25
2. Le geste manuel	26
2.1. La structure du geste manuel	26
2.1.1. Morphologie de la main	26
2.1.2. Le geste humain : définitions	27
2.2. Les trois fonctions du geste manuel	28
2.2.1. Les fonctions épistémiques et ergotiques	29
a) La fonction épistémique	29
b) La fonction ergotique	29
2.2.2. La fonction sémiotique	30
a) Geste à nu	30
b) Les différents gestes à nu : typologie	31
c) Le geste dactylographique et le geste instrumental	32
3. La mesure du geste manuel.....	34
3.1. Les capteurs de flexion	35
3.1.1. Capteurs optiques	35

3.1.2. Capteurs à effet Hall	36
3.1.3. Capteurs résistifs	36
3.1.4. Autres systèmes	37
3.2. Les systèmes de localisation	38
3.2.1. Systèmes mécaniques	38
3.2.2. Systèmes électromagnétiques	39
3.2.3. Systèmes acoustiques	40
3.2.4. Systèmes optiques	40
3.3. Les gants numériques de la littérature	41
3.3.1. Point de vue historique	41
3.3.2. Gants plus récents	44
3.3.3. Tableau récapitulatif	48
3.3.4. Gants de commande binaires	48
3.3.5. Bracelets numériques	49
3.4. Systèmes non intrusifs : systèmes de vision	51
4. Conclusion.....	53

Chapitre 2 - Les différentes approches de la reconnaissance de signes 57

1. Introduction.....	59
2. Reconnaissance de signes - généralités.....	60
2.1. Structure d'un système de reconnaissance de signes	60
2.2. Notations - règle de Bayes	62
3. Reconnaissance de signes statiques	65
3.1. k-plus proches voisins (k-PPV)	65
3.2. Classifieur de Bayes	66
3.3. Réseaux de neurones	67
3.4. Bases de règles - systèmes experts	68
3.4.1. Arbres de décision	70
3.4.2. Classifieurs à base de règles floues	71
3.5. Tableau récapitulatif	74
4. Reconnaissance de signes dynamiques	75
4.1. Systèmes supprimant la variabilité temporelle des signes dynamiques	75
4.1.1. Différenciation des gestes selon leur dynamique	75
4.1.2. Normalisation temporelle des gestes	76
4.1.3. Reconnaissance de composantes invariantes	78
4.2. Systèmes prenant en compte la variabilité temporelle des signes dynamiques	79
4.2.1. Modèles de Markov Cachés	79
4.2.2. Réseaux de neurones récurrents	81
4.3. Tableau récapitulatif	84
5. Conclusion.....	85

Chapitre 3 - Introduction au formalisme des ϕ -symbolisations 89

1. Introduction.....	91
2. Symbolisations	92
2.1. Définition	92
2.2. Symbolisations nettes	93
2.2.1. Définition	93
2.2.2. Exemple	94
2.3. Symbolisations floues	95

2.3.1. Définition	95
2.3.2. ϕ -symbolisations	96
2.3.3. Exemple	98
2.3.4. Sémantique des fonctions d'appartenance	98
3. Construction de ϕ -symbolisations	99
3.1. Construction d'une ϕ -symbolisation sur un intervalle réel	99
3.1.1. Construction d'une ϕ -partition sur un intervalle réel par des intervalles flous (L-R)	99
3.1.2. Construction d'une ϕ -symbolisation sur un intervalle de \mathbb{R}	101
3.1.3. Exemple	102
3.2. Produit cartésien de ϕ -symbolisations	102
3.2.1. Produit cartésien de ϕ -partitions	103
3.2.2. Produit cartésien de ϕ -symbolisations	103
a) Définition	103
b) Propriétés	104
3.2.3. Exemple	105
3.3. Regroupement de symboles d'une ϕ -symbolisation	106
3.3.1. Réunion d'éléments d'une ϕ -partition	106
3.3.2. Regroupement de symboles d'une ϕ -symbolisation	107
3.3.3. Changement d'univers symbolique d'une ϕ -symbolisation par regroupement de symboles	109
3.3.4. Utilisation de règles	110
a) Règles	110
b) Notations	112
c) Exemple	113
3.4. Conclusion - forme des significations obtenues	115
4. Relations d'équivalence floues associées à une symbolisation	117
4.1. T -équivalences et T -égalités	117
4.1.1. Définitions	117
4.1.2. Métriques généralisées et T -équivalences	118
4.1.3. T -équivalences et symbolisations floues	120
4.2. Relations $(\perp_L - T^*)$	120
4.3. Relations $(\perp_L - T_M)$ et Id -symbolisations	121
4.3.1. Relations $(\perp_L - T_M)$: définition et propriétés	121
4.3.2. Relations $(\perp_L - T_M)$ et Id -symbolisations	123
a) Propriétés	123
b) Comparaison d'entités provenant de deux Id -symbolisations	123
c) Limites de la relation $(\perp_L - T_M)$	125
4.4. Autres relations $(\perp - T^*)$	126
4.4.1. Relations $(\perp_L - T_P)$	126
4.4.2. Relations $(\perp_L - T^*)$	127
4.4.3. Relations $(\perp_L^\phi - T_M)$	127
4.5. Autres T -équivalences sur $F(S)$	128
5. Extension d'une métrique définie sur les symboles	130
5.1. Distances induites sur $F_{Id}(S)$: principe	130
5.2. Application des distances définies dans la littérature à $F_{Id}(S)$	133
5.2.1. Approche géométrique	133
5.2.2. Fuzzification	133
5.2.3. Pondération	135

5.2.4. Approche morphologique	136
5.3. Distance du transport	137
5.3.1. Transformations $T_{s_i, s_j, x}$	137
5.3.2. Distance du transport d_{tp}	138
5.3.3. Illustration	143
5.3.4. Rapport avec la distance $d_{(\perp_L - T_M)}$	145
6. Conclusion.....	147

Chapitre 4 - Description lexicale de la configuration de la main _____ 151

1. Introduction	153
2. Description lexicale de la configuration de la main.....	154
2.1. Choix des ensembles lexicaux et définition des symbolisations	154
2.1.1. Ecartement des doigts	154
2.1.2. Flexion des grands doigts	154
2.1.3. Description du pouce	155
a) Flexion	155
b) Orientation	155
2.2. Construction des symbolisations	155
2.2.1. Ecartement des grands doigts	156
2.2.2. Flexion des grands doigts	158
2.2.3. Description du pouce	160
a) Flexion du pouce	160
b) Orientation du pouce	160
2.3. Schéma général	162
2.3.1. L'ensemble des postures P	162
2.3.2. Schéma général	164
2.3.3. La symbolisation $\langle P, L_P, R_P \rangle$	165
2.4. Exemple	165
3. Conclusion.....	168

Chapitre 5 - Reconnaissance de signes _____ 171

1. Introduction	173
2. Reconnaissance de signes statiques	174
2.1. Formalisation de la notion de signe statique	174
2.1.1. Introduction	174
2.1.2. Exemples	174
2.2. Reconnaissance de l'alphabet de la Langue des Signes Française (LSF)	178
2.2.1. L'alphabet de la Langue des Signes Française	178
2.2.2. Résultats expérimentaux	181
2.2.3. Amélioration possible : le cas du signe M	183
2.3. Conclusion	186
3. Reconnaissance de signes dynamiques	187
3.1. Introduction	187
3.2. Distances entre postures	188
3.2.1. Distance $d_{(\perp_L - T_M)}$	189
a) Utilisation de la symbolisation $\langle P, L_P, R_P \rangle$	189
b) Utilisation d'une symbolisation restreinte	190
3.2.2. Distance du transport	191

a) Construction d'une distance sur l'univers symbolique d'une ϕ -symbolisation	191
b) Distance du transport avec la symbolisation $\langle P, L_P, R_P \rangle$	196
c) Distance du transport avec une symbolisation restreinte	196
3.2.3. Conclusion	197
3.3. Distances entre gestes	197
3.3.1. Normalisation des gestes	197
3.3.2. Distance entre gestes	201
3.4. Reconnaissance	202
3.4.1. Première approche pour la reconnaissance	202
3.4.2. Deuxième approche pour la reconnaissance	204
a) Modèle de signe dynamique	204
b) Reconnaissance	205
c) Résultats	207
4. Conclusion.....	209

Chapitre 6 - Contrôle d'un robot par le geste de la main _____ 213

1. Introduction.....	215
2. Contrôle d'un robot mobile par des séquences de signes statiques	216
2.1. Les signes de commande	216
2.2. Segmentation des gestes et reconnaissance	217
3. Génération de commandes graduelles.....	220
3.1. Principe	220
3.1.1. Commande proportionnelle au taux de reconnaissance	220
3.1.2. Commande graduelle contrôlée par la configuration d'un doigt	221
3.2. Défuzzification symbolique	222
3.2.1. Typologie des systèmes flous	222
3.2.2. Défuzzification symbolique	223
a) Passage par une défuzzification numérique	223
b) Méthode des hauteurs	224
c) Défuzzification symbolique basée sur la distance du transport	225
3.3. Génération d'actions graduelles	229
3.3.1. Utilisation de la méthode des hauteurs	229
3.3.2. Utilisation de la distance du transport	232
3.4. Contrôle d'un robot par des actions graduelles	234
3.4.1. Principe	234
3.4.2. Génération de l'action graduelle	235
3.4.3. Exemple de trajectoire	238
4. Conclusion.....	240

Conclusion _____ 243

Annexes _____ 249

1. Regroupement des symboles d'une ϕ -symbolisation à 3 symboles	251
2. Démonstration de la proposition 16 du chapitre 3	252
3. Reconnaissance de signes dynamiques	255
4. Composantes des signes dynamiques.....	257
5. Reconnaissance de signes dynamiques - Méthode n° 2	259

Bibliographie _____ 261



JORGE CHAM © 2002

www.phdcomics.com

Introduction

Des différents éléments constituant un ordinateur ce sont probablement les interfaces entrées-sorties qui le moins évolué ces trente dernières années, le duo clavier/souris restant, en 2005, l'interface d'entrée privilégiée pour les ordinateurs. Pourtant, de nombreux travaux portent sur les interfaces homme machines plus naturelles, dans lesquelles c'est la machine qui s'adapte au mode de communication de l'homme et non l'inverse.

Lorsque l'on parle de communication naturelle pour l'homme, on pense le plus souvent à la parole. L'utilisation de cette dernière pour l'interaction homme-machine constitue un très large champ d'investigations et des interfaces basées sur la parole sont aujourd'hui largement diffusées dans le grand public comme les systèmes de reconnaissance vocale des téléphones portables. Seulement, si l'utilisation de la parole dans la communication homme-machine constitue un point de passage indispensable vers une communication plus naturelle, il est également clair qu'elle possède, au delà des limites technologiques, des limites intrinsèques obligeant à considérer d'autres modalités de communication. Il y a en effet des situations où l'utilisation de la parole est impossible, soit parce que le bruit environnant est trop élevé (concert de rock par exemple) ou qu'il doit rester particulièrement faible (concert de musique de chambre), soit encore parce que l'air, permettant la propagation du son, est absent de l'environnement dans lequel on se trouve (plongée sous-marine). Mais, plus que tout, une interface homme-machine uniquement basée sur l'usage de la parole ne serait pas naturelle tout simplement parce que la communication humaine ne peut se résumer à cette seule modalité.

La communication humaine est en effet profondément multi-modale et les efforts de concentration devant être accomplis pour maintenir une conversation téléphonique, n'impliquant donc que la parole, illustrent parfaitement ce point. Entre autres modalités de communication, le geste manuel tient une place d'importance : il ne se contente pas d'assister la parole mais forme avec elle une seule et même structure communicative. La différence existant entre un discours avec ou sans geste est comparable à celle existant entre des images en deux ou trois dimensions [McNeil 92]. L'existence des langues des signes est une preuve indéniable du fait que le canal gestuel possède un pouvoir d'expression au moins aussi important que celui de la parole.

Laisser de côté la main dans la communication homme-machine serait donc très réducteur. Ce serait en outre assez indélicat vis-à-vis de cet organe que nous utilisons presque constamment, qui remplit simultanément trois fonctions différentes et, surtout, sans lequel nous ne parlerions probablement pas, comme nous le rappelle A. Leroi-Ghouran [Leroi-Gourhan 64], citant G. De Nysse :

“...Ainsi, c’est grâce à cette organisation que l’esprit, comme un musicien, produit en nous le langage et que nous devenons capables de parler. Ce privilège, jamais sans doute nous ne l’aurions, si nos lèvres devaient assurer, pour les besoins du corps, la charge pesante et pénible de la nourriture. Mais les mains ont pris sur elles cette charge et ont libéré la bouche pour le service de la parole.” [De Nysse 379].

Mais si nous nous intéressons ici au geste de la main, c'est, il faut le reconnaître, pour des raisons moins nobles que celle de lui rendre hommage. L'extraction d'informations à partir du geste manuel est en effet un problème complexe et donc intéressant à bien des égards. C'est tout d'abord un problème de fusion d'informations : la main possède plus de vingt degrés de liberté, sans compter ceux correspondant à sa position et à son orientation dans l'espace. Contrairement à la parole qui peut être décrite par un seul paramètre, la pression acoustique variant au cours du temps, le geste manuel doit, lui, être décrit par un grand nombre de paramètres qu'il faut fusionner puis traiter afin d'en extraire l'information utile. Cette information utile peut en outre être de diverses natures. Il peut s'agir de reconnaître des signes statiques - c'est-à-dire des postures particulières de la main - des signes dynamiques - c'est-à-dire des gestes ayant une durée temporelle - une action effectuée par la main ou d'autres paramètres tels que la vitesse d'exécution d'un geste, son amplitude, etc.

D'autre part la communication gestuelle peut se faire uniquement par le biais de l'aspect visuel de la main et cet aspect visuel peut être décrit par des termes du langage naturel. On peut par exemple facilement imaginer quel est le signe statique dans lequel l'index est tendu et tous les autres doigts sont pliés. C'est cette particularité du geste manuel qui nous intéresse plus particulièrement. La plupart des approches utilisées pour extraire l'information contenue dans le geste manuel sont en effet purement numériques et fonctionnent telles des boîtes noires incompréhensibles par un utilisateur ordinaire. Nous allons, dans cette thèse, exploiter cette possibilité de décrire le geste manuel par des termes du langage naturel et construire un système

de fusion d'informations basé sur cette description. Les termes utilisés pouvant être considérés comme un ensemble fini de symboles particuliers, nous serons donc bien dans le cadre d'un problème de traitement d'informations symboliques.

Dans ce travail, nous nous concentrons donc sur deux objectifs :

- Décrire la configuration de la main à l'aide de termes du langage naturel. Par *configuration* nous entendons la forme que prend la main, indépendamment de sa position et de son orientation dans l'espace. Cette description sera qualifiée de lexicale ou de symbolique.
- Extraire des informations à partir de cette description lexicale de la configuration de la main, plus précisément reconnaître des signes statiques, des signes dynamiques et des commandes pour le contrôle d'un robot mobile.

Pour obtenir la description lexicale de la configuration de la main, on mesurera les angles des articulations des doigts à l'aide d'un gant numérique : le Cyberglove. C'est un gant dans lequel sont intégrés plusieurs capteurs de flexion se retrouvant au niveau des différentes articulations dont on veut mesurer l'angle. Ces mesures numériques seront ensuite converties en descriptions lexicales. Pour effectuer cette conversion, on s'appuiera sur le formalisme des symbolisations floues. Ce formalisme mathématique permet d'établir un lien entre un univers numérique et un univers symbolique, éventuellement lexical, plus facilement interprétable par l'homme. Il fournit également un cadre théorique permettant de représenter le lien qu'un capteur classique établit entre le monde réel - qui est un univers empirique - et un univers numérique. Les symboles dont il sera question dans ce manuscrit seront exclusivement des termes du langage naturel et plus particulièrement des termes se rapportant à la configuration des doigts de la main. Ce seront donc des termes tels que *tendu* ou *plié*.

Neuf symbolisations floues convertiront les données numériques issues du gant numérique en données symboliques ou lexicales. Chacune de ces symbolisations permettra de décrire la configuration d'un doigt ou d'un groupe de doigts particulier. Les données symboliques fournies par ces symbolisations prendront la forme de sous-ensembles flous de termes, appelés descriptions floues. Par exemple, la symbolisation associée à la description de l'écartement de deux doigts fournira une description floue qui sera un sous-ensemble flou de l'ensemble lexical $L_{\text{écart}} = \{\text{écartés}, \text{serrés}\}$. Le degré d'appartenance d'un terme à cette description floue représentera l'adéquation de ce terme avec l'écartement réel des doigts considérés. La figure suivante illustre ce principe. L'écartement de l'index et du majeur tels qu'ils sont représentés sur cette figure pourra être décrit par le sous-ensemble flou également représenté sur la figure 1. Dans un tel cas, on dira que les doigts sont mi-écartés, mi-serrés.

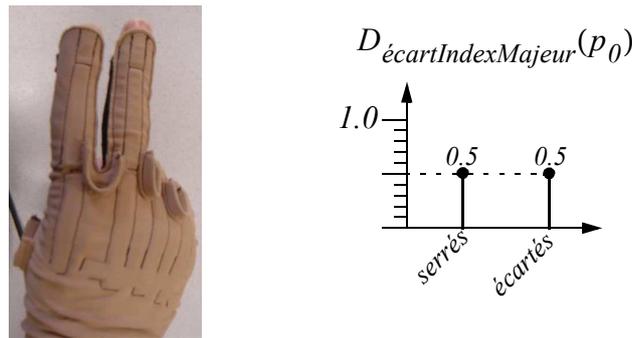


Fig. 1 Exemple de description floue de l'écartement de l'index et du majeur.

Au final, la description lexicale de la configuration de la main sera constituée de neuf descriptions floues, c'est-à-dire neuf sous-ensembles flous de termes. On utilisera un type particulier de symbolisations, les *Id*-symbolisations. Les descriptions floues obtenues auront alors la particularité d'avoir la somme de leur degré d'appartenance égale à 1. Ces *Id*-symbolisations seront présentées dans le cadre plus général des ϕ -symbolisations. Les termes utilisés dans ces descriptions seront bien sûr choisis indépendamment du système de mesure utilisé (ici un gant numérique). Ainsi, la description lexicale que nous construirons présentera l'avantage sur d'autres types de représentation d'être totalement indépendante du système de mesure. Ce sera également le cas de tous les traitements qui seront effectués par la suite sur ces données symboliques.

La comparaison des fonctions d'appartenance de deux sous-ensembles flous permet d'évaluer la similarité de ces sous-ensembles flous. On définira ainsi une relation d'égalité floue entre descriptions floues. Cette relation sera associée à un opérateur de distance qui pourra être utilisé pour effectuer des traitements sur les descriptions. Mais on verra qu'une telle relation d'égalité floue, basée uniquement sur la comparaison terme à terme des fonctions d'appartenance présente de sérieuses limitations. Nous introduirons alors un opérateur de distance entre descriptions floues permettant de prendre en compte l'existence d'une métrique sur l'ensemble lexical dans la comparaison de deux sous-ensembles flous de cet ensemble lexical. Cette distance sera appelée distance du transport, car son calcul se ramène à la résolution d'un problème de transport de masses. Nous verrons que c'est en outre une extension directe de la distance entre descriptions floues basée sur la relation d'égalité floue. Cette dernière distance peut en effet être vue comme le cas discret de la distance du transport.

Le formalisme des symbolisations ainsi que les opérateurs de distance que nous aurons définis seront utilisés pour la reconnaissance de signes statiques et dynamiques. Nous verrons que ces deux types de signes peuvent être correctement reconnus à partir de la description lexicale de la configuration de la main. Nous utiliserons également cette description lexicale pour la commande d'un robot mobile par le geste de la main. Deux types de commandes seront envisagés. Dans un premier temps, un ensemble de signes statiques correspondront à un ensemble de commandes binaires. Cette première application permettra de confirmer que les

signes statiques peuvent être reconnus en temps réel. Dans un deuxième temps, nous nous attacherons à la génération de commandes graduelles. On pourra alors, par exemple, commander la vitesse d'avance du robot en fléchissant plus ou moins un doigt particulier. Cette deuxième application fera intervenir un problème de défuzzification symbolique, bien connu en contrôle flou, pour lequel nous proposerons une nouvelle approche basée sur la distance du transport. Cette nouvelle défuzzification symbolique, inspirée de la méthode classique des hauteurs, permettra de spécifier facilement l'action graduelle devant être générée.

Dans le chapitre 1, nous introduisons le cadre général de la mesure du geste de la main. Dans une première partie, nous présentons le geste manuel, sa structure ainsi que ses différentes fonctions. Nous donnons quelques définitions afin de préciser le sens des termes spécifiques à la littérature du geste manuel. Nous présentons en détail la fonction sémiotique de la main, celle par laquelle la main émet de l'information et qui nous intéresse tout particulièrement dans ce manuscrit. Dans une deuxième partie, nous faisons une revue détaillée des systèmes intrusifs de mesure du geste manuel c'est-à-dire utilisant du matériel venant se fixer sur la main. Nous présentons les capteurs de flexion utilisés ainsi que les systèmes de localisation existant, puis les différents gants numériques que l'on peut trouver aussi bien dans le commerce que dans la littérature scientifique. Nous présentons également brièvement le principe de mesure du geste de la main par les systèmes non intrusifs que sont les systèmes de vision.

Les problèmes de reconnaissance de signes statiques et dynamiques ont depuis longtemps intéressé la communauté scientifique. Nous faisons une revue des différents travaux existant dans ces deux domaines au chapitre 2, en nous concentrant essentiellement sur le problème de la reconnaissance de la configuration de la main. Nous n'aborderons donc pas dans ce chapitre les problèmes liés à la reconnaissance de gestes globaux de la main ou de langages gestuels tels que les langues des signes ou les gestes co-verbaux. Après avoir présenté le cadre général de la reconnaissance de signes, nous détaillons les principales approches utilisées pour la reconnaissance de signes statiques. Ces approches ne peuvent généralement pas être appliquées directement à la reconnaissance de signes dynamiques qui fait intervenir, en plus de la variabilité de la configuration de la main, une variabilité de la dynamique du signe - ou variabilité temporelle - devant également être prise en compte. Les méthodes utilisées pour la reconnaissance de signes dynamiques sont donc traitées dans une troisième section.

Dans le chapitre 3, nous abordons le coeur de notre travail en présentant le formalisme des symbolisations qui est utilisé par la suite. Nous présentons les ϕ -symbolisations, qui sont une extension floue particulière des symbolisations. Nous donnons une méthode de construction des ϕ -symbolisations, méthode que nous utilisons par la suite pour construire les neuf *Id*-symbolisations - cas particuliers des ϕ -symbolisations - permettant de décrire la configuration de la main. Ces neuf fLes deux dernières sections de ce chapitre sont consacrées à l'étude des opérateurs de distance pouvant être définis sur les descriptions floues. Nous introduisons alors la distance du transport, seule distance de la littérature floue, à notre connaissance, permettant d'étendre une métrique sur les symboles aux descriptions floues.

Le chapitre 4 est consacré à la construction des neuf symbolisations permettant d'obtenir la description lexicale de la configuration de la main. Les différents termes utilisés dans cette description sont alors définis, ainsi que certains ensembles particuliers, tel que l'ensemble des postures P , qui a la particularité d'être empirique et qui est largement utilisé dans les deux chapitres suivants. Cet ensemble des postures permettra de ne plus parler du gant numérique par la suite, soulignant ainsi l'indépendance de la représentation sous forme de descriptions lexicales avec le système de mesure utilisé.

La reconnaissance de signes statiques et dynamiques à partir de cette description lexicale est traitée dans le chapitre 5. Nous montrons que la reconnaissance de signes statiques ne nécessite pas d'apprentissage des signes. Il suffit de fournir la définition sous forme lexicale des différents signes à reconnaître pour être capable de les identifier par la suite. Deux approches sont envisagées pour la reconnaissance de signes dynamiques. Ces approches sont toutes deux basées sur une étape de normalisation des gestes nécessitant simplement de pouvoir calculer une distance entre postures. Une telle distance peut être calculée à partir des distances entre descriptions floues. La première approche nécessite, dans tous les cas, de disposer d'un ensemble d'apprentissage des signes dynamiques alors que la deuxième approche permet éventuellement de se passer de cet ensemble d'apprentissage, si les composantes statiques des signes dynamiques - composantes dont nous précisons alors la définition - sont suffisantes pour les séparer.

Dans le dernier chapitre, nous utilisons le geste de la main pour contrôler un robot mobile. Deux types de contrôles sont envisagés. Dans un premier temps, seule des commandes binaires sont envoyées au robot : l'opérateur fait un signe ; si celui-ci est reconnu comme un des signes statiques permettant de contrôler le robot, la commande correspondante est envoyée. Dans la deuxième partie de ce chapitre, nous étudions la possibilité de générer des actions graduelles. Il serait en effet dommage de conserver une information graduelle tout au long du processus de traitement de l'information pour ne générer au final que des actions binaires. La génération d'actions graduelles fait intervenir un problème de défuzzification symbolique pour lequel nous présentons une approche originale basée sur l'utilisation de la distance du transport.



"THANKS FOR AGREEING TO BE INTERVIEWED FOR MY STUDY ON GRAD STUDENTS. LET ME FOCUS... OK, ARE YOU READY?"

"IT'S NO PROBLEM... FIRE AWAY..."



"OK... IN YOUR OWN WORDS, CAN YOU TELL ME WHY YOU CAME TO GRADUATE SCHOOL?"

"SURE... UM..."



"...THAT'S OK, TAKE YOUR TIME... NO RUSH..."

"UM..."

JORGE CHAM ©THE STANFORD DAILY

CHAPITRE 1

La mesure du geste manuel

1. Introduction

Mesurer le geste de la main n'est pas une activité courante et il n'est pas superflu de préciser ce que cela signifie. Qu'est-ce que nous entendons par geste manuel ? Quelles sont ses différentes composantes, ses fonctions ? Comment s'organisent-elles ? Lesquelles peuvent faire l'objet de mesures et lesquelles nous intéressent plus particulièrement ? Le but de ce chapitre est d'apporter les réponses à ces questions dans un premier temps, puis de présenter différents systèmes de mesure du geste manuel.

La première partie présente donc le geste de la main. Nous rappelons très grossièrement l'anatomie de la main, puis nous donnons quelques définitions concernant le geste manuel et sa structure. Nous présentons alors les trois fonctions de la main, qui sont le plus souvent indissociables, puis nous nous concentrons sur la troisième de ces fonctions, la fonction sémiotique, par laquelle la main permet de communiquer de l'information et qui nous intéresse ici.

Dans la deuxième partie, nous présentons les systèmes de mesure du geste manuel, plus particulièrement les systèmes appelés gants numériques. Nous commençons par introduire les technologies des capteurs utilisés dans ces gants puis nous présentons les gants décrits ou utilisés dans la littérature ainsi que ceux actuellement disponibles dans le commerce. Nous finissons par une rapide présentation des principes mis en oeuvre dans les systèmes de vision.

2. Le geste manuel

2.1. La structure du geste manuel

2.1.1. Morphologie de la main

En excluant les six degrés de liberté correspondant à la position et l'orientation de la main dans l'espace, celle-ci possède vingt deux degrés de liberté propres [Ouhaddi 98]. Le système de la motricité humaine possédant en tout une centaine de degrés de liberté [Cadoz 94], les deux mains en détiennent donc, à elles seules, plus de la moitié. Le mouvement de chaque doigt est créé par trois articulations (voir Fig. 2). Les articulations interphalangiennes (IPD et IPP) possèdent chacune un seul degré de liberté (flexion/extension) et les articulations métacarpo-phalangiennes (MCP) en possèdent deux (flexion/extension et abduction/adduction), exceptée celle du pouce qui n'en possède qu'un (flexion/extension). Le pouce possède par contre une articulation de plus, l'articulation carpo-métacarpienne (CMC), qui a trois degrés de liberté (flexion/extension, abduction/adduction, pseudo-rotation). Il est donc généralement différencié des autres doigts appelés les grand doigts.

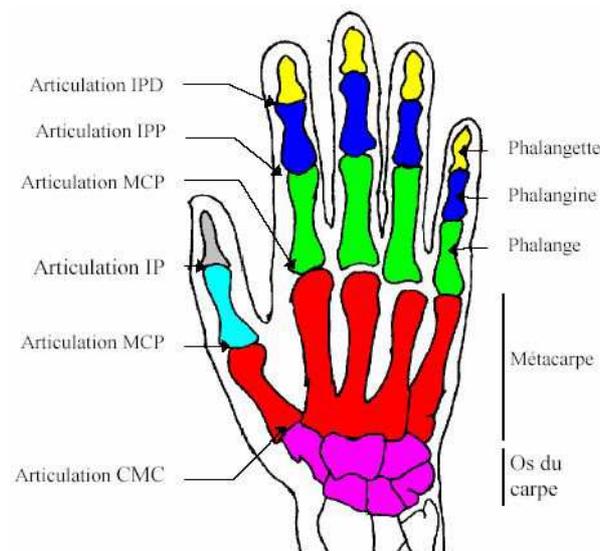


Fig. 2 Les articulations de la main (tiré de [Ouhaddi 98]).

Les vingt deux degrés de liberté de la main ne sont pas indépendants deux à deux : un ensemble de contraintes limite le nombre de configurations pouvant finalement être prises par la main. En plus des contraintes statiques, correspondant aux angles maximaux de flexion des doigts, s'ajoutent des contraintes cinématiques (aussi appelées dynamiques bien qu'elles ne fassent pas intervenir l'inertie des doigts), liant les mouvements des articulations entre eux. Par

exemple, les mouvements des IPDs et IPPs sont très fortement liés et on peut modéliser cette liaison par la relation [Ouhaddi 98] :

$$\theta^{IPD} = \frac{2}{3} \times \theta^{IPP}. \quad (1)$$

2.1.2. Le geste humain : définitions

Nous donnons ici quelques définitions relatives au geste qui éviteront les contresens dans la suite de la lecture. Nous ne prétendons pas apporter *La* définition de tel ou tel terme, mais simplement expliquer ce qu'il représente pour nous, comment nous l'entendons et l'utilisons. Le premier des termes qu'il nous faut définir est évidemment le terme *geste*. Puis nous définirons ce que sont la configuration et la localisation de la main qui sont les deux composantes du geste et enfin, nous donnerons des définitions relatives à certains termes utilisés tout au long de ce manuscrit.

- **Geste** : le terme geste signifiera, dans ce manuscrit, uniquement *geste de la main* et nous ne considérerons que les gestes d'une seule main. Un geste est alors un signal particulier, variant dans le temps et constitué de deux composantes : la configuration de la main et sa localisation. Le geste est en cela l'analogue de la parole qui est, elle, un signal acoustique.
- **La configuration de la main** : c'est la forme que prend la main. Elle est décrite par la configuration de chacun des doigts ainsi que du poignet. Cette composante a donc vingt deux degrés de liberté.
- **La localisation de la main** : elle est composée de l'orientation de la main et de sa position dans l'espace. A cette dernière peut éventuellement s'ajouter la position de la main par rapport au corps ou à un objet particulier qui peuvent entièrement changer la signification d'un geste : le geste d'une claque a sa signification et ses conséquences qui varient selon qu'il est effectué à deux mètres d'une personne ou à cinquante centimètres. Il est le plus souvent sous-entendu que la position des autres objets et même des autres parties du corps sont connues par ailleurs. La localisation est alors une composante à six degrés de liberté.
- **Posture** : une posture est un geste pris à un instant t . En reprenant l'analogie avec la parole, l'analogue de la posture serait la valeur de la pression acoustique à un instant donné.
- **Signe** : il est tentant d'écrire qu'un signe est un geste particulier d'une durée finie, mais cette définition est trop restrictive car la réalisation d'un signe comporte une grande part de variabilité. Il y a une variabilité intra-signeur, qui signifie qu'une personne ne réalise jamais deux fois le même signe, même s'il le

réalise souvent d'une façon similaire. Il y a aussi une variabilité inter-signeurs signifiant que deux personnes ne réalisent pas le même signe de la même manière. Un signe est donc finalement une classe de gestes. Remarquons que l'on ne définit pas un signe d'après sa signification. Il se peut d'ailleurs qu'un signe n'ait qu'une fonction ergotique (voir paragraphe suivant) et donc pas de signification intrinsèque ou qu'il ait une signification qui varie selon le contexte. Enfin, nous distinguerons les signes dynamiques des signes statiques.

- **Signe dynamique** : un signe dynamique est une classe de gestes ayant une certaine durée dans le temps.
- **Signe statique** : un signe statique est une classe de postures. Il n'a donc pas de durée temporelle.

2.2. Les trois fonctions du geste manuel

Puisque nous nous intéressons au geste de la main, il est intéressant de savoir quelles sont ses différentes fonctions et comment elles sont organisées. Celles-ci sont au nombre de trois [Cadoz 94] :

- **Fonction épistémique** : la main reçoit de l'information. Elle permet de sentir les textures, les formes, les températures, etc.
- **Fonction ergotique** : la main produit de l'énergie et la fournit au monde physique pour le transformer. Elle joue le rôle d'organe moteur pour faire bouger des objets, les tordre, les tourner, etc.
- **Fonction sémiotique** : la main produit de l'information. Cela peut se faire entre autre par son aspect visuel (geste à nu) ou par le biais d'un objet physique (clavier, instrument, etc.).

Ces trois fonctions sont en fait indissociables et généralement remplies simultanément. Prenons l'exemple d'une poignée de main. La fonction ergotique est clairement présente puisqu'il faut serrer la main de l'autre personne. La fonction épistémique est elle aussi remplie car nous sentons la chaleur de l'autre main, sa rugosité, sa taille, la pression qu'elle exerce sur notre main. La fonction sémiotique est enfin clairement présente : on va signifier un degré de sympathie à l'autre personne par un mouvement, une tonicité particulière de la main. Ne dit-on pas d'une poignée de main qu'elle peut être froide ou chaleureuse?

Voyons maintenant plus en détail comment sont remplies ces fonctions et les rapports qui existent entre elles. Nous nous attarderons plus longuement sur le geste sémiotique puisque c'est celui qui est à l'honneur dans ce manuscrit.

2.2.1. Les fonctions épistémiques et ergotiques

a) La fonction épistémique

Les informations fournies par la main sont de trois types. Premièrement, les récepteurs mécanosensoriels situés dans la peau de la main - environ au nombre de 17000 [Fuchs 03] - fournissent l'information sur la surface d'un objet : sa température, sa texture, sa rugosité, etc. On dit qu'ils fournissent l'information tactile. Il est intéressant de remarquer que notre capacité à capter cette information tactile est fortement diminuée si nous nous contentons de laisser nos doigts reposer sur une surface. Il faut qu'il y ait un mouvement pour obtenir de bonnes sensations tactiles [Aukstakalnis 92]. Deuxièmement, par le biais des muscles et des tissus, la main nous renseigne sur le poids des objets, leur rigidité. C'est le sens kinesthésique. Associé à une certaine stratégie de palpation, il nous renseigne également sur la forme des objets. Enfin, le canal proprioceptif nous renseigne sur la position de la main dans l'espace et ainsi sur les positions des objets, leur mouvements, leurs trajectoires, leur orientation, etc. Ces trois types d'informations sont fournies simultanément et les trois canaux correspondant sont très fortement intercorrélés, formant finalement ce qui est appelé le sens tactilo-proprio-kinesthésique [Cadoz 94].

De même que des instruments de mesure ont souvent besoin de modifier l'environnement pour le mesurer (mesures intrusives), la main a souvent besoin d'agir sur un objet pour en tirer l'information dont on a besoin. Ainsi, la fonction épistémique est souvent précédée de la fonction ergotique comme l'illustre la figure 3.

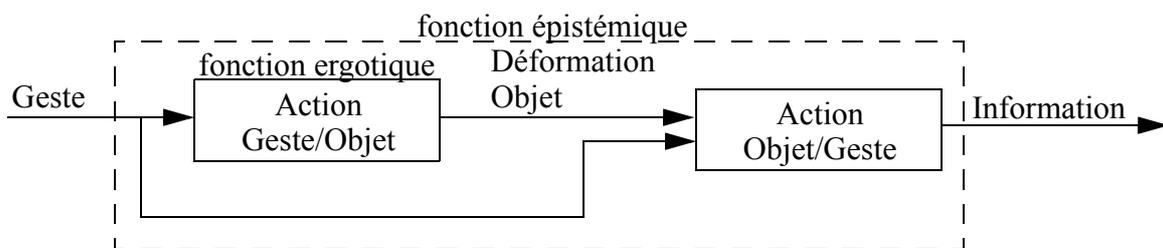


Fig. 3 Le geste épistémique est souvent précédé d'un apport d'énergie (geste ergotique).

b) La fonction ergotique

Différents types d'actions se référant à la fonction ergotique (préhension, lâcher, appui, etc.) ont été étudiés pour les applications de réalité virtuelle ou celles d'apprentissage par démonstration. On trouvera un résumé de ces études dans [Fuchs 03]. On remarquera simplement ici que la fonction ergotique est toujours accompagnée d'un retour d'information, par le biais de la fonction épistémique, permettant de contrôler les déformations du monde physique.

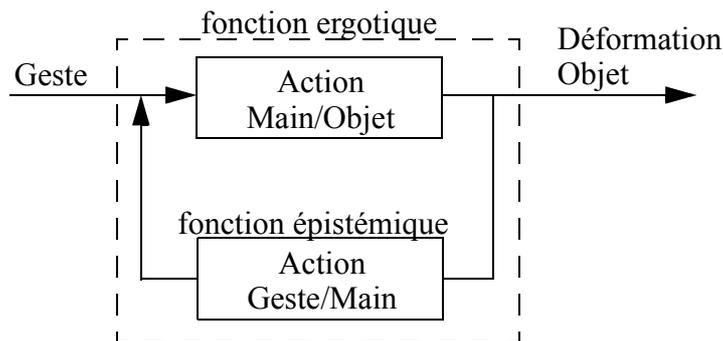


Fig. 4 Le geste ergotique est contrôlé par un retour d'information (geste épistémique).

2.2.2. La fonction sémiotique

On appelle *canal gestuel* le canal de communication utilisant les gestes de la main pour faire passer de l'information. Ce canal gestuel possède plusieurs modalités : la main peut émettre de l'information de différentes manières. Avec ses nombreuses configurations possibles, elle peut s'adresser à la perception visuelle d'un ou plusieurs interlocuteurs, comme c'est le cas pour les langues des signes : on parle alors de *geste à nu* [Cadoz 94]. C'est ce type de geste qui nous intéresse dans ce manuscrit et nous le décrivons donc plus en détail. Il est néanmoins intéressant de comprendre comment la main peut également, pour s'exprimer, utiliser un intermédiaire matériel. Ainsi, le geste dactylographique utilise le clavier pour faire passer une information sous forme de chaîne de caractères et le geste instrumental musical utilise un instrument pour faire passer une information sous forme de son. Ces deux types de gestes sont donc présentés à la fin de ce paragraphe.

a) Geste à nu

Le geste à nu s'adresse à l'oeil sans intermédiaire matériel. Il est, comme la parole, exclusivement sémiotique : il ne sert qu'à faire connaître [Cadoz 94]. Son pouvoir d'expression dépend du langage utilisé (voir paragraphe b). L'information est contenue dans l'apparence visuelle du geste, dans les signes le constituant. Sa structure peut donc être modélisée ainsi :

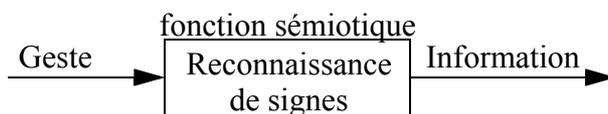


Fig. 5 Structure du geste à nu.

Il est généralement accepté que le geste à nu se décompose en quatre éléments, appelés les quatre composantes du geste, qui sont la configuration de la main pendant ce geste, son orientation, son mouvement dans l'espace et la zone où est effectué ce mouvement par rapport au corps. La connaissance de la configuration et de la localisation de la main pendant un geste,

telles qu'on les a définies au paragraphe précédent, permet donc, si besoin est, d'extraire les quatre composantes du geste à nu.

b) Les différents gestes à nu : typologie

Les études tentant de dresser une classification des gestes de la main ont commencé avec les travaux pionniers de D. Efron sur les comportements non verbaux des immigrants juifs et italiens de New-York [Efron 41]. Elles continuent aujourd'hui, s'intéressant soit au geste sémiotique en général, soit à des gestes particuliers à une application : gestes musicaux [Cadoz 94], gestes d'interaction avec une interface multi-modale [Steininger 01], etc.

On trouve donc de nombreuses classifications différentes des gestes sémiotiques. La typologie de Ekman et Friesen, par exemple, essaie de dresser une liste des différents types de gestes *à nu* utilisés par l'homme pour leur fonction sémiotique [Ekman 72]. La classification de Quek [Quek 94] est plus orientée vers les gestes de la communication homme-machine. Celle de C. Cadoz [Cadoz 94] s'intéresse exclusivement au geste instrumental. La dernière classification, que nous présentons ici, sépare les différents langages gestuels selon leur pouvoir d'expression et la présence ou non de la parole dans leur utilisation. Elle a été proposée par A. Kendon [Kendon 88] et se présente sous la forme d'un continuum : il n'existe pas de limite franche entre les différentes classes. Elle est communément appelée *continuum de Kendon* [McNeil 92].

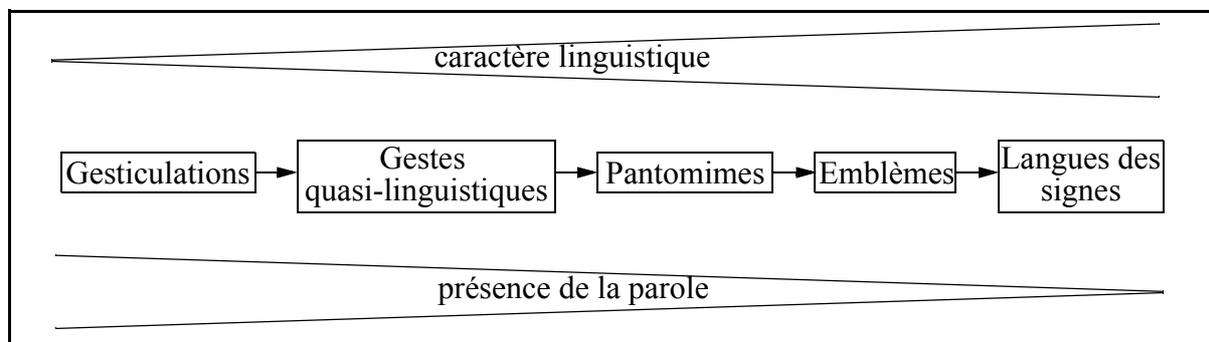


Fig. 6 Le continuum de Kendon [Kendon 88].

- Les gesticulations sont des mouvements spontanés, idiosyncrasiques - c'est-à-dire particuliers à un individu ou un groupe d'individus - qui accompagnent le discours. Par exemple, un conteur qui lève la main en disant « Il a grimpé jusqu'au sommet ».
- Les gestes quasi-linguistiques sont très proches des gesticulations à la différence qu'ils sont intégrés au discours. Par exemple : « La montée était facile, mais la descente, [geste] ».
- Les pantomimes sont les gestes sans parole utilisés dans un mime. Ils ont une structure puisque c'est l'enchaînement de ces gestes qui a un sens, contrairement aux gesticulations et aux gestes quasi-linguistiques qui ne se combinent pas.

- Les emblèmes peuvent être utilisés dans un discours, mais ils sont indépendants du canal verbal [Braffort 96]. Les exemples typiques de ces gestes sont ceux employés pour les insultes ou les éloges. Ils sont souvent utilisés sans la parole, ce qui est probablement leur raison d'être puisqu'ils permettent d'exprimer une conviction en contournant les tabous de la du langage parlé [McNeil 92]. Ils sont aussi généralement définis très précisément. Par exemple, dans le signe de l'auto-stoppeur, le pouce est dressé vers le haut et tous les autres doigts sont pliés : une personne dressant un autre doigt vers le haut aura peu de chances de se faire prendre en stop et risque même de se faire quelques ennemis. Les emblèmes sont propres à une culture, ils varient selon les pays et les classes sociales.
- Les langues des signes sont des ensembles de gestes et de postures qui forment un langage à part entière. Elles sont dotées d'une syntaxe et utilisent toute la puissance d'expression du geste humain. Elles ont beaucoup été étudiées car un système d'interface homme-machine qui serait capable de reconnaître une langue des signes pourrait prendre en compte toutes les potentialités du canal gestuel [Braffort 96]. Il faut noter cependant que ces langues ne se limitent pas à des gestes des mains. Elles font également intervenir des mimiques faciales et des postures du corps tout entier.

Les langages de commande se situent également à l'extrémité droite du continuum de Kendon, mais ils ont un vocabulaire réduit et une syntaxe triviale. Ils ne permettent pas de remplacer le langage de manière continue mais seulement dans certaines situations. Un exemple typique est le langage utilisé en plongée sous-marine. Dans ce manuscrit, il sera principalement question de gestes de commande.

c) Le geste dactylographique et le geste instrumental

Le geste dactylographique est également sémiotique. Son but est bien de produire une chaîne de caractères, donc de l'information, mais, contrairement au geste à nu, il doit, pour s'exprimer, utiliser un support matériel : un clavier. Il fait donc intervenir la fonction ergotique de la main, qui fournit de l'énergie pour appuyer sur les touches et n'est pas exclusivement sémiotique. Ainsi, le schéma du geste dactylographique est le suivant :

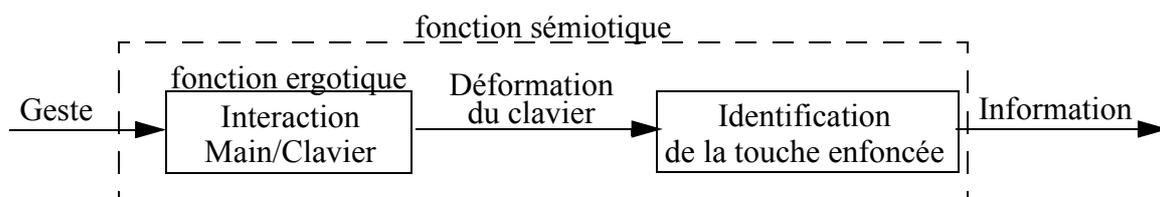


Fig. 7 Structure du geste dactylographique.

Cependant, dans le geste dactylographique, le clavier a un rôle normatif. On ne modifie pas la façon d'appuyer sur une touche en fonction de l'information que l'on veut faire passer. L'énergie transmise au clavier ne contient aucune information. On pourrait tout à fait remplacer les touches mécaniques d'un clavier par des touches tactiles ne nécessitant pas d'apport d'énergie. La fonction ergotique du geste dactylographique ne contient pas d'information et c'est ce qui le distingue du geste instrumental.

Dans le geste instrumental, tel que défini par C. Cadoz [Cadoz 00], les fonctions épistémiques, ergotiques et sémiotiques sont impliquées. Le geste instrumental modifie tout d'abord la structure de l'instrument pour que celui-ci produise un son. Il utilise donc la fonction ergotique de la main. Le son produit par l'instrument contient une information qui doit être décodée par l'ouïe des auditeurs et du musicien. Le geste instrumental produit donc bien de l'information et est un geste sémiotique. Mais lors du geste instrumental, la main remplit également sa fonction épistémique et renseigne le musicien sur la déformation actuelle de l'instrument. Ce retour d'information, associé au retour qu'il a par le biais de son ouïe, lui permet de contrôler son geste de manière à produire exactement le son voulu (voir Fig. 8). Finalement, la structure de la production d'information par le geste instrumental pourrait être schématisée comme suit :

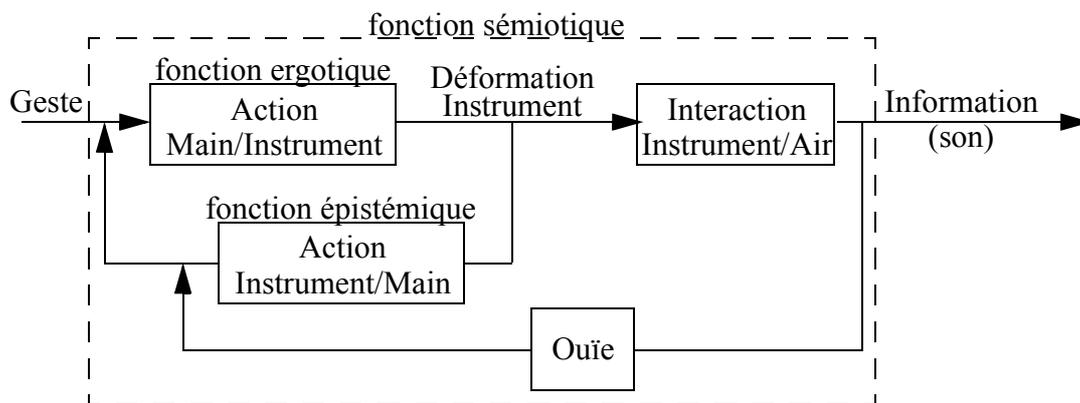


Fig. 8 Du geste au son : le geste instrumental.

Dans le geste instrumental, l'énergie fournie à l'instrument contient de l'information. On ne peut pas se passer de la fonction ergotique et de la fonction épistémique qui lui sont associées et qui permettent d'avoir une interaction avec l'instrument. C'est cette information, contenue dans l'énergie du geste ergotique, ainsi que l'interaction avec l'instrument par le biais de la fonction épistémique, qui caractérisent le geste instrumental.

3. La mesure du geste manuel

Dans cette partie, nous décrivons les différentes technologies utilisées pour mesurer le geste de la main. Nous nous intéressons uniquement à la mesure du geste à nu, c'est-à-dire à la mesure directe ou indirecte de son aspect visuel. Quatre types de mesures pouvant être effectuées ont été identifiés :

- **Mesure de la configuration de la main** : ces mesures donnent les angles des articulations de la main. On peut théoriquement, à partir de telles mesures et d'un modèle 3D de la main, reconstituer la posture effectuée.
- **Localisation de parties de la main** : on donne la position d'une ou plusieurs parties de la main. Le plus souvent, on s'intéresse à la position et à l'orientation globale de la main. On peut aussi mesurer la position et l'orientation des extrémités des doigts.
- **Détection de contacts** : ce sont des mesures binaires qui informent sur la présence ou non de contact entre des parties de la main, par exemple entre le pouce et un autre doigt.
- **Mesure d'effets indirect** : on peut mesurer les effets indirects de la position de la main ou du mouvement de la main et utiliser ces mesures pour commander un dispositif quelconque.

Notons que si l'on s'intéressait à un autre type de geste que le geste à nu, d'autres mesures, principalement des mesures d'effort, pourraient être effectuées. Elles n'ont ici pas d'intérêt et nous ne mentionnerons donc pas les systèmes permettant d'effectuer ces mesures.

Les dispositifs de mesures peuvent être intrusifs - des instruments de mesure sont alors fixés sur la main - ou non intrusifs - laissant la main exempte de tout matériel. Ces derniers sont presque exclusivement des systèmes de vision.

Cette partie présente en détail les systèmes intrusifs car c'est ce type d'interface que nous utilisons pour nos travaux. Dans ces interfaces, des capteurs sont fixés sur la main et les premiers systèmes de fixation utilisés ayant été des gants, on désigne généralement tout système intrusif sous le terme de gant numérique ou gant de données. Le support de fixation peut cependant être non seulement un gant mais aussi un exosquelette, des bagues, un bracelet, etc.

Les deux principaux types de capteurs que l'on trouve sur les gants numériques sont des capteurs de flexion et des capteurs de localisation. Les deux premiers paragraphes décrivent en détail les différentes technologies sur lesquelles sont basées ces deux familles de capteurs. On présente ensuite un certain nombre de gants numériques illustrant l'évolution de ces interfaces depuis leur apparition ainsi que leur diversité. Enfin, nous décrivons brièvement les principes des systèmes de vision.

3.1. Les capteurs de flexion

Les trois principaux types de capteurs de flexion sont les systèmes optiques, qui utilisent des tubes ou des fibres optiques, les systèmes à capteurs résistifs, qui transforment la mesure d'un angle en une mesure d'élongation d'un matériau résistif et les systèmes à effet hall, qui mesurent directement les angles de flexion avec des compas à effet hall.

3.1.1. Capteurs optiques

Historiquement, les premiers capteurs conçu pour mesurer la flexion des doigts ont utilisé des tubes optiques. Un tube flexible est attaché au doigt dont on veut mesurer la flexion. A un bout de ce tube se trouve une source lumineuse et à l'autre un récepteur photoélectrique (voir Fig. 9). Plus le tube est fléchi, plus il y a de réflexions sur les parois de celui-ci et plus l'énergie lumineuse arrivant au récepteur est faible. On obtient alors un signal électrique variant en fonction de la flexion du doigt.

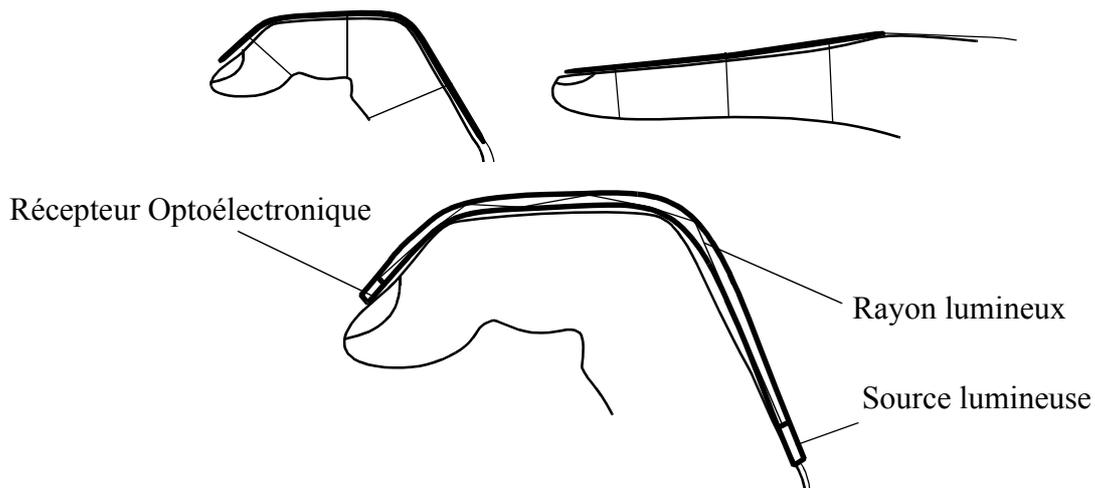


Fig. 9 Principe de la mesure de flexion par tube optique.

Ces capteurs ont été utilisés en premier par T. DeFanti et D. Sandin sur le gant qu'ils ont développé d'après une idée de R. Sayre [DeFanti 77]. L'idée a été reprise sur le Digital Data Entry Glove de G. Grimes [Grimes 83]. En 1985, Zimmerman a déposé un brevet pour des capteurs de flexion optiques [Zimmerman 85], capteurs qu'il a utilisés sur le DataGlove [Zimmerman 87]. Puis il a amélioré ces capteurs pour le DataGlove II, en utilisant des fibres optiques à la place des tubes.

La mesure de flexion par fibres optiques est toujours utilisée aujourd'hui car elle est relativement précise, peu coûteuse et permet d'obtenir des capteurs très peu encombrants. Les fibres peuvent être traitées sur une de leur face de manière à ce qu'il y ait une perte de

transmission lumineuse proportionnelle à la flexion de la fibre uniquement dans certaines plages de fréquences et à des endroits particuliers de la fibre. La société Measurand fournit de tels capteurs de flexion (ShapeSensor, ShapeTape) [www.measurand.com]. Parmi les gants instrumentés disponibles aujourd'hui sur le marché, on retrouve ce principe de fibres optiques sur les Data Glove 5 et 16 de la compagnie Fifth Dimension Technologies [www.5dt.com].

3.1.2. Capteurs à effet Hall

Les capteurs d'angle à effet hall sont très précis mais assez encombrants. Pour mesurer la flexion des doigts avec de tels capteurs, il faut les intégrer à un exosquelette, c'est-à-dire à une structure rigide articulée venant se fixer sur la main. Les gants numériques à effet hall sont donc précis mais leur poids ne permet pas de les utiliser pour effectuer des mouvements rapides. Ils ont surtout été utilisés pour des applications médicales [Micera 03] et la télé-manipulation [www.sarcos.com]. Parmi ces gants, on peut citer le Dextrous Hand Master (voir Fig. 13), développé à la fin des années 80 et le HumanGlove (voir Fig. 19), vendu jusqu'en 2003 par la société HumanWare [Dipietro 03].

3.1.3. Capteurs résistifs

Les capteurs résistifs sont fondés sur un composant dont la résistance varie en fonction d'une déformation élastique. Pour mesurer un angle de flexion, il faut donc transformer le mouvement de flexion en un mouvement de déformation élastique. De nombreuses solutions existent pour cette transformation [Gentile 92]. L'une d'elle, basée sur l'utilisation d'une encre résistive déposée sur un substrat élastique, a été implémentée dans le PowerGlove de Mattel [Aukstakalnis 92] (voir Fig. 15).

Les solutions brevetées dans [Gentile 92] n'utilisent, pour chaque capteur de flexion, qu'un seul capteur résistif et sont assez sensibles aux variations de la température de la surface des doigts, à la position du gant sur la main ainsi qu'aux phénomènes de dérivation. Dans le Cyberglove (voir Fig. 14), J.F. Kramer utilise des capteurs résistifs possédant chacun deux jauges de contraintes séparées par une couche intermédiaire élastique [Kramer 87]. Lorsque le capteur est plié, la jauge de contrainte placée du côté extérieur de la courbure se retrouve en extension, tandis que la jauge de contrainte placée à l'intérieur se retrouve en compression (voir Fig. 10). Les deux jauges de contrainte sont intégrées dans un pont de Wheatstone ce qui rend le capteur très peu sensible aux variations de température et stable dans le temps. Au final, les capteurs de flexion utilisés dans le Cyberglove sont légers, peu encombrants et très précis. Pour une description complète des caractéristiques de ces capteurs, on peut se reporter à [Kramer 96].

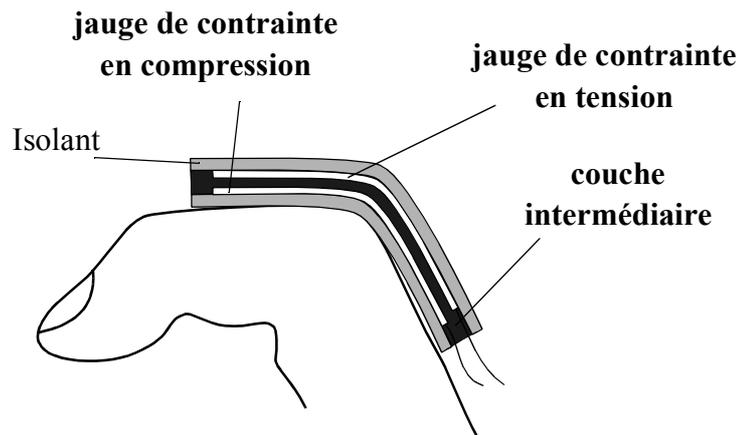


Fig. 10 Capteur de flexion utilisant deux jauges de contraintes utilisés dans le Cyberglove [Kramer 96].

3.1.4. Autres systèmes

Les contractions musculaires et notamment celles des muscles des doigts, sont déclenchées par des potentiels électriques (les potentiels d'action) générés par le système nerveux. L'activité électrique qui en résulte peut être mesurée par des électrodes placées sur la peau. C'est le principe de l'électromyographie ou EMG. Il est alors envisageable de connaître la position de chaque doigt à partir de l'EMG des muscles le contrôlant. De telles mesures auraient l'avantage d'utiliser des électrodes très légères qui pourraient, de surcroît, être placées sur l'avant-bras, où se situe une grande partie des muscles des doigts, laissant la main libre de tout matériel. Mais l'appareil musculaire des doigts est très complexe et il n'existe pas réellement un muscle pour chaque doigt. Les signaux mesurés sont en fait la superposition de multiples potentiels. Toutefois, Gupta et Reddy ont montré qu'il est possible de contrôler en temps réel un doigt virtuel avec une précision assez bonne en utilisant l'EMG globale en un point de l'avant-bras [Gupta 96]. Il existe également des résultats encourageants de reconnaissance de quelques configurations de la main à partir d'EMGs dont les différents potentiels sont séparés par une analyse en ondelettes [Krysztoforki 04]. On est, toutefois, encore loin des précisions obtenues avec les capteurs *classiques*.

Dans [Toney 98], des tubes sont attachés à chacun des doigts de la main, à la façon des tubes lumineux (§ 3.1.1). Ces tubes sont fermés à une extrémité et un capteur de pression est disposé à l'autre extrémité. La flexion du doigt entraîne une diminution du volume du tube et donc une augmentation de la pression qui est mesurée par le capteur. Ce système donne une mesure peu précise de la flexion des doigts et est très encombrant en comparaison des autres solutions techniques.

Des micro-bobines ont été spécialement développées dans [Velten 99] pour la mesure des angles des doigts. Elles possèdent un noyau mobile qui permet de modifier l'inductance de la

bobine et d'en faire des capteurs de déplacement linéaires. Il faut donc transformer le mouvement de flexion en un mouvement de translation. On peut pour cela utiliser une des techniques proposées dans [Gentile 92]. La principale difficulté rencontrée dans la réalisation de ces capteurs est l'obtention d'une bobine de faible dimension mais suffisamment solide pour que le mouvement du noyau ne la détruise pas. De tels capteurs ont été utilisés sur le TUB-SensorGlove de Hofmann [Velten 98] (voir Fig. 16).

Les matériaux piézo-électriques possèdent des propriétés intéressantes au premier abord pour la mesure de la flexion : ils sont peu coûteux, disponibles sous de nombreuses formes (câble co-axial, film, peinture, etc.) et fournissent un signal électrique en réponse à une déformation. Ce sont ces raisons qui ont conduit J. Edmison à fabriquer un gant numérique avec des films piézoélectriques. Mais ce gant permet uniquement de donner une information binaire sur la flexion des doigts de la main et sur le contact de ceux-ci avec une surface rigide [Edmison 02]. Le principal inconvénient de ces capteurs est en effet qu'ils fonctionnent bien pour la mesure de vibrations hautes fréquences mais sont mal adaptés aux changements quasi-statiques des configurations des doigts. Ils n'ont donc pas pu être utilisés pour fournir une mesure des angles des articulations.

3.2. Les systèmes de localisation

Nous présentons maintenant les principaux systèmes de localisation de la main dans l'espace, qui sont décrits dans la littérature. Seul les grands principes mis en oeuvre sont décrits. Pour des présentations plus détaillées, on peut se reporter à [Meyer 92], [Kindratenko 01] et [Fuchs 03].

Les capteurs de localisation peuvent être classés selon le phénomène physique qu'ils utilisent pour effectuer leur mesure. Il y a alors quatre familles principales correspondant à quatre phénomènes physiques : la mécanique, l'électromagnétique, l'acoustique et l'optique [Fuchs 03].

3.2.1. Systèmes mécaniques

Dans cette famille, on trouve en premier lieu les systèmes qui utilisent une liaison mécanique directe entre l'objet devant être localisé et une référence. Cette liaison peut se faire par des ressorts, des systèmes articulés, des systèmes de câbles et poulies ou autres. La position ou l'orientation de l'objet est alors donnée directement et généralement avec une grande précision, mais la liaison mécanique limite fortement la liberté de mouvement de l'utilisateur [Fuchs 03].

D'autres systèmes mécaniques n'utilisent pas une liaison mécanique directe, mais des principes mécaniques permettant de mesurer la vitesse, l'accélération ou l'orientation d'un objet. Notons qu'il n'existe pas, à notre connaissance, de tels systèmes permettant la mesure directe d'une position. Il faut nécessairement effectuer une simple intégration de la vitesse ou

une double intégration de l'accélération. La principale difficulté rencontrée est alors la tendance à la déviation des mesures de position due à cette intégration. Les deux types de capteurs les plus utilisés sont les capteurs inertiels et les inclinomètres. Les capteurs inertiels sont soit des accéléromètres - utilisés pour mesurer la position en 3D par une double intégration du signal - soit des gyroscopes - utilisés pour mesurer l'orientation en 3D par une simple intégration. Notons que les accéléromètres peuvent aussi être utilisés pour mesurer directement, sans intégration, l'orientation relativement à la verticale, donnée par la gravité terrestre. Les inclinomètres utilisent également la gravité pour mesurer directement une orientation par rapport à la verticale, par un système de fluide par exemple.

Parmi les capteurs inertiels présents actuellement sur le marché, on trouve l'InertiaCube3 de Intersense [www.isense.com] qui est un capteur d'orientation 3D. Le même constructeur, propose aussi l'IS-900 qui combine capteurs inertiels et capteurs acoustiques et fournit une très bonne localisation 6D - l'acoustique permettant de limiter la déviation des capteurs inertiels - mais qui est encore trop volumineux pour être utilisé sur les gestes de la main. On a enfin le 3D-BIRD, d'Ascension Technologies [www.ascension-tech.com] qui est un capteur d'orientation 3D.

3.2.2. Systèmes électromagnétiques

L'électromagnétisme est la première technologie à avoir été utilisée pour la localisation de la main. En 1980, R.A. Bolt a utilisé le capteur Polhemus pour le projet Put-That-There [Bolt 80] dans lequel une interface graphique était dirigée par la voix et le geste de la main. Les capteurs de localisation électromagnétiques se sont depuis largement répandus. Ceux actuellement disponibles sur le marché permettent de mesurer très précisément la position et l'orientation de la main dans l'espace.

Ces systèmes sont basés sur le principe général suivant. Une base émettrice crée un champ magnétique environnant dans lequel est plongé l'utilisateur. Des capteurs passifs (inductances) sont attachés sur celui-ci et mesurent l'intensité et l'orientation du champ magnétique, ce qui permet d'obtenir leur propre position et orientation (voir Fig. 11).

Les principaux fabricants de capteurs magnétiques sont Ascension Technologies (microBIRD, miniBIRD, MotionStar, etc.) [www.ascension-tech.com] et Polhemus (Liberty, Patriot, FastTrack) [www.polhemus.com]. Les inconvénient de ces systèmes sont la sensibilité du champ magnétique aux objets métalliques se trouvant à proximité ainsi que leur coût. Leurs principaux avantages résident dans la taille des capteurs et la précision des mesures effectuées.

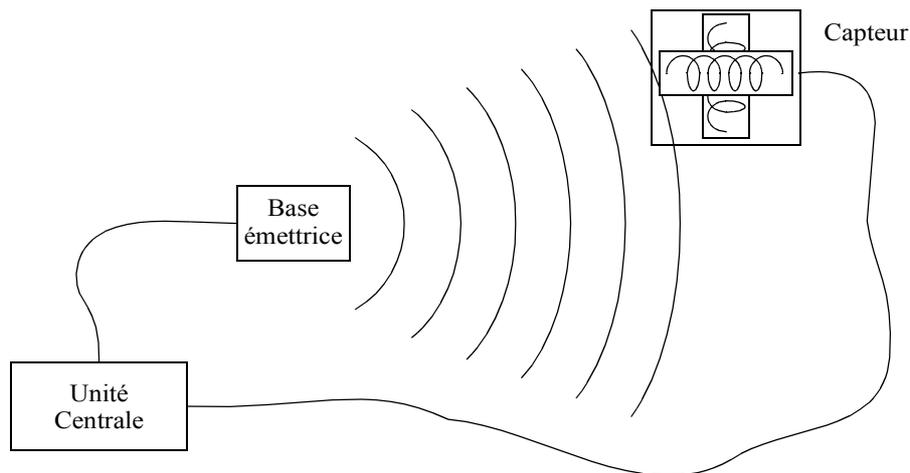


Fig. 11 Principe des systèmes électromagnétiques de localisation 6D.

3.2.3. Systèmes acoustiques

Ces systèmes sont basés sur la mesure du temps de propagation des ondes acoustiques pour la localisation des objets. On peut obtenir une mesure précise de la position en 3D de récepteurs ultrasonores par triangulation. Ces systèmes nécessitent plusieurs sources acoustiques mais aussi plusieurs récepteurs pour une localisation précise. Leur principaux inconvénients sont leur temps de réponse, limité par la vitesse de propagation des ondes acoustiques, ainsi que la nécessité de ne pas obstruer les lignes d'ondes acoustiques. Ces systèmes sont en revanche peu coûteux. Ce type de capteur a été utilisé sur le PowerGlove de Mattel [Aukstakalnis 92]. On les retrouve aujourd'hui associés à des capteurs inertiels dans le système IS-900 d'InterSense [www.isense.com].

3.2.4. Systèmes optiques

Ce sont des systèmes qui utilisent une ou plusieurs sources lumineuses, généralement infrarouges, et un ou plusieurs récepteurs. Il sont à distinguer de ce que nous appellerons les systèmes de vision où aucune source lumineuse n'est employée et où l'on cherche à retrouver la position et l'orientation d'un objet à partir de l'image donnée par une ou plusieurs caméras.

Il existe plusieurs systèmes optiques de localisation que l'on peut classer selon le type de récepteurs utilisés (ponctuels ou caméras), selon le type de sources utilisées et enfin selon que le récepteur soit sur l'objet à localiser ou que ce soit, à l'inverse, la source qui soit mobile. Pour avoir plus de détails sur le fonctionnement de ces systèmes, on peut se reporter à [Fuchs 03]. La localisation de la main par un système optique a été premièrement mise en oeuvre par le MIT au début des années 80 dans le LED Glove. Plus récemment, le P5 glove d'Essential Reality

[www.essentialreality.com] semble posséder un tel système infrarouge pour la localisation de la main dans l'espace.

3.3. Les gants numériques de la littérature

Après ce tour d'horizon des technologies utilisables pour la mesure de la configuration et de la localisation de la main, nous présentons les gants numériques qui sont décrits ou utilisés dans la littérature scientifique ainsi que ceux que l'on trouve actuellement dans le commerce. Nous nous concentrons sur la description des gants mesurant la configuration de la main ou sa position puis nous présentons les gants de commandes binaires qui détectent des contacts entre les doigts, ainsi que les bracelets numériques. Ces derniers ne sont pas, à proprement parler, des gants, mais ils sont intrusifs et leur but est de mesurer la position des doigts de la main. Ils ont donc leur place dans cette section qui finit par une brève présentation des principes utilisés dans les systèmes de vision.

3.3.1. Point de vue historique

Le premier gant numérique a été réalisé en 1977 par T. De Fanti et D. Sandin à partir d'une idée de R. Sayre [DeFanti 77]. Il possédait sept capteurs de flexion à tubes optiques et était utilisé pour le contrôle simultané de plusieurs paramètres continus, remplaçant avantageusement l'utilisation de multiples potentiomètres.

Au début des années 80, le MIT a développé le « LED Glove ». Ce gant était pourvu de diodes électroluminescentes permettant de capturer les mouvements des doigts avec une caméra. Il a été utilisé pour l'animation graphique virtuelle, mais jamais comme interface de contrôle, la technologie n'étant pas assez au point [Sturman 94].

En 1983, G. Grimes a développé le « Digital Data Entry Glove » [Grimes 83]. Ce gant possédait des contacts, des capteurs de flexion (tubes optiques) et des capteurs inertiels. La disposition de ces capteurs ainsi que le câblage du gant étaient conçus de telle manière que le gant fournisse en sortie 84 codes correspondant à autant de postures différentes. Il avait donc la particularité de ne pas nécessiter de partie logicielle : le système de reconnaissance était entièrement câblé.

En 1985, Zimmerman a créé le DataGlove qui possédait cinq à quinze capteurs de flexion selon la version [Zimmerman 87]. Ces capteurs de flexion étaient des tubes lumineux (§ 3.1.1). Il était équipé du capteur magnétique 3SPACE de Polhemus (§ 3.2.2) qui fournissait la localisation 6D de la main. Il a été commercialisé et distribué par VPL Research, à un prix qui a contribué à sa large diffusion, et a été utilisé tant par les centres de recherches que par les fabricants de produits de réalité virtuelle. Il était en effet plus précis, plus rapide et plus confortable que tous les systèmes de mesure de la main existant alors. Ce gant possédait de plus un retour tactile sous forme de vibrations engendrées par des piézo-céramiques. Zimmerman a

aussi développé une version simplifiée du DataGlove, le Z-Glove, pour lequel la localisation se faisait par des capteurs ultrasoniques (§ 3.2.3).

Dans le DataGlove II, introduit en 1987, les tubes lumineux ont été remplacés par des fibres optiques spécialement traitées dans les zones devant mesurer la flexion. Les capteurs de flexion de ce DataGlove II avaient une précision de l'ordre de 5 degrés pour une fréquence d'échantillonnage d'environ 30 Hz [Sturman 94].



Fig. 12 Le DataGlove II de VPL (Tiré de [MacKenzie 95]).

En 1987 également, la société ADL, en collaboration avec l'entreprise Sarcos, a développé un contrôleur pour la main articulée de l'UTAH/MIT [Sturman 92]. Ce contrôleur se présentait sous la forme d'un exosquelette possédant des capteurs à effet hall (§ 3.1.2). Le porteur du projet à ADL fonda alors la compagnie Exos pour vendre une version commerciale de ce contrôleur. Cette version, qui mesurait les angles des cinq doigts de la main, était le Dextrous Hand Master (DHM). Le DHM a ensuite évolué jusqu'à devenir un instrument de mesure de la main donnant 20 degrés de liberté avec une précision très inférieure au degré. Il a été utilisé dans le milieu médical pour l'étude de troubles de la main mais aussi pour la création musicale par T. Machover [Machover 90].

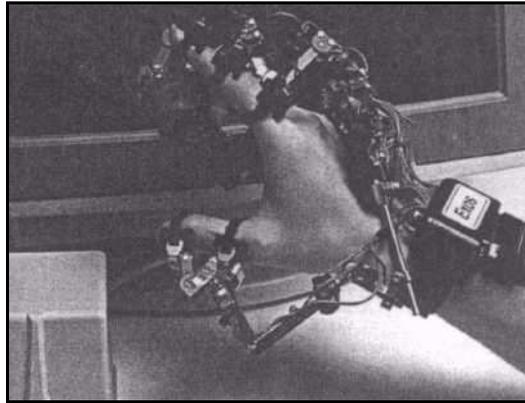


Fig. 13 Le Dextrous Hand Master de Exos (tiré de [Sturman 94]).

La même année, le travail de J.F. Kramer sur la traduction de l'alphabet de la Langue des Signes Américaine (ASL) a débouché sur la conception du Cyberglove [Kramer 87]. Ce gant est toujours d'actualité. Presque vingt ans plus tard, il reste le gant numérique le plus intéressant du marché tant pour la précision, inférieure au degré, que pour sa fiabilité et son confort d'utilisation. Il s'est donc largement répandu malgré un prix assez dissuasif (env. 13000 EUR). Il a d'abord été vendu comme une partie du système de reconnaissance de l'alphabet de l'ASL TalkingGlove [Kramer 96], puis séparément sous la marque Cyberglove aux alentours de 1993. Il est aujourd'hui distribué dans le monde entier, en France par la société Immersion [www.immersion.com]. Les capteurs utilisés sont décrits au paragraphe 3.1.3. Nous utilisons ce gant pour nos travaux et le décrivons plus en détail dans le chapitre 4.



Fig. 14 Le Cyberglove de J.F. Kramer (tiré de [www.immersion.com]).

En 1988 est apparu le PowerGlove, vendu par Mattel comme interface d'entrée pour la console de jeux de Nintendo [Aukstakalnis 92]. Ce gant utilisait de l'encre résistive pour fournir

un signal sur deux bits indiquant la configuration de quatre doigts. Il possédait également un système de localisation acoustique. Bien qu'il ait été créé pour les jeux vidéo, il a en fait beaucoup été utilisé pour des travaux de recherche car il était environ cent fois moins cher que le Cyberglove, ce qui a eu tendance à compenser sa très faible précision. Il a notamment été utilisé pour la reconnaissance de la Langue des Signes Australienne par W. Kadous qui a obtenu de bons résultats. Il n'a en revanche pas pu reconnaître l'alphabet de cette même langue du fait de l'absence de capteur sur l'auriculaire [Kadous 95].



Fig. 15 Le PowerGlove de Mattel.

3.3.2. Gants plus récents

L'institut pour les systèmes temps réels et la robotique de l'université de Berlin a développé le TUB-SensorGlove, breveté en 1995. Ce gant mesure les angles des articulations de la main ainsi que les forces de pression s'appliquant dans les gestes de préhension. Il est également équipé d'accéléromètres tri-axes miniaturisés qui ne sont pas utilisés pour mesurer sa position dans l'espace mais pour pouvoir séparer plus efficacement les différents gestes. Les capteurs de flexion sont les micro-bobines développées par Velten [Velten 99]. Selon ses inventeurs, la précision de ce gant est supérieure à celle du VPL DataGlove (toujours en vente à l'époque). Ce gant n'a cependant pas été commercialisé.

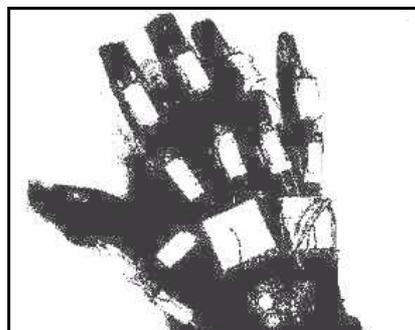


Fig. 16 Schéma du TUB SensorGlove (tiré de [Velten 98]).

L'Acceleration Sensing Glove possède cinq accéléromètres disposés à l'extrémité de chacun des doigts, plus un sur le dos de la main [Perng 99]. En mesurant la direction de la gravité terrestre, ils permettent de déduire l'orientation de la main et des doigts en 2D. vingt huit postures peuvent être reconnues efficacement à partir de ces mesures. Par contre, le fait que les mesures soient faites en référence à la gravité implique que certaines postures, pourtant très différentes visuellement, ne peuvent pas être différenciées par le gant. L'avantage de cette approche réside dans l'espoir de pouvoir miniaturiser les accéléromètres, de les associer à une antenne, un système de traitement du signal et une source d'énergie, le tout dans un volume de l'ordre du mm³. Si cela était réalisé, l'Acceleration Sensing Glove se résumerait à quelques pastilles minuscules que l'on collerait sur la main : un gant instrumenté qui ne serait pas plus gênant que du vernis à ongle...



Fig. 17 L'Acceleration Sensing Glove (tiré de [Perng 99]).

La société Fifth Dimension Technologies [www.5dt.com] propose plusieurs versions de son 5DT Data Glove. Ce gant est le descendant du DataGlove de VPL : il possède des capteurs de flexion à fibres optiques au nombre de cinq ou quatorze selon les versions. Un capteur d'orientation peut, de plus, être monté sur le dos du gant. La mesure par fibres optiques le rend moins précis que le Cyberglove mais il est également moins cher (env 500 EUR pour la version 5 capteurs et 4000 EUR pour la version 14 capteurs). Il est disponible en version sans fil. Le gant en lycra est confortable mais il n'est pas prévu d'aération comme c'est le cas pour le Cyberglove. Néanmoins ce gant est le digne successeur du VPL DataGlove, il offre une résolution raisonnable, une fréquence d'échantillonnage de 100 Hz permettant de capturer les mouvements rapides et la version 14 capteurs permet au final d'avoir une mesure correcte de la configuration de la main. Il a donc beaucoup été utilisé et notamment comme interface d'entrée pour des systèmes de réalité virtuelle [Grant 98].



Fig. 18 Le 5DT DataGlove 16-W, 14 capteurs, version sans fil (Tiré de [www.5dt.com]).

Le HumanGlove de la société HumanWare utilise des capteurs à effet hall intégrés à un exosquelette [Dipietro 03]. Il est bien moins encombrant que le Dextrous Hand Master bien qu'il mesure lui aussi 20 degrés de liberté. Il est montré, dans [Dipietro 03], que sa précision est à peu près équivalente à celle du Data Glove de la société Fifth Dimension. Il a été utilisé dans [Micera 03] pour l'étude de problèmes orthopédiques. Il semble ne plus être disponible actuellement.



Fig. 19 Le HumanGlove de la société HumanWare (tiré de [Dipietro 03]).

Oda [Oda 98] a fixé 20 jauges de contraintes sur un gant chirurgical. Il a ainsi obtenu un gant numérique aux performances de l'ordre de celles du DataGlove de 5DT mais pouvant, d'après l'auteur, être utilisé en chirurgie.

La société Essential Reality [www.essentialreality.com] a récemment commercialisé un gant très bon marché (env 50 EUR) aux performances très honorables : le « P5 Glove ». Il possède cinq capteurs de flexion dont la technologie n'est pas communiquée par le fabricant. Ces capteurs ont une résolution de 0.5 degrés, mais ils sont fixés aux doigts par un système très basique d'anneaux, ce qui abaisse probablement la précision des mesures de façon sensible. Ce gant est également équipé d'un système de localisation 6D à infrarouge. Le principal avantage

de ce gant réside dans son prix qui l'a vite rendu très populaire. On l'utilise en tant qu'interface d'entrée de systèmes de réalité virtuelle, de jeux vidéos, pour remplacer la souris ou encore en tant que contrôleur midi.



Fig. 20 Le P5 Glove d'Essential Reality (tiré de [www.essentialreality.com]).

Un gant numérique très intéressant est celui de J.L. Hernandez-Rebollar [Hernandez 02.a] qui est basé sur le même principe que l'Acceleration Sensing Glove de J.K. Perng décrit précédemment. J.L. Hernandez-Rebollar s'est attaqué aux deux principaux problèmes des gants numériques : leur prix et leur encombrement. Son Accele Glove est constitué de cinq accéléromètres bi-axes miniaturisés. Il est donc très léger (voir Fig. 21) et peut être fabriqué à un coût raisonnable. A partir des données de ces capteurs, on peut extraire les angles de flexion des doigts et celui de l'orientation de la main avec une précision comparable au DataGlove de 5DT. Hernandez-Rebollar l'utilise pour reconnaître efficacement l'alphabet de la Langue des Signes Américaine [Hernandez 02.b]. Ce gant est en cours de commercialisation.



Fig. 21 L'Accele Glove d'Hernandez-Rebollar (Tiré de [Hernandez 02.a]).

3.3.3. Tableau récapitulatif

Gant	Année	Référence	Type de mesures	Type de capteurs	Nombre de capteurs	Précision	Echantillonnage
Sayre	1977	[DeFanti 77]	flexion	tubes optiques	7		
Digital data entry glove	1983	[Grimes 83]	contacts	contacts	15		
			flexion	tubes optiques	4		
			mouvement	accéléromètres	2		
			orientation	inclinomètres	2		
DataGlove	1985	[Zimmerman 87]	flexion	tubes optiques	5 à 15 selon la version		
			position/orientation	magnétique	1 capteur 6D Polhemus 3SPACE		
DataGlove II	1987	[Sturman 94]	flexion	fibres optiques	10	5-10 degrés	30 Hz
			position/orientation	magnétique	1 capteur 6D Polhemus 3SPACE		
Dextrous Hand Master	1987	[Sturman 92]	flexion	effet hall	20	8 bits (< 1 degré)	200 Hz
CyberGlove	1987	[Kramer 96]	flexion	jauges de contrainte	18 ou 22 selon la version	< 1 degré	100 Hz
PowerGlove	1989	[Sturman 94]	flexion	encre résistive	4	2 bits	
			position/orientation	ultrasons	1 capteur 4D (position + 1 orientation)		
Data Glove 5 et 16	1995	[www.5dt.com]	flexion	fibres optiques	5 ou 14 selon la version	5-10 degrés	> 75 Hz
TUB-SensorGlove	1995	[Velten 99]	flexion	micro-bobines	≈ 12	< 1 degré	
			mouvement	accéléromètres	1 (3 axes)		
HumanGlove	1998	[Dipietro 03]	flexion	effet hall	20	< 1 degré	
Acceleration Sensing Glove	1999	[Perng 99]	accélération	accéléromètres bi-axes	6		
P5 Glove	2001	[www.essentialreality.com]	flexion	?	5	< 1 degré	
			position/orientation	optique (magnétique?)			
Accele Glove	2002	[Hernandez 02b]	accélération	accéléromètres bi-axes	6		

3.3.4. Gants de commande binaires

Certains gants sont appelés gants de commande binaires dans [Fuchs 03] car ils ne mesurent pas les angles des articulations mais envoient des signaux binaires associées à des configurations de la main. Le plus répandu de ce type de gant est incontestablement le Pinch Glove des laboratoires FakeSpace [www.fakespacelabs.com]. Celui-ci possède des pastilles conductrices au bout de chacun des doigts. Lorsque plusieurs pastilles sont réunies, un signal est envoyé indiquant quels doigts sont en contact. L'ensemble des configurations pouvant être ainsi reconnues est très important, bien plus grand que l'ensemble de celles pouvant être retenues par une personne non entraînée. Le Pinch Glove a été utilisé principalement comme interface d'entrée pour des systèmes de réalité virtuelle [Bowman 02].



Fig. 22 Le Pinch Glove de FakeSpace.

Le Chording Glove de Rosenberg s'inspire des claviers à accord (chord keyboard) sur lesquels l'entrée d'un caractère se fait par l'appui simultané de plusieurs touches, tel un accord de piano, ce qui réduit le nombre de touches nécessaires [Rosenberg 99]. L'idée du Chording Glove est de mettre les touches sur un gant, à l'extrémité des doigts. Ce gant est donc utilisé pour remplacer un clavier traditionnel.

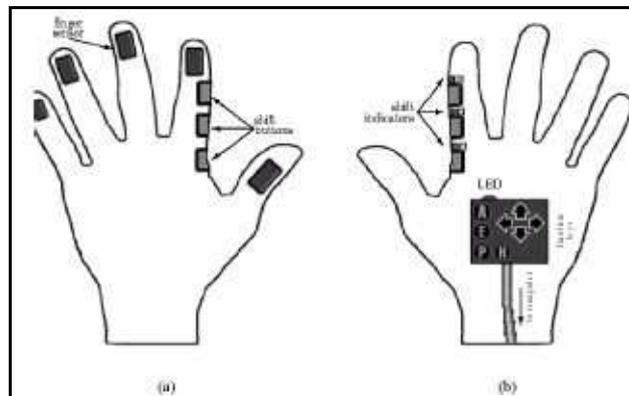


Fig. 23 Schéma du Chording Glove (Tiré de [Rosenberg 99]).

3.3.5. Bracelets numériques

On appelle bracelets numériques, des systèmes qui se limitent à un dispositif se fixant autour du poignet. Ils ne permettent pas d'identifier la configuration de la main de la même façon que les gants numériques décrits jusqu'ici. Ils restent néanmoins des interfaces d'entrée utilisant la main dans son ensemble (les *Whole hand Inputs* définies dans [Sturman 92]) et permettent de reconnaître un certain nombre de configurations de la main. Ils sont, en outre, très intéressants car ils ne sont pas plus contraignants à porter qu'une montre. Les capteurs à EMG décrits au paragraphe 3.1.4 pourraient rentrer dans cette catégorie mais il n'existe aujourd'hui aucun système fini utilisant de tels capteurs.

Le GestureWrist de J. Rekimoto est clairement un bracelet numérique. Il mesure les déformations du poignet. Cette mesure permet de reconnaître certaines postures simples. Il est associé à un accéléromètre qui permet de reconnaître le mouvement de la main [Rekimoto 01].

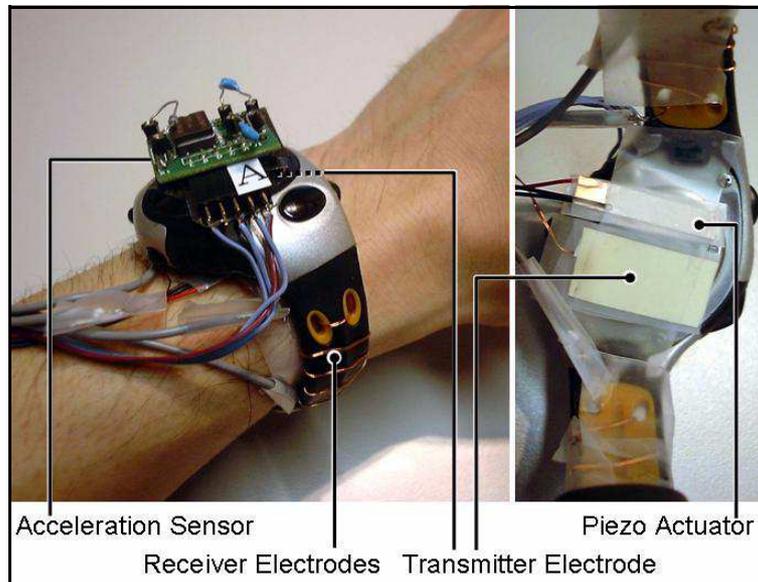


Fig. 24 Le GestureWrist de J. Rekimoto (tiré de [Rekimoto 01]).

Le LightGlove de la société du même nom [www.lightglove.com] est également un bracelet numérique [Howard 01]. Il se réduit à un petit boîtier qui se porte sous le poignet. Ce boîtier possède des émetteurs et des récepteurs de lumière dans le domaine visible ou infrarouge. Ceux-ci scannent la surface se trouvant sous la paume de la main, détectent si un doigt s'y trouve, générant alors une commande binaire. Ils peuvent également détecter le mouvement des doigts dans cette surface. Le très faible encombrement de ce dispositif ainsi que les nombreuses possibilités de commande en font une interface d'entrée possible pour diverses applications. Le LightGlove est typiquement une interface qui se trouve à la frontière entre les gants numériques et les systèmes de vision.



Fig. 25 Contrôle d'un assistant personnel avec le LightGlove.

3.4. Systèmes non intrusifs : systèmes de vision

Les interfaces non intrusives laissent la main totalement libre, car exempte de tout matériel. Elles sont donc plus proches de la façon dont nous, humains, captions les gestes et c'est pourquoi elles sont souvent qualifiées de naturelles. Elles possèdent néanmoins plusieurs limitations, la principale étant qu'elles nécessitent beaucoup plus de traitements que les interfaces intrusives. Le but d'une interface gestuelle est en effet d'extraire des caractéristiques relatives au geste manuel et sur lesquelles des traitements peuvent être effectués. Dans le cas des interfaces intrusives, on peut souvent utiliser comme caractéristiques les données brutes fournies par les capteurs ou, du moins, une fonction simple de ces données brutes. Le problème de l'extraction des caractéristiques du geste dans le cas des interfaces non intrusives est souvent beaucoup plus complexe.

Nous ne présentons ici que les systèmes de vision qui forment le principal type d'interfaces non intrusives. Notons tout de même qu'il existe des prototypes d'interfaces non intrusives qui ne sont pas basées sur la vision comme par exemple les capteurs utilisant des interférences électromagnétiques présentés dans [Holzrichter 01]. On trouvera des revues complètes des systèmes de reconnaissances des gestes à base de vision dans [Huang 95], [Kohler 98] et [Wu 99].

Le problème de l'acquisition des caractéristiques du geste manuel pour les systèmes de vision se décompose en trois étapes :

- **Acquisition de l'image :** Cette acquisition peut se faire par une ou plusieurs caméras. Le dernier cas permettant, par exemple, de limiter les problèmes d'occultation de la main ou encore d'obtenir une image stéréo.
- **Suivi/segmentation :** Une fois l'image obtenue, il faut localiser la main et l'isoler de l'arrière plan ou des autres objets présents dans la scène.
- **Extraction des caractéristiques :** Cette étape dépend des caractéristiques que l'on choisit de mesurer.

On peut alors classer les systèmes à base de vision dans trois catégories selon la technique utilisée pour résoudre l'un et/ou l'autre problème(s) que sont le suivi/segmentation et l'extraction des caractéristiques [Martin 00].

- **Systèmes utilisant des gants de couleurs ou des gants marqués.** Ces systèmes sont en fait intrusifs puisqu'ils nécessitent le port d'un gant comportant par exemple différentes parties colorées et permettant de simplifier l'étape de suivi/segmentation de la main.
- **Systèmes basés sur un modèle de la main.** Ces systèmes utilisent un modèle 3D de la main et tentent de faire correspondre ce modèle avec l'image observée. Deux approches sont possibles : l'estimation du modèle puis sa mise en corres-

pondance avec l'image ou l'extraction de caractéristiques puis l'estimation du modèle. Ces systèmes permettent d'évaluer les angles des articulations de la main. Ils fournissent donc une information très complète sur le geste effectué. Mais l'évaluation des paramètres du modèle est généralement difficile et coûteuse en temps de calcul.

- **Systèmes basés sur l'apparence visuelle.** Ces systèmes n'essaient pas d'identifier les angles des articulations de la main : ils se contentent de mesurer certaines caractéristiques propres à l'image observée (position du centre de gravité, surface de la main, etc.). Ces caractéristiques ont peu de signification intrinsèque mais permettent d'opérer une classification de différentes images de la main et ainsi de discriminer des gestes statiques ou dynamiques assez efficacement.

4. Conclusion

Le geste manuel est un vaste sujet d'étude et on n'en attendrait pas moins d'une partie du corps possédant plus de la moitié des degrés de liberté de nos articulations, qui nous sert à chaque instant et peut remplacer la parole tout en permettant d'écrire, de couper du bois, de faire de l'alpinisme, etc. Dans la première partie de ce chapitre, nous avons vu les types de fonctions de la main qui est donc simultanément un organe sensoriel, moteur et communicatif. Nous avons également présenté brièvement les différents types de langages utilisant le canal gestuel, illustrant là encore une grande diversité des utilisations possibles de la main dans une fonction particulière.

La deuxième partie de ce chapitre a montré que les systèmes de mesure du geste manuel sont très variés et ont fait l'objet de nombreuses études, aboutissant à la commercialisation de produits performants. C'est d'ailleurs souvent le manque d'applications permettant de prendre en compte toute la potentialité du geste humain qui fait défaut à la large diffusion de ces systèmes de mesure, d'où l'importance des travaux sur l'interaction homme-machine au sens large utilisant le canal gestuel.

Ce chapitre nous a également permis de préciser ce que nous entendons par les termes « *geste manuel* ». Nous avons décidé de définir le geste d'un point de vue de la mesure, c'est-à-dire comme un signal particulier variant dans le temps. Contrairement à la parole qui peut se définir très exactement comme un signal acoustique, il n'existe pas de modèle physique universel du geste sur lequel s'appuyer pour le définir formellement. Par contre, deux composantes principales peuvent être identifiées, il s'agit de la configuration de la main et de sa localisation. La composition de chacune de ces composantes dépend ensuite du système de mesure utilisé. Si on utilise un PowerGlove pour mesurer le geste de la main, la configuration sera un signal de dimension quatre, chaque composante étant codée sur deux bits. Si par contre on utilise un Cyberglove pour effectuer cette mesure, la configuration devient un signal de dimension dix-huit ou vingt et un, codé sur huit bits (voir tableau récapitulatif, § 3.3.3).

Cette définition a l'avantage d'être indépendante du type de geste étudié. Mais son inconvénient est d'être une définition bas niveau, elle dépend donc du système de mesure et les valeurs fournies par les gants numériques ne sont pas interprétables directement. Dans ce manuscrit, nous proposons d'appliquer un formalisme utilisé dans les capteurs symboliques flous pour la mesure de la configuration de la main. Nous ne nous intéressons pas à sa localisation.

On obtiendra alors un nouveau type de mesure de la configuration de la main, basé sur l'utilisation de symboles plutôt que de nombres et donc plus facilement interprétable et indépendant du système de mesure utilisé. Cette mesure symbolique du geste manuel sera ensuite utilisée pour la reconnaissance de signes ainsi que pour la commande d'un robot mobile. La reconnaissance de signes, en particulier, a fait l'objet d'un nombre très important de travaux qui sont présentés dans le chapitre suivant.



www.phdcomics.com

CHAPITRE 2

*Les différentes approches de la
reconnaissance de signes*

1. Introduction

Les principales méthodes de reconnaissance de signes utilisées dans la littérature vont être exposées dans ce chapitre. Insistons sur le fait que nous considérons dans cette thèse des signes ne faisant intervenir que la configuration de la main. Nous nous intéressons donc ici essentiellement aux travaux portant sur la reconnaissance de configurations de la main, le problème de la reconnaissance de gestes globaux ou de langages gestuels que sont les langues des signes sortant du cadre de nos travaux.

Dans un premier temps, nous présentons le cadre commun à tous les problèmes de reconnaissance et situons le cas de la reconnaissance de signes. Nous nous intéressons ensuite plus particulièrement au problème de classification des gestes, en présentant tout d'abord la règle de Bayes sur laquelle sont basées beaucoup de classifieurs. Cette présentation nous permettra également d'introduire quelques notations.

Dans un deuxième temps, nous présentons successivement les principaux types de classifieurs utilisés pour la reconnaissance de signes statiques et dynamiques. Rappelons que nous avons défini ces deux types de signes respectivement comme des classes de postures - n'ayant donc pas de durée temporelle - et des classes de gestes ayant une durée dans le temps. La reconnaissance de signes dynamiques est plus difficile que celle de signes statiques car, à la variabilité de la configuration de la main, s'ajoute une variabilité de la dynamique des gestes. Les algorithmes utilisés en reconnaissance des signes statiques ne peuvent donc pas être appliqués tels quels à la reconnaissance de signes dynamiques. Nous avons alors séparé en deux parties distinctes la présentation des systèmes de reconnaissance de signes statiques et dynamiques. Un tableau se trouvant à la fin de chacune de ces parties récapitule les caractéristiques ainsi que les performances des systèmes cités dans le texte. Précisons pour finir que cette présentation des systèmes de reconnaissance de signes n'a pas pu être exhaustive tant le nombre et la diversité des approches sont importants. Nous nous sommes concentrés sur la présentation des approches les plus importantes en terme de résultats et de nombre de travaux les utilisant.

2. Reconnaissance de signes - généralités

2.1. Structure d'un système de reconnaissance de signes

Tout problème de reconnaissance se décompose en trois étapes : l'acquisition des données, leur représentation et leur classification (voir Fig. 26). Dans le cas de la reconnaissance de signes, la première étape consiste en la mesure du geste manuel et a été traitée dans le chapitre 1. Les données issues des différents capteurs ou caméras sont dites brutes. L'étape de représentation consiste à mettre ces données sous une forme facilitant leur classification. Les performances de la classification dépendent donc fortement du choix de l'espace de représentation. Celui-ci doit remplir deux objectifs contradictoires. D'un côté, la représentation des données doit être compacte, pour limiter le nombre de paramètres à apprendre et ainsi faciliter l'apprentissage de l'algorithme de classification. De l'autre, les différentes classes à reconnaître doivent être bien séparées, ce qui incite à choisir un espace de dimension élevée. D'autre part, l'étape de représentation dépend du système d'acquisition utilisé. Dans le cas des gants numériques, les données brutes sont sous une forme déjà relativement compacte et discriminante et la représentation se réduit souvent à une simple normalisation ou à une discrétisation de l'espace brut en différentes zones [Braffort 96]. Si c'est, au contraire, un système de vision qui est utilisé pour l'acquisition des données, l'espace des données brutes est de dimension très élevée et l'étape de représentation transforme largement ces données pour ne garder que les informations pertinentes.

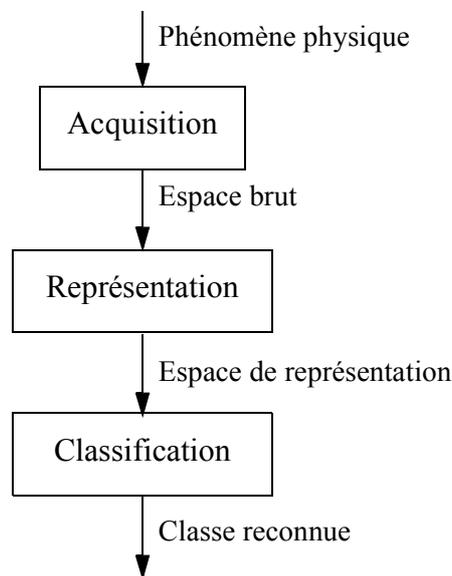


Fig. 26 Les trois étapes d'un système de reconnaissance.

La reconnaissance de signes s'inscrit donc dans le cadre général de la classification où l'on rencontre trois types de problèmes [Dreyfus 04].

- **Classification déterministe.** Dans le cas le plus simple, les classes sont clairement séparées. Il existe des descripteurs des données tels qu'il est facile de concevoir un automate mettant en oeuvre des règles simples pour la classification. Ces règles traduisent une connaissance acquise par une analyse du problème. Il n'est pas nécessaire d'avoir recours à des méthodes statistiques permettant d'acquérir cette connaissance. Ce type de classification se retrouve, par exemple, pour l'identification des pièces de monnaie dans les distributeurs automatiques. La variabilité des objets, due au procédé de fabrication des pièces, ainsi que les erreurs de mesures sont faibles devant l'espacement des classes. Celles-ci sont alors de « petits parallélépipèdes » de l'espace de représentation, tout le reste constituant la classe de rejet. Il peut éventuellement y avoir rejet si une pièce est fautive ou très abîmée mais il ne peut théoriquement pas y avoir d'erreur de classification.
- **Classification qualitative.** Dans ce deuxième cas, les descripteurs ne sont pas connus ou pas mesurables numériquement. Par exemple, si on veut classer des voitures selon leur confort, il est, d'une part, difficile d'identifier tous les descripteurs pertinents pour cette classification et, d'autre part, certains des descripteurs choisis, comme par exemple la qualité acoustique ou la beauté de l'habitacle, peuvent ne pas être mesurables numériquement mais uniquement descriptibles qualitativement ou linguistiquement. Dans ce type de problèmes, il n'existe pas forcément de classe de rejet (on a toujours une idée sur la qualité d'une voiture) et la frontière entre les différentes classes est souvent floue. Les méthodes de classifications floues sont, selon [Dreyfus 04], adaptées à ce genre de problèmes.
- **Classification statistique.** Dans ce dernier cas, on dispose de mesures numériques des descripteurs, mais les données sont très variables et/ou les mesures très imprécises. On ne peut alors pas éliminer le risque d'une erreur de classification ou d'un rejet à tort. Le choix des frontières des différentes classes est fait de façon à minimiser ces risques. On est alors typiquement dans un schéma statistique.

La reconnaissance des signes fait clairement partie de la troisième catégorie : on dispose de mesures numériques décrivant précisément les gestes à classer mais la variabilité de la réalisation des signes est telle qu'on ne peut éliminer totalement le risque de se tromper.

On s'intéresse ici plus particulièrement aux différentes approches utilisées pour l'étape de classification.

2.2. Notations - règle de Bayes

Dans la suite de ce chapitre, on notera x une observation devant être classifiée. Cette observation est issue du processus de représentation (voir Fig. 26). Elle est généralement constituée de p composantes ou attributs, $x = (x_1, x_2, \dots, x_p)$. On note X_i l'ensemble des valeurs que peut prendre la composante x_i . Cette dernière est le plus souvent de nature numérique, X_i étant un intervalle de \mathbb{R} . Elle peut aussi être catégorielle ou symbolique, X_i étant un ensemble fini de catégories ou symboles. L'espace de représentation est le produit cartésien des ensembles X_i soit $X = X_1 \times X_2 \times \dots \times X_p$.

Un classifieur doit être capable de dire, pour toute nouvelle observation x , si elle appartient ou non à l'une des n classes à reconnaître. On note C l'ensemble de ces classes, $C = \{c_1, c_2, \dots, c_n\}$. Un classifieur se définit alors formellement par une fonction δ , appelée règle de décision, qui va de l'espace de représentation X vers un ensemble fini D de $n+1$ décisions possibles.

$$\begin{aligned} \delta : X &\rightarrow D = \{d_0, d_1, \dots, d_n\}, \\ x &\mapsto \delta(x), \end{aligned} \quad (2)$$

avec :

- $\delta(x) = d_i$ et $i = 1, 2, \dots, n$ qui signifie que la classe c_i est choisie pour l'observation x .
- $\delta(x) = d_0$ qui signifie qu'aucune classe n'est choisie (rejet).

Dans le cas de la classification statistique, telle qu'elle est définie au paragraphe 2.1, toute décision est associée à un risque de se tromper. Il faut minimiser ce risque et, pour cela, le quantifier. On associe alors un coût $C(d_i/c_j)$ au fait de prendre la décision d_i alors que la classe réelle de l'observation est c_j . Une distribution de coûts classique est la suivante :

$$\begin{aligned} C(d_j/c_j) &= 0 \text{ (bonne classification),} \\ C(d_i/c_j) &= C_e \text{ si } i > 0 \text{ et } i \neq j \text{ (erreur de classification),} \\ C(d_0/c_j) &= C_r \text{ (rejet).} \end{aligned} \quad (3)$$

Le coût d'une décision $\delta(x) = d_i$ dépend de la classe réelle de l'observation x , que l'on ne connaît pas. On peut, au mieux, espérer connaître, pour chaque classe c_j , sa probabilité $P(c_j/x)$ d'être effectivement la classe de l'observation x . $P(c_j/x)$ est appelée la probabilité a

a posteriori de la classe j . Le coût est, dans ce cas, une variable aléatoire et le risque $R(d_i/x)$ est l'espérance de cette variable aléatoire :

$$R(d_i/x) = \sum_{j=1}^n C(d_i/c_j) \times P(c_j/x). \quad (4)$$

La règle de décision de Bayes consiste à choisir la décision minimisant le risque $R(d_i/x)$:

$$\delta^B(x) = \underset{d_i \in D}{\operatorname{argmin}} R(d_i/x). \quad (5)$$

Avec la distribution de coûts (3), la règle de décision de Bayes revient à choisir la classe c_i qui a la plus forte probabilité *a posteriori* si celle-ci est supérieure à $1 - C_r/C_e$. On choisira donc la classe de rejet si toutes les probabilités *a posteriori* sont inférieures à $1 - C_r/C_e$, ce qui ne peut se produire que si le coût de rejet C_r est inférieur au coût d'erreur de classification C_e .

La règle de Bayes est la meilleure règle possible au sens où elle minimise l'espérance du coût [Devroye 96]. Mais les probabilités *a posteriori* ne sont généralement pas connues. La seule information dont on dispose au départ est, pour chaque classe, un ensemble d'exemples de la classe. Dans le cadre de la reconnaissance de signes, cet *ensemble d'apprentissage* est obtenu en enregistrant des exemples des différents signes effectués par une ou plusieurs personnes. La classification s'inscrit alors dans le cadre plus général de l'apprentissage supervisé¹ : il faut apprendre la règle de décision à partir des exemples dont on dispose.

Une première approche pour cet apprentissage est d'estimer les probabilités *a posteriori* afin d'en déduire une règle de décision s'approchant de la règle de Bayes. On se ramène alors à un problème d'estimation d'une densité de probabilité [Lengellé 02]. C'est le cas des k plus proches voisins ou des classifieurs de Bayes. On peut aussi chercher à définir plus directement la règle de décision, sans passer de manière explicite par les probabilités *a posteriori*, mais en minimisant un critère tel que l'erreur (quadratique ou autre) entre la règle apprise appliquée à l'ensemble d'apprentissage et la classification véritable de cet ensemble. C'est le cas des arbres de décisions, des réseaux de neurones, etc. On distingue les approches non paramétriques des approches paramétriques. Toutes ces méthodes font des hypothèses quant à la forme de la fonction apprise (probabilités *a posteriori* ou règle de décision ou autre). Dans les méthodes non paramétriques, la valeur de cette fonction dépend de tout ou partie des points d'apprentissage. Ces méthodes incluent les k plus proches voisins et les classifieurs à noyaux. Il faut alors, pour

1. En opposition à l'apprentissage non supervisé où des données sont fournies au système d'apprentissage sans information sur leur appartenance à une classe. Il faut alors non seulement apprendre à séparer les différentes classes, mais surtout à les identifier à partir de la répartition des données. On ne parle pas, dans ce cas, de classification mais plutôt de regroupement de données ou clustering.

être capable de classifier n'importe quelle entrée, conserver l'ensemble de ces points en mémoire. Dans les méthodes paramétriques, la fonction apprise ne dépend au final que d'un certain nombre de paramètres. L'apprentissage consiste alors à déterminer ces paramètres. On trouve parmi ces méthodes les classifieurs Bayesiens, les arbres de décision, les bases de règles, les réseaux de neurones, les machines à vecteurs supports ou encore les algorithmes génétiques. Pour une introduction plus détaillée à la théorie de la classification, on pourra se reporter à [Devroye 96] et [Duda 01].

Nous présentons maintenant les principaux classifieurs qui ont été utilisés pour la reconnaissance de signes statiques puis pour la reconnaissance de signes dynamiques. A la fin de chacune de ces deux parties, on trouvera un tableau récapitulatif des performances des différents systèmes de reconnaissance cités dans le texte.

3. Reconnaissance de signes statiques

Un signe statique a été défini au chapitre 1 comme une classe de postures de la main. Si les gestes à reconnaître sont présentés *un à un* au système de reconnaissance, les trois étapes de la figure 26 sont suffisantes. On parle alors de reconnaissance de signes isolés. Mais une telle situation n'est pas naturelle puisque le geste de la main est un processus continu. Un système de reconnaissance ne peut donc être utilisé en situation réelle que s'il intègre une étape de segmentation permettant d'extraire les postures à reconnaître du geste continu. On parle, dans ce cas, de reconnaissance de signes connectés. Cette étape de segmentation peut se trouver à différents niveaux du processus de reconnaissance et faire appel à des calculs plus ou moins complexes. Elle peut, par exemple, se situer après l'étape d'acquisition ou de représentation : l'examen des données peut révéler des points caractéristiques du début ou de la fin d'un signe, comme une pause [Takahashi 91], un pic d'accélération [Sawada 98], un minimum de vitesse [Kramer 96], etc. La segmentation peut également se faire après l'étape de classification, par exemple par un système de seuillage sur les taux de reconnaissance [Murakami 91].

Dans cette partie, nous présentons les différents systèmes de reconnaissance de signes statiques présentés dans la littérature en les regroupant selon la méthode employée pour l'étape de classification. On trouvera en fin de partie un tableau récapitulatif des caractéristiques des systèmes présentés.

3.1. k -plus proches voisins (k -PPV)

Ces classifieurs nécessitent qu'une distance soit définie sur l'espace de représentation. On attribue alors à une observation x la classe ayant le plus de représentants parmi les k points d'apprentissages les plus proches. Remarquons que la plupart des classifieurs utilisent à un moment donné une stratégie similaire. Les k -PPV sont des classifieurs pour lesquels la distance utilisée est choisie a priori et non apprise à partir de l'ensemble d'apprentissage. Les performances de ces classifieurs dépendent du choix de l'espace de représentation, de l'opérateur de distance utilisé (distance euclidienne, max, city-block, Mahalanobis, etc.) ainsi que du choix de k . La figure 27 illustre la différence de comportement que peut avoir un tel classifieur selon le choix de k . Les k -PPVs font certainement partie des classifieurs les plus simples à implémenter, ce qui leur a valu une grande popularité dans le cadre de la classification. La phase d'apprentissage est pratiquement inexistante : il suffit de conserver en mémoire l'ensemble des points d'apprentissage. Ils sont donc intéressants pour la reconnaissance d'un faible nombre de signes. Par exemple, Eisenstein les a utilisés pour reconnaître 10 signes statiques [Eisenstein 01] (voir Fig. 33).

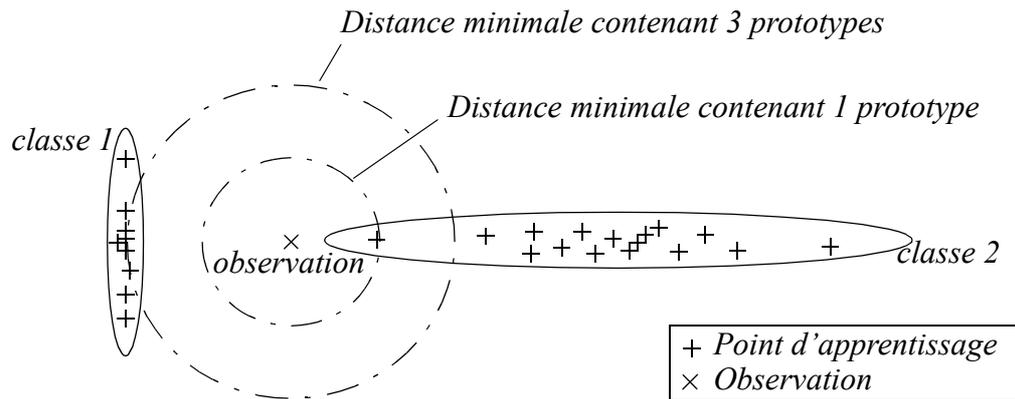


Fig. 27 Principe de la classification par un k -PPV. La distance euclidienne est utilisée. Un classifieur 3-PPV classera l'observation dans la classe 1 et un 1-PPV dans la classe 2.

Les k -PPVs peuvent aussi être utilisés dans une version paramétrique où chaque classe est représentée par un ensemble restreint de prototypes et non plus par l'ensemble des points d'apprentissage. Les performances du classifieur dépendent alors grandement du choix des prototypes des classes [Huang 02]. Dans son étude sur les méthodes de regroupement de postures, Eisenstein utilise de tels k -PPVs [Eisenstein 01]. Il arrive à obtenir de bons résultats de classification mais le nombre de signes à reconnaître est faible et les prototypes des classes sont appris par des méthodes de regroupement, l'apprentissage n'est donc pas supervisé. Sawada reconnaît efficacement 14 signes statiques avec un 1-PPV et une distance dite *normalisée* [Sawada 98].

3.2. Classifieur de Bayes

Un classifieur de Bayes utilise la formule de Bayes pour calculer les probabilités a posteriori :

$$P(c_i/x) = \frac{p(x/c_i) \times P(c_i)}{\sum_{j=1}^n p(x/c_j) \times P(c_j)}, \quad (6)$$

avec :

- $P(c_i)$ = probabilité *a priori* de la classe c_i .
- $p(x/c_i)$ = densité de probabilité de l'observation x conditionnelle à la classe c_i .
On l'appelle également la *vraisemblance* de la classe c_i .
- $P(c_i/x)$ = probabilité *a posteriori* d'appartenance de l'observation x à la classe c_i .

Dans son étude sur la reconnaissance de l'alphabet de la Langue des Signes Américaine, Kramer compare les performances de deux classifieurs de Bayes, d'un réseau de neurones et d'un arbre de décision [Kramer 96]. Le premier classifieur de Bayes - Bayes(W) - utilise des vraisemblances multinormales et l'autre - Bayes(I) - est un classifieur dont la structure est adaptée aux signes à reconnaître et utilise des vraisemblances multinormales réduites. Des deux classifieurs, c'est Bayes(W) qui donne le meilleur taux de reconnaissance pour les 26 lettres de l'alphabet, avec un ensemble d'apprentissage de 100 exemples par lettre et un ensemble de test de 25 exemples par lettre. Mais d'autres résultats indiquent que ce classifieur est également le plus lent, ainsi que celui nécessitant le plus grand ensemble d'apprentissage. D'autre part, il a de très mauvaises capacités de généralisation et a montré de mauvaises performances pour la reconnaissance de signes connectés. Le classifieur Bayes(I) a aussi donné de très bons résultats pour la reconnaissance des 26 lettres de l'alphabet. Mais il a présenté de meilleures performances en terme de rapidité et de capacité de généralisation.

Au final, le classifieur de Bayes permet d'obtenir de très bons taux de reconnaissance, à condition de connaître précisément les vraisemblances des différentes classes ainsi que leur probabilité a priori. Dans la pratique, on ne peut obtenir que des estimations de ces probabilités. S'il est généralement possible d'obtenir une bonne estimation des probabilités a priori, par un simple dénombrement, l'estimation des vraisemblances, qui sont des densités de probabilité, implique de disposer d'un échantillon dont la taille croît exponentiellement avec la dimension de l'espace de représentation.

3.3. Réseaux de neurones

Un neurone est une fonction non linéaire, paramétrée, à valeurs bornées [Dreyfus 04]. Les deux principaux types de neurones sont les neurones à fonction d'activation sigmoïde - les entrées sont combinées linéairement puis cette combinaison est transformée par une fonction en forme de S - et les neurones à fonction radiale de base - les entrées sont combinées par une fonction non linéaire, en forme de cloche ou d'ondelette.

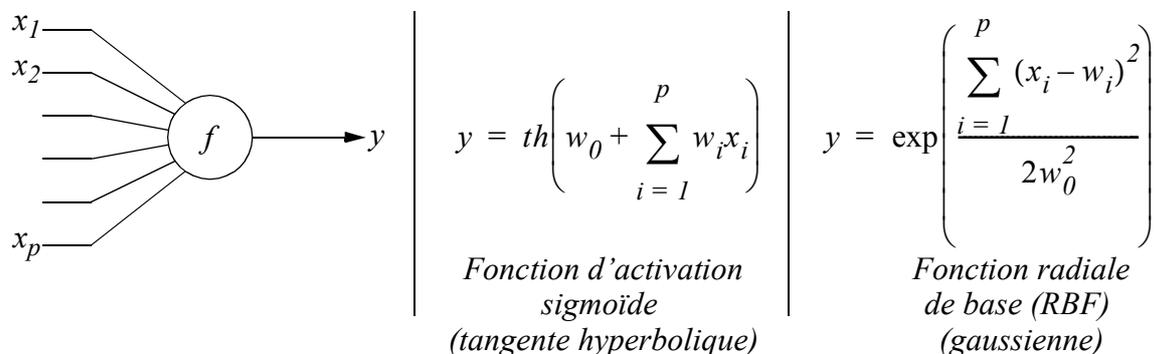


Fig. 28 Les deux principaux types de neurones.

Les neurones sont le plus souvent disposés en couches successives. Un tel réseau utilisant des neurones sigmoïdes est appelé Perceptron Multicouche. Un réseau de neurones effectue au final une combinaison non linéaire paramétrée des entrées. Les paramètres de cette combinaison sont appris de façon à ce que les sorties correspondent au mieux à l'ensemble d'apprentissage.

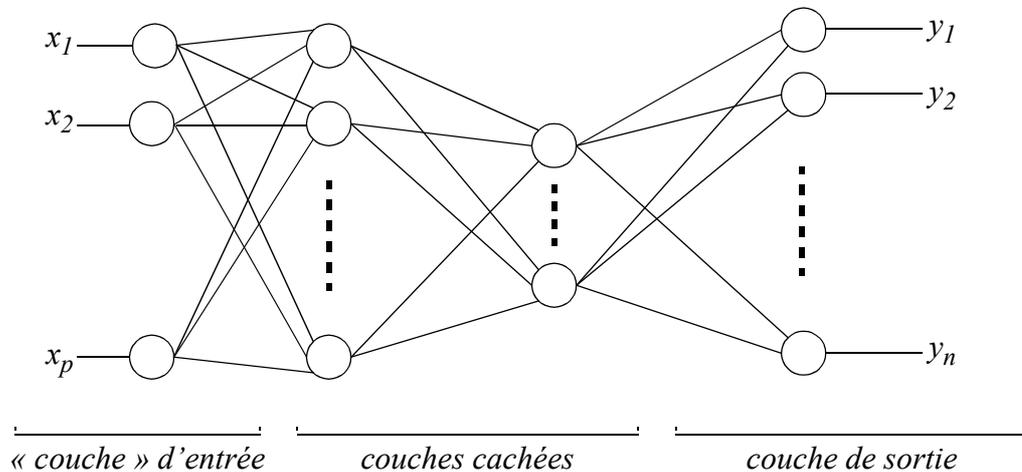


Fig. 29 Exemple de réseau de neurones à 4 couches.

L'approche la plus classique pour la reconnaissance de signes statiques est celle utilisant un perceptron à trois couches, où la taille de la couche d'entrée est égale à la dimension de l'espace de représentation et celle de la couche de sortie au nombre de signes à reconnaître. Le nombre de neurones de l'unique couche cachée est choisi selon des critères expérimentaux. Le signe reconnu est celui associé au neurone de sortie ayant la plus grande valeur [Murakami 91][Fels 93][Vamplew 96][Kramer 96].

D'autres types de réseaux de neurones ont également été utilisés pour la reconnaissance de signes statiques tels que les réseaux de neurones à fonctions radiales de base (RBF) [Weissman 99][Ng 02] ou les réseaux de neurones min-max flous [Kim 96].

Les réseaux de neurones se sont montrés très performants pour la reconnaissance de postures (voir Fig. 33). Leur grande capacité d'apprentissage en a fait un outil privilégié pour les systèmes basés sur la vision [Kohler 98][Wu 99]. Mais cette phase d'apprentissage est aussi leur point faible, chaque signe devant être répété un grand nombre de fois, si possible par plusieurs personnes.

3.4. Bases de règles - systèmes experts

La classification à base de règles est la plus simple dans son principe. La règle de décision $\delta(x)$ est définie par un ensemble de propositions logiques ou règles, ayant communément la forme suivante :

$$\frac{SI (x_1 \in A_1^k) \text{ et } (x_2 \in A_2^k) \text{ et } \dots \text{ et } (x_p \in A_p^k)}{\text{Prémisse } k} \text{ ALORS } \underline{\delta(x) = d_k}, \quad k = 0, 1, \dots, n. \quad (7)$$

Conclusion k

Un classifieur à base de règles, comme tout classifieur, discrétise l'espace de représentation en zones et attribue une classe, dont éventuellement la classe de rejet, à chacune de ces zones. Leur particularité est que cette discrétisation se fait composante par composante, donnant au final des frontières de décision parallèles aux axes de l'espace de représentation. Celui-ci doit donc être particulièrement bien choisi afin que les différentes classes soient facilement séparables par de telles frontières. La figure 30 illustre ce point : alors qu'il faut trois règles pour séparer les trois classes de la figure 30.a, il en faut trente-six pour séparer celles de la figure 30.b. Un autre choix de l'espace de représentation, tel que l'axe Y, aurait permis de séparer les trois classes de la figure 30.b avec seulement trois règles.

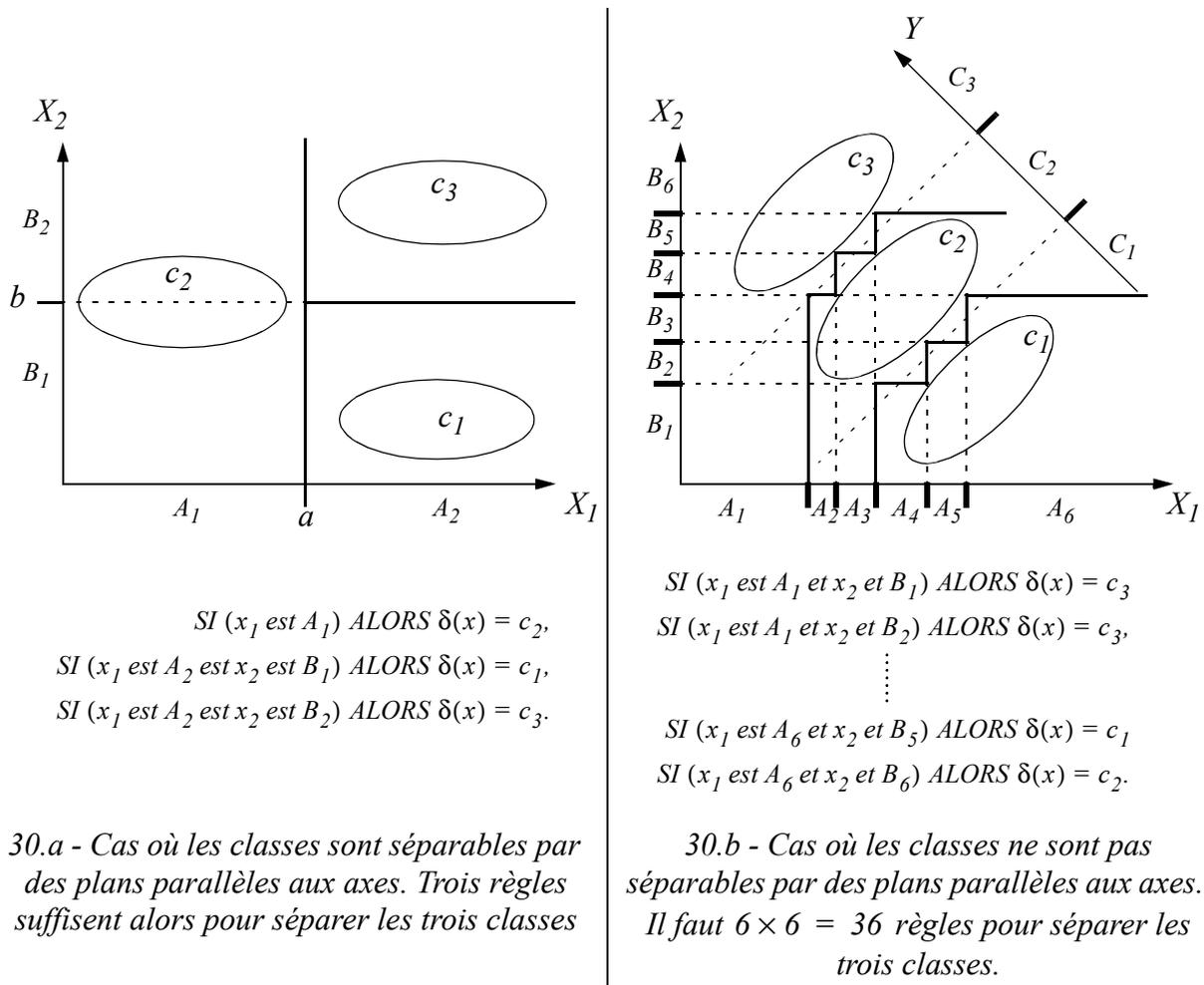


Fig. 30 Discretisation de l'espace de représentation par des bases de règles.

Les bases de règles ont plusieurs avantages sur les autres types de classifieurs. Elles sont premièrement très rapides du fait de leur simplicité. D'autre part, à condition que les partitions

de l'espace de représentation aient été choisies selon une certaine logique, les bases de règles ont une approche experte de la classification. Les données sont plus interprétées qu'elles ne sont classifiées. Le système est capable d'expliquer comment il attribue les classes aux observations. Il est en outre capable d'intégrer de la connaissance experte en modifiant ou en ajoutant des règles de façon heuristique. Enfin, les classifieurs à base de règles peuvent intégrer des variables catégorielles ou symboliques très simplement.

La classification de données issues d'un gant numérique par des bases de règles a été l'une des premières approches utilisées pour la reconnaissance de signes [Zimmerman 87]. Dans les travaux de Zimmerman, les ensembles A_i^k de la règle (7) n'ont pas de signification particulière : ils sont uniquement déterminés de façon à séparer au mieux les différentes classes dans l'ensemble d'apprentissage. L'inconvénient d'une telle approche est que l'on peut alors se retrouver avec des partitions telles que celle de la figure 30.b. D'autres travaux ont tenté de partitionner *a priori* les plages de variations des capteurs. Par exemple, Takahashi & Kishino discrétisent les mesures issues des capteurs du VPL DataGlove en leur attribuant le code H si elles sont supérieures à 45° et le code B sinon [Takahashi 91]. L'avantage de cette approche est que les ensembles A_i^k des prémisses ont une signification intuitive et de ce fait les règles ont une fonction explicative. Liang & Ouhyoung utilisent la même approche à cette différence près que le seuil entre les configurations *plié* et *tendu* n'a pas une valeur fixée à 45° , mais est ajusté en fonction des données [Liang 95].

Deux extensions des bases de règles ont été plus largement utilisées. La première fixe une structure hiérarchique entre les règles ce qui permet d'optimiser la détermination des frontières de décision et éventuellement de rechercher des axes de projection plus pertinents tels que l'axe Y de la figure 30.b. Il s'agit des arbres de décision. La seconde assouplit le caractère strict des classifieurs à base de règles en utilisant une discrétisation en zones floues. Ce sont les bases de règles floues.

3.4.1. Arbres de décision

Les arbres de décision ont d'abord été utilisés pour la classification de données décrites par plusieurs attributs catégoriels et donc discrets [Mitchell 97]. Ils ont été étendus à la classification de données représentées par des descripteurs continus en discrétisant l'espace de chacun de ces descripteurs en plusieurs zones. La classification d'une observation x se fait en examinant successivement la valeur de chacun de ses descripteurs. Prenons l'exemple de données représentées sous la forme de vecteurs à 3 composantes : $x = (x_1, x_2, x_3)$. L'espace de chaque composante est divisé en deux zones que l'on appellera zone + et zone -. Un arbre de décision classant ces données en 3 classes pourra être de la forme présentée à la figure 31.a ou, en optimisant l'ordre dans lequel sont testés les attributs, de celle de la figure 31.b.

Les deux arbres de la figure 31 sont des arbres de décision binaires car en chaque noeud se fait un choix entre deux possibilités. Se limiter à ce type d'arbre simplifie l'apprentissage simultané de la structure de l'arbre et des frontières de décision. Il est alors possible d'étendre la recherche de ces frontières de décision en ne se limitant plus au test de la valeur d'un seul descripteur, mais en considérant, par exemple, une combinaison linéaire des descripteurs. Dans ce cas, on cherche les frontières de décision parmi tous les hyperplans de l'espace de représentation et non plus uniquement parmi ceux qui sont parallèles aux axes. Cette approche a été utilisée par Kramer pour la reconnaissance de l'alphabet de la Langue des Signes Américaine. L'apprentissage de la structure de l'arbre ainsi que des combinaisons linéaires est décrit en détail dans [Kramer 96]. Il obtient des performances de reconnaissance comparables aux autres classifieurs pour les signes isolés, mais est bien plus rapide et a de très bonnes performances pour la reconnaissance des signes enchaînés. C'est, des quatre types de classifieurs étudié par Kramer (Bayes(W), Bayes(I) (§ 3.2), réseau de neurones (§ 3.3) et arbre de décision), celui qui a finalement été retenu pour le système *TalkingGlove*.

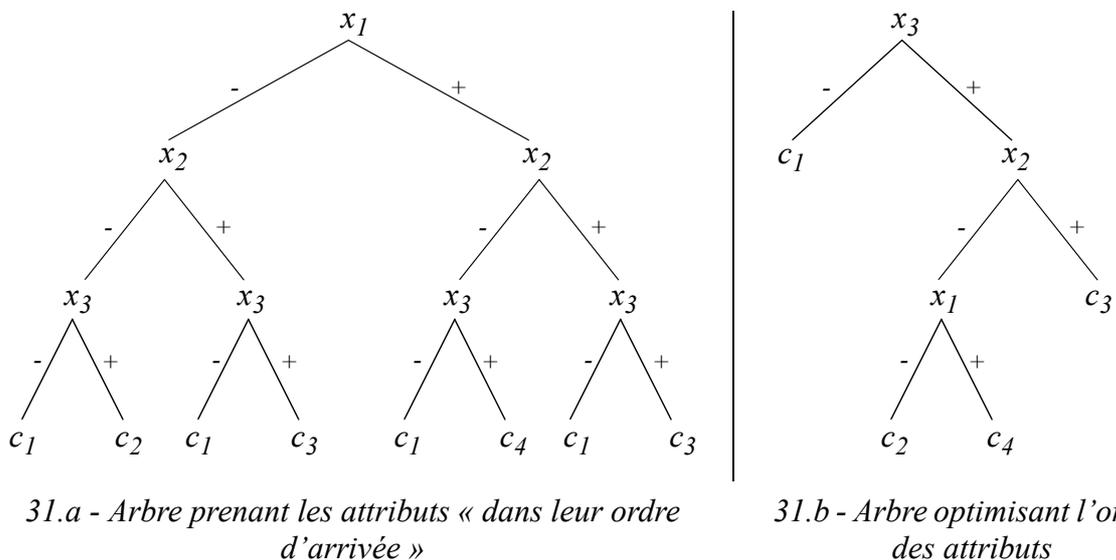


Fig. 31 Arbres de décision pour la classification de données représentées sous la forme de vecteurs de dimension 3.

Hernandez-Rebollar a également utilisé des arbres de décision pour reconnaître des postures à partir de mesures fournies par l'Accel Glove (voir Chapitre 1). Il obtient un classifieur aux très bonnes performances et tenant sur un micro-contrôleur PIC à bas prix [Hernandez 02.b].

3.4.2. Classifieurs à base de règles floues

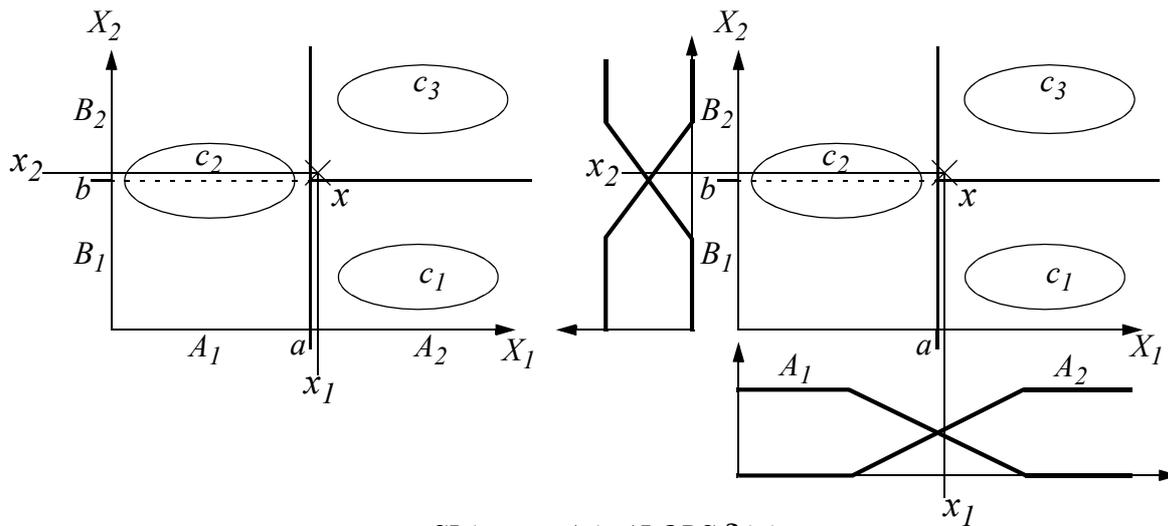
L'inconvénient des systèmes à base de règles classiques est qu'ils prennent rapidement des décisions irrévocables même dans les cas incertains. Ils ne peuvent pas repousser une décision pour attendre d'avoir plus d'informations. Reprenons l'exemple de la figure 30.a, on considère

le point, $x = \{a + \epsilon, b + \epsilon\}$, ϵ étant un réel positif proche de 0, représenté sur les figures 32.a et 32.b. Dans le cas net (voir Fig. 32.a), ce point sera classé c_3 car ses deux composantes appartiennent respectivement à A_2 et B_2 . Pourtant, ces deux appartenances sont très « faibles » : un bruit dans les mesures peut faire basculer le point x dans la classe c_1 ou c_2 . Le problème, ici, est que l'on accumule deux décisions peu fiables : l'appartenance de x_1 à A_2 et l'appartenance de x_2 à B_2 . Si les ensembles A_1, A_2, B_1 et B_2 sont maintenant des sous-ensembles flous, on aura alors :

$$\mu_{A_1}(x_1) \approx \mu_{A_2}(x_1) \approx \mu_{B_1}(x_2) \approx \mu_{B_2}(x_2) \approx 0.5. \quad (8)$$

Et si on choisit le produit et la somme bornée comme opérateurs pour le calcul des valeurs d'appartenance de x aux classes c_1, c_2 et c_3 , on aura :

$$\mu_{c_1}(x) \approx \mu_{c_3}(x) \approx 0.25 \quad \mu_{c_2}(x) \approx 0.5. \quad (9)$$



SI (x_1 est A_1) ALORS $\delta(x) = c_2$,
 SI (x_1 est A_2 est x_2 est B_1) ALORS $\delta(x) = c_1$,
 SI (x_1 est A_2 est x_2 est B_2) ALORS $\delta(x) = c_3$.

Fig. 32 Classification d'une observation avec une base de règles nettes et floues.

L'approche floue utilisant les deux opérateurs produit et somme bornée mènera donc ici à une décision en faveur de la classe c_2 , avec une certitude accrue par rapport à l'approche nette. Cependant, d'autres opérateurs auraient conduit à d'autre décision. Par exemple, l'utilisation des opérateurs min et max donnerait les degrés suivants :

$$\mu_{c_1}(x) \approx \mu_{c_3}(x) \approx \mu_{c_2}(x) \approx 0.5. \quad (10)$$

Dans ce cas, le classifieur peut choisir la classe de rejet. Il peut également fournir, comme sortie du système, les trois degrés d'appartenance, laissant l'interprétation de ces degrés aux soins d'un utilisateur ou d'un système flou en aval.

Au final, les systèmes à base de règles floues conservent leur capacité d'interprétation des données, tout en adoucissant les décisions tout au long du processus de classification. On trouvera une description détaillée de ce type de classifieurs dans [Bezdek 99].

Su utilise des règles floues pour la reconnaissance des 34 postures de la Langue des Signes Taïwanaise qu'il a mesurées avec un gant numérique de sa fabrication, le « EMI-Glove » [Su 00]. Les règles floues utilisées sont issues d'un ensemble de règles nettes fuzzifiées à l'aide d'une relation de similarité. L'ensemble de ces règles nettes est lui-même équivalent à un réseau de neurones d'un type particulier (réseaux composites hyper-rectangulaires). Ce système peut donc être vu comme un réseau de neurones modifié par une relation de similarité et n'a pas la capacité explicative que l'on recherche généralement dans un système à base de règles.

Dans [Karlsson 98.b], cinq signes statiques sont reconnus à l'aide d'un gant numérique utilisant des capteurs de pression, similaire au système de [Toney 98] décrit au chapitre 1 (§ 2.1.4). Le système de reconnaissance employé est un système à base de règles floues utilisant un opérateur d'agrégation original, issu des systèmes de fusion multi-capteurs [Karlsson 98.a].

Holden reconnaît 31 postures de la Langue des Signes Australienne avec un système expert flou [Holden 96]. Les postures sont capturées par un système de vision, la main étant équipée d'un gant de couleur, puis, les angles des doigts sont calculés en faisant correspondre un modèle 3D de la main aux images obtenues. Les domaines de variation de ces angles sont discrétisés en un ensemble de zones floues correspondant chacune à un terme tel que *plié* ou *tendu*. Les postures sont alors reconnues en utilisant une base de règles traduisant l'aspect de chaque angle dans cette configuration.

3.5. Tableau récapitulatif

Le premier des tableaux de la figure 33 présente les caractéristiques des système de reconnaissance de signes statiques cités dans cette section. La signification des différents sigles est donnée dans le deuxième tableau.

Référence	Index	An	TdS	Comp	Voc	Ench	Seg	Acqu	Rep	Clas	nbre	perf (%)	Remarque	
[Murakami 91]	423	1991	(S)	(Co) (Or)	(aLSJ)	(C)	(RCS)	(G)	(NA)	(NN)	42	98		
[Takahashi 91]	360	1991	(S)	(Co)	(aLSJ)	(I)		(G) (VDG)	(ACP)	(ADB)	46	20		
			(S)	(Co) (Or)								50		
[Fels 93]	124	1993	(S)	(Co) (Or)	(P)	(C)	(DMP)	(G) (VDG) (A) (Pol6)	(raw) (TP)	(NN)	66	98		
[Liang 95]	368	1995	(S)	(Co) (Mo) (Or) (PC)	(aLSA)	(C)	(RC)	(G) (VDG)	(P)	(BdR)	26	100		
[Holden 96]	254	1996	(S)	(Co) (Po) (Or)	(LSAu)	(I)		(V) (GCo) (M3D)	(PF)	(BdRF)	22	100?	Taux de reconnaissance que l'article laisse supposer	
[Kim 96]	355	1996	(S)	(Co)	(aLSJ)	(I)		(G) (VDG)	(raw)	(NN) (FmmNN)	14	?		
[Kramer 96]	975	1996	(S)	(Co)	(aLSA)	(I)		(G) (CG)	(raw)	(B(W))	100	26	100	
			(S)							(B(I))	100			
			(S)							(NN) (PMC)	100			
			(S)							(NN)	100			
			(S)			(C)	(MVi)	(B(W))	84					
			(S)			(B(I))	100							
			(S)			(NN) (PMC)	90							
			(S)			(NN)	99							
[Vamplew 96]	1047	1996	(S)	(Co)	(aLSAu)	(I)		(G) (CG)	(raw)	(NN)	61	91	30 configurations de bases + 31 variantes de ces configurations	
			(S)								30	99	Intégration des variantes des configurations dans les 30 signes standards	
[Sawada 98]	122	1998	(S)	(Co)	(LSJ)	(I)		(G) (CG)	(raw)	(1-PPVP)	14	100		
[Weissman 99]	40	1999	(S)	(Co) (Or)	(P)	(I)		(G) (CG)	(raw)	(NN) (PMC) (RBF)	20	100		
[Su 00]	30	2000	(S)	(Co)	(LST)	(I)		(G) (EMI)	(raw)	(BdRF)	34	100?	Taux de reconnaissance que l'article laisse supposer	
[Eisenstein 01]	39	2001	(S)	(Co) (Po)	(LSAa)	(I)		(G) (CG) (CGr)	(raw)	(1-PPVP)	10	83	Apprentissage non supervisé des prototypes des classes (k-moyennes)	
			(S)	(Or) (Fo)							(5-PPV)	88	Pas d'apprentissage	
[Hernandez 02b]	972	2002	(S)	(Co) (Or)	(LSAa)	(I)		(G) (AccG)	(TP)	(AD)	26	98	Taux de reconnaissance moyen. 21 postures sont parfaitement reconnues	

comp	Voc	Ench	Seg	Acqu	Rep	Clas	
Composante	Vocabulaire	Enchaînement	Segmentation	acquisition	représentation	classification	
Co Configuration	LSA Langue des Signes Américaine	I Isolés	DMP Détection d'un mouvement particulier	G Gant	ACP Analyse en composantes principales	AD Arbre de Décision	
Fo Force		C Connectés		AccG Accele Glove			
Mo Mouvement		LSAu Langue des Signes Australienne		RC Reconnaissance continue	CG CyberGlove	NA Normalisation d'amplitude	ADB Arbre de décision Binaire
Or Orientation					CGr CyberGrasp		
PC Points de contacts	EMI EMI Glove						
Po Position	LSJ Langue des Signes Japonaise		RCS Reconnaissance continue seuillée	VDG VPL DataGlove	P Partition de l'espace des données brutes	B Classifieur de Bayes	
				A Accéléromètre			
	P Vocabulaire personnel			Pol6 Polhemus (6D)	PF Partition floue	BdR Base de Règles Floues	
	LST Langue des Signes Taïwanaise			V Vision			
		aLSx Alphabet de la Langue des Signes x			Gco Gant de couleur	TP Traitement Personnalisé	k-PPV k-plus proches voisins "classique"
				M3D Reconstruction de la configuration de la main à l'aide d'un modèle 3D	raw Données brutes	k-PPVP k-plus proches voisins paramétré	
						NN Réseau de neurones	
						PMC Perceptron Multi-couches	
						RBF NN à fonctions radiales	
						FmmNN NN min-max flou	

LEGENDE

Fig. 33 Systèmes de reconnaissance de signes statiques.

4. Reconnaissance de signes dynamiques

Les signes dynamiques, tels qu'ils ont été définis au chapitre 1 (§ 1.1.2), sont des classes de gestes ayant une certaine durée temporelle. La reconnaissance de ces signes est plus difficile que celle des signes statiques car, à la variabilité des configurations, se rajoute une variabilité de la dynamique du geste : le même geste peut être exécuté à différentes vitesses. Les systèmes de reconnaissance doivent donc prendre en compte cette variabilité temporelle. Une première solution consiste à la supprimer. Soit en considérant qu'un signe doit toujours être réalisé de la même manière : le même geste effectué à deux vitesses différentes aura alors deux significations différentes. Soit en intégrant à l'étape de représentation une procédure de normalisation temporelle des gestes, raccourcissant les exécutions lentes d'un geste et étirant les exécutions rapides. Soit, enfin, en ne considérant que des paramètres qui restent invariables quelle que soit la vitesse d'exécution du signe. Une deuxième solution consiste à construire des algorithmes de classification permettant d'intégrer cette variabilité temporelle, comme ils intègrent la variabilité de réalisation. C'est ce que font les chaînes de Markov cachées ainsi que les réseaux de neurones récurrents.

De même que pour les signes statiques, les gestes à reconnaître peuvent être présentés un à un au système de reconnaissance (signes *isolés*) ou de manière continue. Dans ce deuxième cas, le début d'un signe peut être modifié par la fin du signe précédent. Si ce phénomène de co-articulation est pris en compte, on parlera de signes *enchaînés*, sinon de signes *connectés*.

La plupart des travaux en reconnaissance de signes dynamiques ont été faits dans l'objectif de la reconnaissance des langues des signes. Ces langages n'utilisent pas que la configuration de la main pour communiquer, mais bien le geste de la main dans son ensemble (configuration, orientation, position, mouvement) ainsi que des mimiques faciales. Les travaux que nous présentons dans cette section portent donc rarement sur la reconnaissance de configurations dynamiques uniquement, mais intègrent cette reconnaissance dans un système plus global de reconnaissance de gestes. D'autre part, les langages étudiés et les corpus de gestes reconnus diffèrent fortement d'une équipe de recherche à l'autre. S'il est possible de comparer qualitativement les techniques utilisées, il est difficile de comparer numériquement les résultats des différentes équipes. Nous donnerons néanmoins à la fin de la section un tableau récapitulatif permettant de se faire une idée générale des performances obtenues.

4.1. Systèmes supprimant la variabilité temporelle des signes dynamiques

4.1.1. Différenciation des gestes selon leur dynamique

Cette approche a été utilisée par Fels et Hinton pour leur interface geste/parole Glove-Talk [Fels 93]. Leur idée est d'utiliser la granularité de la parole et des gestes afin de définir plusieurs niveaux de correspondance entre ces deux modalités. Au niveau le plus grossier, un geste entier correspond à un mot en entier. Au niveau le plus fin, les mouvements des articulations de la main correspondent à des mouvements des articulateurs de la parole ou à des variations des paramètres du son parlé (fréquences et amplitudes des premiers formants, volume, hauteur,

etc.). Dans ce deuxième cas, il faut donc définir la correspondance geste/parole, mais il n'y a pas d'interprétation des gestes. Fels et Hinton ont implémenté ces deux solutions, respectivement, dans les interfaces Glove-Talk [Fels 93] et Glove-Talk II [Fels 98], utilisant toutes deux des réseaux de neurones.

Dans Glove-Talk, la configuration de la main permet de sélectionner la racine d'un mot tandis que la direction du mouvement de la main permet de sélectionner un suffixe pour cette racine (pluriel, passé, etc.). La durée et l'amplitude du geste permettent respectivement de contrôler la vitesse d'élocution et l'intonation. Ce geste est toujours précédé d'un mouvement déclencheur, comme si l'utilisateur devait pousser un bouton imaginaire avant de *signer* un mot. Les quatre correspondances, ainsi que la détection du mouvement de déclenchement, sont chacune réalisées par un Perceptron multi-couches à une couche cachée. Ainsi, le même geste effectué à des vitesses ou des amplitudes différentes, donnera lieu à deux sorties différentes du système. Quand un mouvement de déclenchement est détecté, la configuration de la main est envoyée au réseau de reconnaissance de la configuration. Ce dernier est donc un système de reconnaissance de signes statiques dont les performances ont été données dans la section précédente.

4.1.2. Normalisation temporelle des gestes

Les gestes sont des séquences de postures échantillonnées à intervalles de temps égaux. Si le même geste est effectué à deux vitesses différentes, il en résultera deux séquences de tailles différentes. Le but de la normalisation des gestes est de ramener toutes les séquences à une taille standard afin de pouvoir les comparer point par point. Plusieurs méthodes ont été proposées pour cette mise à l'échelle. On présente ici la méthode utilisée par Ishikawa et Sasaki [Ishikawa 02]. Soit N_0 la taille standard à laquelle doivent être ramenées toutes les séquences. La figure 34 illustre le principe appliqué pour normaliser un geste de taille $N \neq N_0$. Ce geste est rééchantillonné en N_0 échantillons. La valeur de chaque nouvel échantillon est calculée à partir des valeurs des anciens échantillons par une interpolation non-linéaire : une gaussienne est centrée sur le nouvel échantillon dont on veut calculer la valeur. Chaque ancien échantillon participe à cette valeur selon un poids déterminé par la gaussienne.

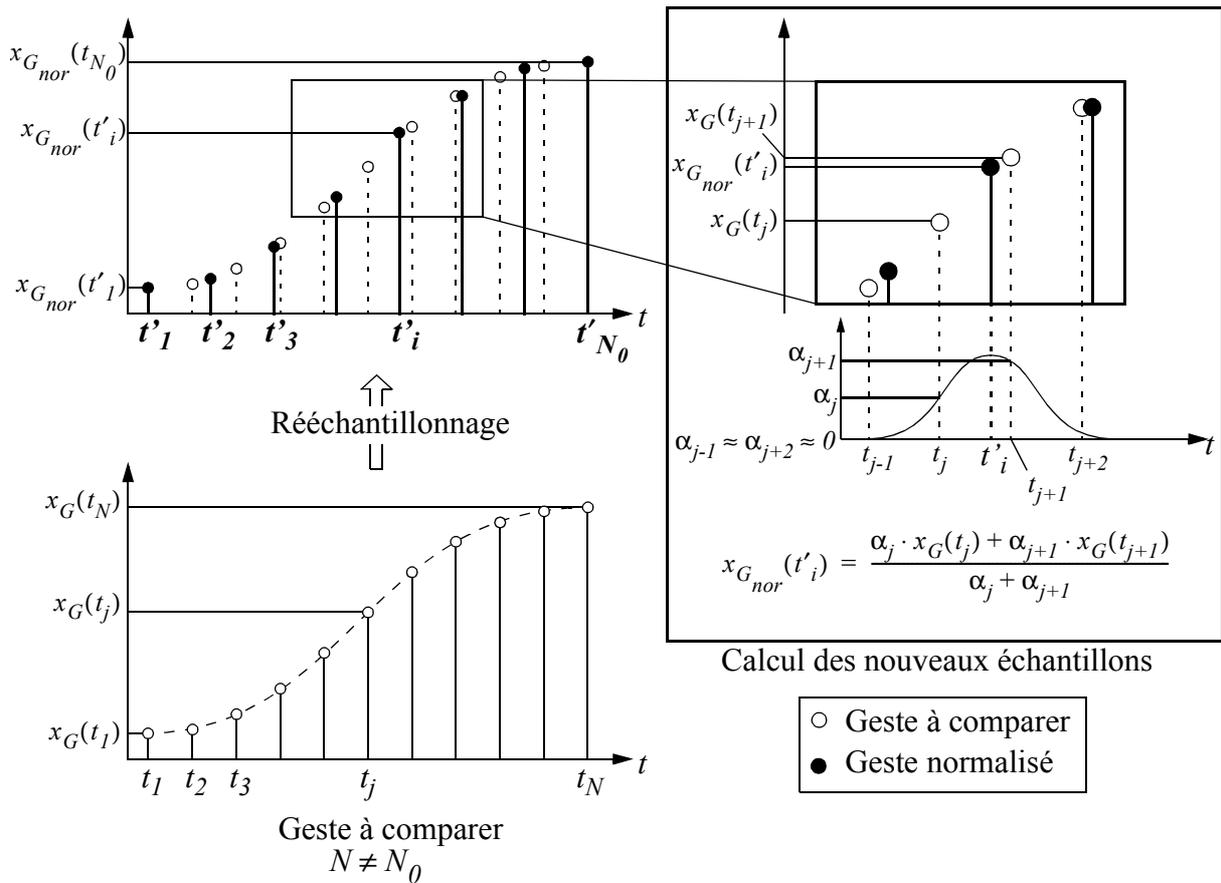


Fig. 34 Normalisation temporelle utilisée dans [Ishikawa 02].

Ishikawa et Sasaki appliquent cette normalisation sur des données issues de deux DataGloves (voir Chapitre 1) et d'un système de vision. Cette étape est nécessaire car ils utilisent, pour la classification, une carte auto-organisatrice (SOM) qui n'accepte qu'une seule taille de données en entrée. Lamar applique également une normalisation temporelle similaire, utilisant des splines à la place des gaussiennes [Lamar 00].

Ces normalisations sont efficaces pour traiter des gestes qui sont effectués globalement plus rapidement ou plus lentement que le signe à reconnaître. Mais elles ne permettent pas de traiter des gestes dont la variation de vitesse n'est pas globale. On peut, en effet, avoir un geste effectué plus rapidement dans sa première partie puis plus lentement dans sa seconde partie, de telle sorte qu'il soit globalement effectué à la même vitesse que le signe prototype. Ces techniques n'appliqueront, dans ce cas, aucune normalisation et le geste sera comparé tel quel, point par point, au prototype ce qui est illustré à la figure 35.

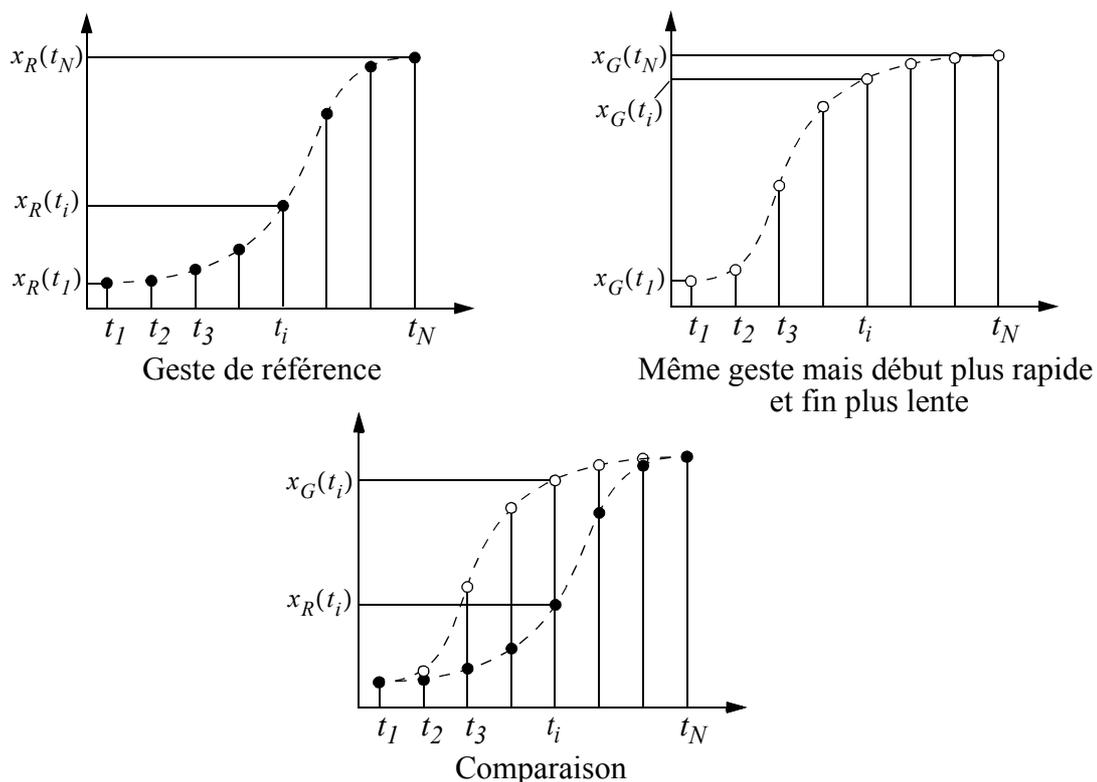


Fig. 35 Problème rencontré avec la normalisation telle qu'elle est effectuée dans [Lamar 00][Ishikawa 02]. La normalisation n'a ici aucun effet, elle ne permet pas de masquer les différences de dynamiques de réalisation des gestes.

Dans le chapitre 4, nous utiliserons une nouvelle méthode de normalisation, basée sur un rééchantillonnage en déplacements égaux plutôt qu'en intervalles de temps égaux, qui permettra de normaliser les gestes, quelle que soit leur dynamique interne.

4.1.3. Reconnaissance de composantes invariantes

Il s'agit ici de s'affranchir de la variabilité temporelle des gestes en ne représentant plus ces derniers comme des séquences de postures, mais par des caractéristiques invariantes temporellement, c'est-à-dire qui ne dépendent pas de la vitesse d'exécution du geste.

Par exemple, dans [Lee 97] et [Hernandez 04], le modèle de Stokoe est utilisé pour définir les gestes [Stokoe 80]. Un signe est donc décomposé en une posture initiale, une posture finale (seulement dans [Hernandez 04]) et un mouvement de la main entre ces deux postures. Les postures initiales et finales sont des caractéristiques statiques, donc invariantes temporellement et le mouvement est représenté par des paramètres globaux tels que la courbure globale de la trajectoire, l'amplitude ou la direction générale du mouvement.

Le système Grasp développé par Kadous reconnaît 95 signes de l'Auslan (Langue des Signes Australienne) mesurés par un PowerGlove (voir Chapitre 1). Un ensemble de caractéristiques globales (limites spatiales, histogrammes, etc.) est extraite des gestes et utilisée pour la classification. Deux classifieurs sont utilisés : un 1-PPV et un arbre de décision construit avec l'algorithme C4.5 [Kadous 95].

On trouve aussi une telle approche dans [Zacksenhouse 01] où des gestes manipulatifs, encore appelés gestes ergotiques (voir chapitre 1), sont reconnus à partir des mesures du Cyberglove. Ce système porte sur les gestes ergotiques dit simultanés, c'est-à-dire où les doigts sont coordonnés de telle sorte que leurs mouvements forment, dans l'espace des données, une hyperligne. Ces gestes se différencient les uns des autres d'après la direction de cette hyperligne, qui est un paramètre indépendant de la vitesse d'exécution du geste. Ils peuvent donc être reconnus avec un système n'intégrant pas la composante temporelle (ici, un réseau de neurone de type ART).

4.2. Systèmes prenant en compte la variabilité temporelle des signes dynamiques

4.2.1. Modèles de Markov Cachés

Un modèle de Markov permet de représenter un système pouvant se trouver à un instant t dans un état pris parmi un ensemble de N états possibles. L'état dans lequel se trouvera le système à l'instant $t + 1$ dépend uniquement de son état à l'instant t , le passage d'un état à un autre étant un processus stochastique. Un tel modèle est défini par ses états S_1, S_2, \dots, S_N , ses probabilités de transitions $a_{ij} = P(q_t = S_j | q_{t-1} = S_i)$, $1 \leq i, j \leq N$, q_t étant l'état du processus à l'instant t , ainsi que par ses probabilités a priori : $\pi_i = P(q_0 = S_i)$, q_0 étant l'état initial du processus. La figure 36 donne l'exemple d'un modèle de Markov à trois états.

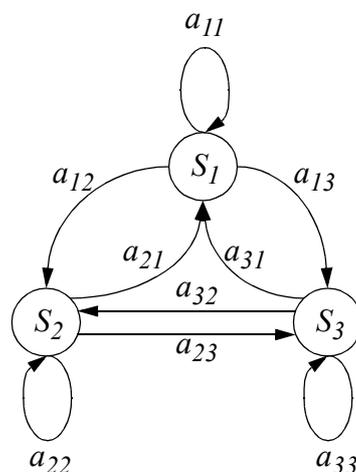


Fig. 36 Modèle (ou chaîne) de Markov à 3 états.

Dans un modèle de Markov classique, on sait, à un instant donné, dans quel état se trouve le processus, car chaque état correspond à une observation unique. Dans un modèle de Markov caché (HMM pour Hidden Markov Models), lorsque le processus est dans un état donné S_i , il peut produire différentes observations selon différentes probabilités appelées probabilités d'émission. Les états ne sont donc pas accessibles directement à l'observation, d'où le nom de modèles cachés. La définition d'un tel modèle nécessite de préciser les valeurs des probabilités d'émission des différents états. L'ensemble O des observations possibles peut être discret ou continu, on parle alors de modèles de Markov cachés discrets ou continus. Dans le cas continu, les probabilités d'émissions sont des densités que l'on modélise généralement par des gaussiennes.

Les HMMs sont donc des modèles doublement stochastiques qui permettent de prendre en compte la variabilité temporelle d'un phénomène. Ils ont largement été utilisés en reconnaissance de la parole, où ils ont permis d'obtenir de très bonnes performances [Rabiner 89][Picone 90]. Ces résultats en parole ont conduit la communauté scientifique à les utiliser massivement pour la reconnaissance de signes dynamiques [Loeding 04].

Dans un système de reconnaissance utilisant des HMMs, chaque signe à reconnaître est modélisé par un HMM. Un premier choix, généralement empirique, porte donc sur la structure de ces HMMs (nombres d'états, passages possibles d'un état à un autre). Une fois ceci réalisé, les différents paramètres des modèles sont estimés à partir d'un ensemble d'apprentissage en utilisant, par exemple, la procédure itérative de Baum-Welch [Rabiner 89]. Dans le cas de la reconnaissance de signes connectés ou enchaînés, les ensembles d'apprentissage des HMMs n'ont pas besoin d'être segmentés. Etant donnée une séquence d'apprentissage, ou phrase, comportant un ensemble de signes, il suffit de donner la traduction de cette phrase et la procédure d'apprentissage cherchera elle-même la meilleure segmentation possible [Young 02]. C'est là un des grands avantages des modèles de Markov cachés.

Les classifieurs à base d'HMMs font partie de la famille des classifieurs de Bayes. Ils utilisent la formule de Bayes pour estimer la probabilité a posteriori des différents signes. L'étape de reconnaissance se fait donc en cherchant, pour un signe donné, le modèle ayant la plus forte vraisemblance. La vraisemblance d'un HMM est la probabilité que ce modèle produise le signe à reconnaître. Mais on lui préfère généralement la vraisemblance des séquences. Celle-ci consiste à chercher, pour chaque HMM, la séquence d'états qui a la plus forte probabilité d'avoir eu lieu étant donné l'observation. Cette recherche peut se faire avec l'algorithme de Viterbi. On choisit alors l'HMM ayant la séquence la plus vraisemblable. Là encore, dans le cas de la reconnaissance de signes connectés ou enchaînés, l'algorithme de reconnaissance cherchera lui-même la meilleure segmentation possible de la phrase à reconnaître.

La première idée pour la reconnaissance de signes utilisant des HMMs a été de modéliser chaque signe à reconnaître par un HMM [Starner 95][Braffort 96][Lee 96][Vogler 98][Hienz 99][Ma 00]. Cette approche a l'avantage d'être très simple dans sa mise en oeuvre et est

particulièrement adaptée aux vocabulaires de faible dimension. Elle est donc toujours beaucoup utilisée pour reconnaître efficacement un faible nombre de signes mesurés dans des conditions difficiles comme par exemple des systèmes de vision en milieu naturel [Chen 03] [Ramamoorthy 03] ou des électromyogrammes [Wheeler 03].

La limitation principale de ces systèmes se manifeste lorsque le nombre de signes à reconnaître devient important : la phase d'apprentissage est alors vite contraignante, nécessitant un corpus d'apprentissage de taille conséquente. C'est pourquoi les nouvelles approches, suivant une fois de plus le chemin tracé par les recherches en reconnaissance de la parole, se sont concentrées sur la décomposition des gestes en un ensemble fini de sous-unités minimales ou phonèmes, chaque modèle de Markov représentant une de ces sous-unités. Contrairement aux langues parlées pour lesquelles les phonèmes sont bien identifiés, le choix de ces unités minimales diffère selon les auteurs. Dans [Liang 98], elles sont basées sur le modèle linguistique de Stokoe [Stokoe 80]. Les sous-unités sont alors des gestes pendant lesquels la configuration de la main ne change pas. Vogler et Metaxas utilisent, quant à eux, le modèle segmental *Movement-Hold* introduit par Liddell et Johnson [Liddell 90] dans lequel les signes sont une succession de mouvements et de pauses [Vogler 99.a]. Enfin, Bauer et Kraiss ont une troisième approche qui ne s'appuie pas sur un modèle phonologique de la langue des signes [Bauer 02]. Les unités minimales ne sont alors pas définies a priori mais apprises à partir d'un ensemble de phrases : elles sont auto-organisatrices. Le principe est d'attribuer le même nombre de phonèmes à chaque signe et d'attribuer un HMM par phonème, tous les HMMs ayant la même structure. Les capacités d'auto-segmentation des algorithmes d'apprentissage des HMMs permettent alors d'apprendre les phonèmes sans savoir a priori à quoi ils correspondent. Cette approche est d'autant plus intéressante qu'elle est applicable à tous les langages gestuels et pas uniquement aux langues des signes.

On trouve dans la littérature de nombreuses extensions des HMMs. Ainsi, dans [Gao 00][Corradini 01][Fang 04], les probabilités d'émissions sont modélisées par des réseaux de neurones, formant au final des HMMs hybrides pouvant s'adapter à différents *signeurs*. Dans [Wilson 01], ces probabilités d'émission dépendent de paramètres globaux, tels que l'amplitude du geste, ce qui permet d'identifier ces paramètres et de reconnaître les inflexions présentes dans les gestes co-verbaux. Enfin, des HMMs parallèles ont été utilisés dans [Vogler 99.b] [Wilson 01][Vogler 03] pour tenir compte de la simultanéité des différentes composantes du geste.

4.2.2. Réseaux de neurones récurrents

Les réseaux de neurones qui se sont révélés efficaces pour la classification de signes statiques, tels que les perceptrons multicouches, ne permettent pas d'intégrer la variabilité temporelle des signes dynamiques. D'autres types réseaux de neurones sont adaptés à ce type de données : il s'agit des réseaux à décalage et des réseaux de neurones récurrents [Dreyfus 04]. Les réseaux de neurones récurrents d'Elman (voir Fig. 37) sont souvent utilisés. On retrouve les trois couches « classiques » : couche d'entrée, couche cachée et couche de sortie, plus une

couche appelée couche de contexte qui est une copie de la couche cachée à l'instant $t - 1$ (caractéristique des réseaux de neurones récurrents). La couche d'entrée peut elle-même être constituée des données d'entrée à l'instant t , mais aussi des données d'entrée aux instants $t - 1$, $t - 2$, etc. (caractéristique des réseaux de neurones à décalage).

Un tel réseau est utilisé dans [Murakami 91]. Il est précédé d'un réseau de type perceptron pour la détection des postures initiales des gestes. Ng et Ranganath ont également utilisé des réseaux de neurones récurrents dans un système de reconnaissance basé sur la vision [Ng 02]. Ils reconnaissent tout d'abord un ensemble de cinq postures avec un réseau de neurones à fonctions radiales de bases. La sortie de ce réseau, ainsi que des informations sur le mouvement de la main sont utilisées pour la reconnaissance de 14 signes dynamiques. Ils utilisent dans un premier temps 14 modèles de Markov cachés (discrets et continus) et dans un deuxième temps 14 réseaux de neurones récurrents. Les meilleurs taux de reconnaissance sont obtenus avec les HMMs continus et les réseaux de neurones récurrents qui donnent tous deux des résultats similaires.

Lamar utilise des réseaux d'Elman comme sous-unités du réseau global T-CombNET qui a une structure de peigne (*comb*) [Lamar 00]. Ce travail est intéressant dans sa partie expérimentale où sont comparées les performances du réseau T-combNET avec celles d'autres types de classifieurs. Ces résultats montrent qu'un simple *I-PPV* précédé d'une procédure de normalisation temporelle peut parfois donner de meilleurs résultats qu'un réseau de neurones récurrent d'Elman.

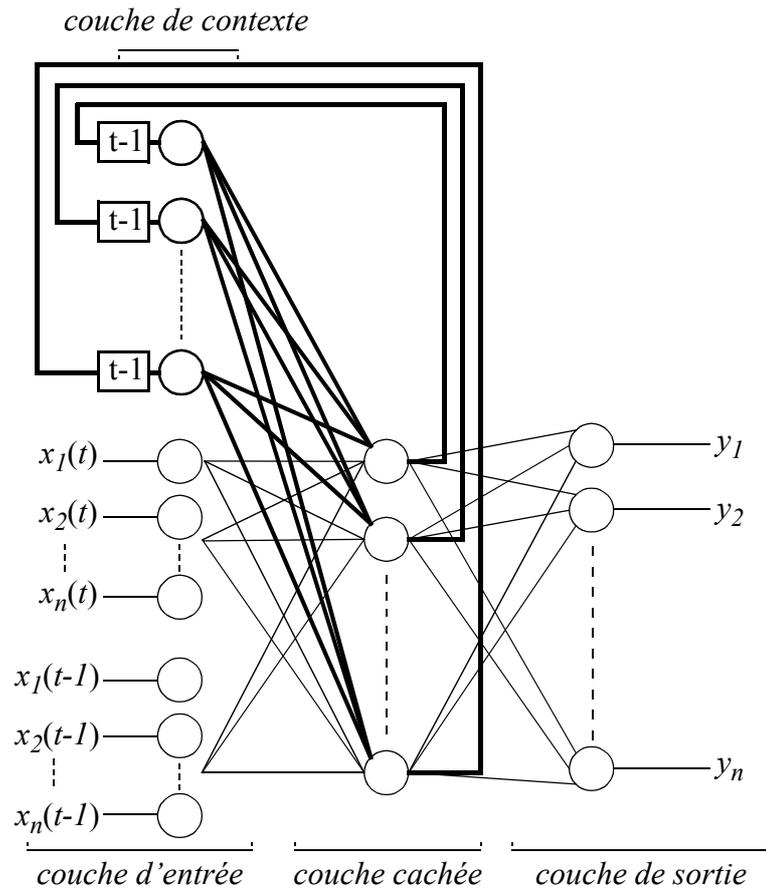


Fig. 37 Réseau récurrent d'Elman.

Le système SLARTI développé par Vamplew est un système de reconnaissance de la Langue des Signes Australienne [Vamplew 96] utilisant presque exclusivement des réseaux de neurones. Dans une première étape, sept composantes du geste sont identifiées par quatre réseaux de neurones, à partir des mesures fournies par un Cyberglove et un capteur de localisation 6D. Trois d'entre eux sont des perceptrons multicouches et permettent de reconnaître les configurations, orientations et positions de début et de fin du geste. Il y a 30 configurations, 15 orientations et 19 positions possibles. Un quatrième réseau de neurone, récurrent celui-ci, se charge de la classification du mouvement de la main parmi les 13 mouvements possibles. Au final, un geste est représenté par 7 attributs catégoriels. La reconnaissance des signes se fait alors avec un classifieur 1-*PPV*, la distance entre deux gestes étant définie comme la somme des 7 distances entre leurs caractéristiques. Deux distances possibles entre caractéristiques sont étudiées : une distance simple - qui est égale à 0 si les caractéristiques sont identiques et à 1 sinon - et une distance heuristique dont la valeur dépend des deux caractéristiques si celles-ci sont différentes. Par exemple, pour la configuration de la main, deux configurations peuvent être plus ou moins semblables et la distance heuristique traduit cette éventuelle similarité. Ces similarités sont établies à partir des erreurs effectuées pendant l'apprentissage par le perceptron chargé de leur classification.

4.3. Tableau Récapitulatif

Référence	Index	An	TdS	Comp	Voc	Ench	Seg	Acqu	Rep	Clas	nbre	perf (%)	Remarque
[Murakami 91]	423	1991	(D)	(Co) (Or) (Po)	(LSJ)	(C)	(DPD)	(G)	(TP)	(NN) (rNN)	10	96	
[Starner 95]	259	1995	(D)	(Im)	(LSA)	(E)	(Mvr)	(V)	(TP)	(HMM)	40	97	
[Braffort 96]	227	1996	(D)	(Co) (Or) (Po) (Mo)	(LSF)	(E)	(Mvr)	(G) (CG) (Pol6)	(raw) (TP)	(HMM)	44	96	44 phrases utilisant 7 signes
[Holden 96]	254	1996	(D)	(Co) (Po) (Or) (Mo)	(LSAu)	(I)		(V) (GCo) (M3D)	(PF)	(BdRF)	22	95	
[Kadous 96]	460	1996	(D)	(Co) (Po) (Or) (Mo)	(LSAu)	(I)		(G) (PG)	(TP)	1-ppv AD	95	83 55	
[Kim 96]	355	1996	(D)	(Po)	(LSK)	(C)	(RC)	(G) (VDG)	(P)	(ADB)	10	85	
[Vamplew 96]	1047	1996	(D)	(Co) (Po) (Or) (Mo)	(aLSAu)	(I)		(G) (CG) (Pol6)	(Cac)	(1-ppv)	52	94	
[Liang 98]	427	1998	(D)	(Co) (Or) (Po) (Mo)	(LST)	(C)	(Dpa)	(G) (VDG) (Pol3)	(raw)	(HMM)	250	81	Phrases utilisant 250 signes construits à partir des sous-unités du modèle de Stokoe
[Sawada 98]	122	1998	(D)	(Co) (Po) (Or) (Mo)	(LSJ)	(C)	(PA)	(G) (CG) (A) (Pol6)	(Cac)	(Aut) (HMM)	30	89 96	
[Vogler 98]	465	1998	(D)	(Im)	(LSA)	(E)	(Mvi)	(V) (CaM)	(TP)	(HMM)	97	90	97 phrases utilisant 54 signes
[Hienz 99]	425	1999	(D)	(Im)	(LSG)	(E)	(Mvr)	(V) (Gco)	(TP)	(HMM)	?	95	Phrases utilisant 52 signes
[Gao 00]	1011	2000	(D)	(Co) (Po) (Or) (Mo)	(LSC)	(E)	(Mvr)	(G) (CG)	(raw)	(NN) (HMM)	1065	93	80 phrases utilisant 220 signes, effectuées par différentes personnes. (même système que Ma (184))
[Lamar 00]	1093	2000	(D)	(Im)	(aLSJ)	(I)		(V)	(TP)	(rNN) (1-ppv) (PMC) (T-CombNET)	34	81 91 92 93	Résultats expérimentaux montrant qu'un 1-PPV précédé d'une normalisation peut faire mieux qu'un réseau de neurone récurrent
[Ma 00]	184	2000	(D)	(Co) (Po) (Mo)	(LSC)	(E)	(Mvr)	(G) (CG)	(raw)	(NN) (HMM)	80	98	80 phrases utilisant 220 signes, effectuées par différentes personnes
[Su 00]	30	2000	(D)	(Co)	(LST)	(I)		(G) (EMI)	(Cac)	(Aut)	90	94	
[Bauer 02]	1162	2002	(D)	(Im)	(LSG)	(I)		(V)	(TP)	(HMM)	100	93	100 signes constitués de 150 sous unités auto-organisatrices
[Ishikawa 02]	1052	2002	(D)	(Co) (Po)	(LSJ)	(E)	(Mvr)	(G) (V)	(NT)	(SOM)	17	97	
[Ng 02]	470	2002	(D)	(Co) (Mo)	(P)	(C)	(Mvr)	(V)	(cac) (TP)	(HMM) (rNN)	14	90 90	
[Vogler 03]	1059	2003	(D)	(Co) (Or) (Po) (Mo)	(LSA)	(E)	(Mvr)	(G) (CG) (MoS)	(TP)	(HMM)	99	96	99 phrases utilisant 22 signes. Les signes sont construits à partir d'unités minimales du type <i>Mouvement-Hold</i>
[Fang 04]	1010	2004	(D)	(Co) (Po) (Or) (Mo)	(LSC)	(I)		(G) (CG)	(raw)	(Hybride)	5113	92	Signes effectués par différentes personnes
[Hernandez 04]	995	2004	(D)	(Co) (Or) (Po) (Mo)	(LSA)	(I)		(G) (AccG)	(TP)	(AD)	176	94	

comp		Voc		Ench		Seg		Acqu		Rep		Clas	
Composante		Vocabulaire		Enchaînement		Segmentation		acquisition		représentation		classification	
Co	Configuration	LSA	Langue des Signes Américaine	I	Isolés	DPa	Détection de pause	G	Gant	Cac	Caractéristiques catégorielles	AD	Arbre de Décision
Im	Image de la main			C	Connectés			AccG	Accele Glove				
Mo	Mouvement	LSAu	Langue des Signes Australienne	E	Enchaînés	DPD	Détection Posture de Départ	CG	CyberGlove	P	Partition de l'espace des données brutes	Aut	Automate d'Etat
Or	Orientation							EMI	EMI Glove			BdR	Base de Règles
PC	Points de contacts	LSC	Langue des Signes Chinois	Mvi	Minimum de vitesse	PA	Pic d'accélération	PG	PowerGlove	PF	Partition floue	BdRF	Base de Règles Floues
Po	Position							VDG	VPL DataGlove			k-PPV	k-plus proches voisins "classique"
		LSG	Langue des Signes Allemande	RC	Reconnaissance continue	Mvr	Maximum de Vraisemblance (HMM)	A	Accéléromètre	TP	Traitement Personnalisé	NN	Réseau de neurones
								MoS	MotionStar (Ascension Technology)			rNN	NN récurrent
		LSJ	Langue des Signes Japonaise	Pol3	capteur Polhemus (3D)	Pol6	Capteur Polhemus (6D)	V	Vision	raw	Données brutes	SOM	Carte auto organisatrices
								CaM	Caméras Multiples			HMM	Chaines de Markov Cachées
		LSF	Langue des Signes Française	V	Vision	Gco	Gant de couleur	M3D	Reconstruction de la configuration de la main à l'aide d'un modèle 3D				
										P	vocabulaire personnel		
		LST	Langue des Signes Taiwanaise										
		aLSx	Alphabet de la Langue des Signes x										

LEGENDE

Fig. 38 Systèmes de reconnaissance des signes dynamiques.

5. Conclusion

La diversité des approches utilisées montre à quel point le problème de la reconnaissance des signes a intéressé la communauté scientifique ces vingt dernières années. Les applications sont très nombreuses, allant de l'aide aux personnes malentendantes à la communication homme-machine en passant par la commande de procédés. Les techniques utilisées pour l'aborder ont beaucoup emprunté aux systèmes de reconnaissance vocale, puis se sont spécialisées à la problématique spécifique du geste. Aujourd'hui, la recherche dans ce domaine est très active. Plusieurs conférences internationales sont dédiées à la reconnaissance de gestes et de nombreux projets de recherche sur le geste naissent chaque année. Pourtant, la diversité des gestes étudiés, des applications envisagées ainsi que des systèmes de mesures utilisés freinent l'unification et la comparaison des différents travaux. Certains corpus de signes permettant de comparer les performances des différentes approches ont vu le jour ces dernières années, mais ce sont exclusivement des corpus visuels (vidéos ou images).

L'un des intérêts du travail que nous allons maintenant présenter réside dans le fait que nous utilisons une représentation symbolique de la configuration de la main qui, bien que nous la construisons à partir des données du Cyberglove, se veut générale et pourrait être construite à partir d'autres systèmes de mesures du geste de la main, gants numériques ou systèmes de vision. Les opérations effectuées, pour la reconnaissance de signes statiques et dynamiques entre autres, sur cette description symbolique de la main sont totalement indépendantes du système de mesure utilisé. C'est là un avantage des approches expertes : quel que soit le système de mesure utilisé, une main reste une main et les mots et relations utilisés pour la décrire ne changent pas.

La description symbolique que nous utilisons est basée sur le formalisme mathématique des symbolisations, s'appuyant lui-même en grande partie sur la théorie des sous-ensembles flous et ses nombreux concepts. C'est ce formalisme qui va maintenant être présenté.



www.phdcomics.com

CHAPITRE 3

*Introduction au formalisme des
 ϕ -symbolisations*

1. Introduction

Dans ce manuscrit, nous nous intéressons à la description de la configuration de la main. Les trois autres composantes du geste manuel, à savoir l'orientation, la position et le mouvement de la main, ne sont pas prises en compte. Nous utilisons le Cyberglove (voir chapitre 1) pour mesurer les angles de flexion des doigts de la main. Ce gant possède dix-huit capteurs de flexion. Il fournit donc une représentation numérique de la configuration de la main sous la forme d'un vecteur à dix-huit composantes.

[1.58, 1.17, 1.65, 0.32, 1.22, 0.91, 1.51, 1.76, 1.64, 1.58, 1.73, 1.01, 1.84, 1.42, 1.38]

Fig. 39 A quelle posture de la main correspond ce vecteur?

Cette représentation numérique permet de différencier efficacement les configurations de la main et de construire des systèmes de reconnaissance de signes performants (voir chapitre 2). Un tel système sera capable de décider si le vecteur de la figure 39 correspond à un des signes à reconnaître ou non car il intègre une connaissance sur les données issues du Cyberglove qui lui permet de les interpréter. Cette connaissance lui vient, d'une part, de son concepteur, qui a préalablement traduit sa propre connaissance en un programme de classification et, d'autre part, d'une phase d'apprentissage, effectuée à partir d'exemples des différents signes. Mais si on ne possède pas cette connaissance, les données telles qu'elles sont fournies par le Cyberglove ne sont pas interprétables. C'est pourquoi on dira que ces données, bien que complètes et précises, sont sémantiquement pauvres.

Nous rappelons dans ce chapitre le formalisme des symbolisations, introduit dans le cadre de la science de la mesure [Finkelstein 75] et dont l'extension floue est au coeur de la construction des capteurs symboliques [Benoit 93]. Plus particulièrement, nous utiliserons des ϕ -symbolisations, qui sont un cas particulier de symbolisations floues. Ce chapitre a un caractère formel et permet de poser les bases théoriques utiles à la représentation symbolique de la configuration de la main et aux traitements effectués sur cette représentation.

Dans la section 2, nous rappelons la définition des symbolisations et des différents concepts s'y rapportant. La section 3 décrit la méthode de construction des ϕ -symbolisations que nous utiliserons par la suite. Les sections 4 et 5, enfin, introduisent des outils permettant de traiter les informations converties sous forme symbolique par une ϕ -symbolisation.

2. Symbolisations

Un symbole peut se définir comme “*un objet ou un évènement qui a une relation définie avec une entité et dont le but est de pouvoir représenter cette entité en son absence*” [Finkelstein 75]. Les symboles peuvent être de natures très diverses (panneau de signalisation, drapeau, lettre de l’alphabet, dessin, nombre, etc.) de même que les entités qu’ils représentent (règle de circulation, nation, résultat d’un examen, logo d’une entreprise, température d’une pièce, etc.). La donnée d’un symbole isolé ne suffit pas à fournir l’information sur l’entité qu’il représente : le nombre 23 sans autre information ne nous apprend rien. S’agit-il du numéro d’un joueur, d’une température, d’une limitation de vitesse, d’un entier quelconque, d’un réel ou d’autre chose ? Pour savoir à quelle(s) entité(s) se rapporte ce nombre, il faut connaître la procédure par laquelle il a été attribué. Cette procédure s’appelle la symbolisation.

2.1. Définition

La symbolisation désigne le processus d’assignation d’un ensemble fini de symboles S , appelé univers symbolique, à un ensemble E d’entités, appelé univers de discours [Finkelstein 75]. Lorsque l’univers symbolique est formé uniquement de termes du langage naturel, on l’appelle alors plutôt univers lexical. Pour éviter toute confusion avec les mots de notre discours, nous suivons la convention consistant à noter les termes d’un univers lexical en *italique* et à utiliser le tiret bas « _ » lorsqu’un terme est constitué de plusieurs mots. Dans le cadre de la description de la configuration de la main, les symboles utilisés seront des termes liés à la flexion (*plié*, *écarté*, etc.), l’orientation (*externe*, *en_avant*, etc.) ou l’écartement (*écartés*, *serrés*) des doigts.

Une mesure est définie comme *l’assignation, par un processus objectif et empirique, de nombres ou autres symboles aux attributs d’objets ou évènements du monde réel, afin de décrire ces derniers* [Finkelstein 94]. Les mesures sont donc des symbolisations particulières dont l’univers de discours est composé d’attributs d’objets ou évènements du monde réel.

Chaque capteur du Cyberglove définit alors une symbolisation dont l’univers de discours est un ensemble empirique constitué de l’ensemble des flexions possibles du capteur. L’univers symbolique est l’ensemble des valeurs que peuvent prendre les mesures de ce capteur. Les capteurs du Cyberglove fournissant des mesures codées sur 8 bits, cet ensemble symbolique contient 256 éléments.

Les convertisseurs numériques-symboliques, qui sont au coeur de la construction des capteurs symboliques, *ont pour fonction de recevoir des informations sous forme numérique et de produire des informations symboliques en relation avec les premières* [Benoit 93]. Ces convertisseurs définissent donc également des symbolisations. Si on considère un convertisseur numérique-symbolique permettant de convertir les mesures d’un capteur de flexion du Cyberglove en informations symboliques utilisant, par exemple, les termes *tendu* et *plié*. Dans ce cas, les 256 valeurs pouvant être prises par une mesure de ce capteur forment l’univers de

discours de la symbolisation, dont l'univers symbolique est l'ensemble lexical $L = \{tendu, plié\}$.

2.2. Symbolisations nettes

2.2.1. Définition

Le processus d'assignation de symboles aux entités d'une symbolisation est formalisé par la définition d'une relation entre l'univers de discours et l'univers symbolique. Une symbolisation se définit donc comme un triplet $\langle E, S, R \rangle$, où E est l'univers de discours, S l'univers symbolique et R une relation sur $E \times S$.

Définition 1 ([Finkelstein 75]) : Soit E et S deux ensembles, S étant fini, et R une relation sur $E \times S$. Le triplet $\langle E, S, R \rangle$ est appelé symbolisation. Les ensembles E et S sont respectivement l'univers de discours et l'univers symbolique de la symbolisation.

On n'utilise généralement pas la relation R directement, mais plutôt l'une de ses deux projections que sont la description et la signification. Dans le cas flou, ces concepts ont été introduits par Zadeh pour formaliser le lien qui peut exister entre des termes d'un langage et un ensemble numérique [Zadeh 71.a] :

- La *description* D d'une entité est l'ensemble des symboles représentant cette entité.
- La *signification* M (meaning) d'un symbole est l'ensemble des entités représentées par ce symbole.

Dans le cadre de la symbolisation, ces deux concepts sont définis formellement comme suit :

Définition 2 ([Benoit 93]) : La description D associée à une symbolisation $\langle E, S, R \rangle$ est une fonction de l'univers de discours E dans l'ensemble $P(S)$ des parties de S , qui, à tout $e \in E$, associe le sous-ensemble $D(e)$:

$$D(e) = \{s \in S, (e, s) \in R\}. \quad (11)$$

Définition 3 ([Benoit 93]) : La signification M associée à une symbolisation $\langle E, S, R \rangle$ est une fonction de S dans l'ensemble des parties de E , $P(E)$, qui, à tout $s \in S$, associe le sous-ensemble $M(s)$:

$$M(s) = \{e \in E, (e, s) \in R\}. \quad (12)$$

Un couple $\langle s, M(s) \rangle$ forme ce que l'on appelle un *concept* [Benoit 93]. Un concept peut être défini plus généralement comme l'association d'un symbole, d'un univers symbolique et d'un sous-ensemble d'un univers de discours :

Définition 4 : Soit E et S deux ensembles, S étant fini. Soit P un sous-ensemble de E et s un élément de S . Le couple $\langle s, P \rangle$ forme un concept, dont l'univers de discours est l'ensemble E et l'univers symbolique est S . Le symbole s est le nom du concept et le sous-ensemble P la signification du concept.

Une symbolisation forme donc une collection de concepts ayant tous le même univers de discours et le même univers lexical. Une symbolisation $\langle E, S, R \rangle$ est *complète* si toute entité est représentée par au moins un symbole et *non redondante* si toute entité est décrite par un symbole unique [Finkelstein 75]. On a alors clairement la proposition suivante :

Proposition 1 : Une symbolisation $\langle E, S, R \rangle$ est complète et non redondante si et seulement si l'ensemble des significations forme une partition de E .

Lorsque la symbolisation associée à une mesure est complète et non redondante, elle forme ce que l'on appelle une échelle nominale en science de la mesure.

2.2.2. Exemple

Nous prenons l'exemple de la symbolisation des mesures d'un capteur du Cyberglove. On note $U = \{u_0, u_1, \dots, u_{255}\}$ l'ensemble des valeurs que peuvent prendre les mesures de ce capteur et L un ensemble de trois termes du langage courant, $L = \{\text{plié}, \text{mi_plié}, \text{tendu}\}$. On considère les deux symbolisations, $\langle U, L, R_1 \rangle$ et $\langle U, L, R_2 \rangle$ dont les significations des termes sont représentées à la figure 40.

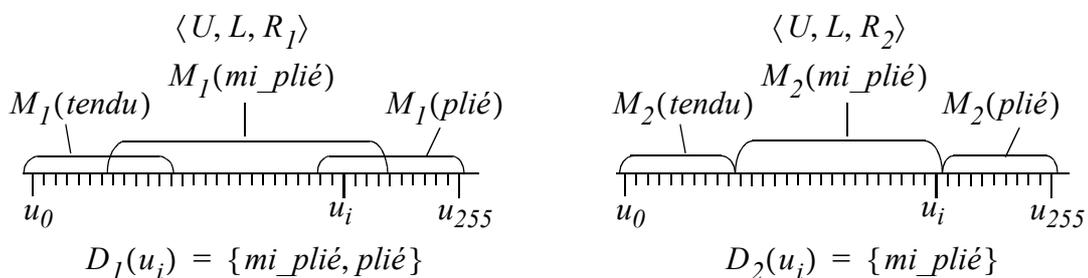


Fig. 40 Deux exemples de symbolisation. Seule $\langle U, L, R_2 \rangle$ est complète et non redondante.

La symbolisation $\langle U, L, R_1 \rangle$ est complète mais elle est redondante. Les significations des termes de L dans cette symbolisation ne forment pas une partition de U . La symbolisation $\langle U, L, R_2 \rangle$ est, elle, complète et non redondante.

Les symbolisations que nous venons de définir sont qualifiées de nettes car les significations et les descriptions qui leur sont associées sont des sous-ensembles nets. Lorsque de telles symbolisations sont utilisées dans des conversions numérique-symbolique, celles-ci s'apparentent alors à un processus de seuillage et sont peu robustes à la proximité des seuils. La description de la mesure u_i dans la symbolisation $\langle U, L, R_2 \rangle$ est le singleton $\{mi_plié\}$, mais si cette mesure était légèrement supérieure, sa description deviendrait $\{plié\}$. Le concept de symbolisation a donc été étendu au cas flou pour pouvoir introduire de la gradualité dans les significations.

2.3. Symbolisations floues

2.3.1. Définition

Le concept de symbolisation peut être élargi en considérant des relations floues entre les symboles et les entités.

Définition 5 ([Benoit 93]) : Soit E et S deux ensembles, S étant fini, et R une relation floue sur $E \times S$. Le triplet $\langle E, S, R \rangle$ est appelé symbolisation floue. Les ensembles E et S sont respectivement l'univers de discours et l'univers symbolique de la symbolisation.

Dans ce cas, les significations et descriptions deviennent également des sous-ensembles flous :

Définition 6 ([Benoit 93]) : La description D associée à une symbolisation floue $\langle E, S, R \rangle$ est une fonction de l'univers de discours E dans l'ensemble $F(S)$ des sous-ensembles flous de S , qui, à tout $e \in E$, associe le sous-ensemble flou $D(e)$:

$$\forall s \in S, \quad \mu_{D(e)}(s) = \mu_R(e, s). \quad (13)$$

Définition 7 ([Benoit 93]) : La signification M associée à une symbolisation floue $\langle E, S, R \rangle$ est une fonction de S dans l'ensemble $F(E)$ des sous-ensembles flous de E , qui, à tout $s \in S$, associe le sous-ensemble flou $M(s)$:

$$\forall e \in E, \quad \mu_{M(s)}(e) = \mu_R(e, s). \quad (14)$$

On a alors toujours l'égalité suivante :

$$\forall s \in S, \forall e \in E, \quad \mu_{M(s)}(e) = \mu_{D(e)}(s). \quad (15)$$

Lorsque le contexte nécessitera cette précision, on pourra éventuellement parler de description floue ou de signification floue pour indiquer que l'on est dans le cadre d'une symbolisation floue.

La notion de concept doit aussi être étendue au cas flou. On parlera alors de concept flou :

Définition 8 : *Soit E et S deux ensembles, S étant fini. Soit F un sous-ensemble flou de E et s un élément de S . Le couple $\langle s, F \rangle$ forme un concept flou, dont l'univers de discours est l'ensemble E et l'univers symbolique est S .*

Une symbolisation floue forme donc une collection de concepts flous, tous associés aux mêmes univers de discours et univers symbolique.

2.3.2. ϕ -symbolisations

Dans le cas net, une symbolisation complète et non redondante se caractérise par le fait que l'ensemble des significations des symboles forme une partition de l'univers de discours. L'extension floue d'une symbolisation complète et non redondante est une symbolisation floue dont l'ensemble des significations des symboles forme une partition floue de l'univers de discours. De nombreuses extensions du concept de partition aux sous-ensembles flous ont été données dans la littérature. On trouvera une revue de ces définitions dans [Montes 03]. Nous utiliserons pour nos travaux des partitions floues de Ruspini [Ruspini 69], qui ont largement été utilisées dans le cadre de la modélisation de connaissances expertes, notamment dans les convertisseurs numériques-symboliques de capteurs flous [Benoit 93]. Certains auteurs justifient leur utilisation en affirmant qu'elle garantissent une intégrité sémantique des sous-ensembles flous les composant [Guillaume 04].

Définition 9 : *Soit U un ensemble et $\mathcal{A} = \{A_i, i \in I\}$ une famille finie de sous-ensembles flous de U . La famille \mathcal{A} forme une partition de Ruspini de U , si et seulement si elle vérifie :*

$$\forall u \in U, \quad \sum_{i \in I} \mu_{A_i}(u) = 1. \quad (16)$$

Les partitions de Ruspini sont des cas particuliers de ϕ -partitions. Ces dernières ont été définies par De Baets et Mesiar.

Définition 10 ([De Baets 96]) : Soit U un ensemble et $\mathcal{A} = \{A_i, i \in I\}$ une famille finie de sous-ensembles flous de U . Soit ϕ une bijection strictement croissante de $[0, 1]$ dans lui-même. La famille \mathcal{A} forme une ϕ -partition de U si et seulement si elle vérifie :

$$\forall u \in U, \quad \sum_{i \in I} \phi(\mu_{A_i}(u)) = 1. \quad (17)$$

On définit alors les ϕ -symbolisations comme des symbolisations dont l'ensemble des significations forme une ϕ -partition de l'univers de discours. On impose également que ces significations soit normées, c'est-à-dire qu'elles aient un noyau non vide.

Définition 11 : Soit ϕ , une bijection strictement croissante de $[0, 1]$ dans lui-même. Une ϕ -symbolisation est une symbolisation $\langle E, S, R \rangle$ vérifiant :

$$\begin{aligned} \forall s \in S, \exists e \in E, \quad \phi(\mu_R(e, s)) &= 1, \\ \forall e \in E, \quad \sum_{s \in S} \phi(\mu_R(e, s)) &= 1. \end{aligned} \quad (18)$$

La première condition de (18) assure que les significations d'une ϕ -symbolisation $\langle E, S, R \rangle$ soient normées et la seconde qu'elles forment une ϕ -partition de S , puisqu'elles vérifient alors :

$$\forall e \in E, \quad \sum_{s \in S} \phi(\mu_{M(s)}(e)) = \sum_{s \in S} \phi(\mu_R(e, s)) = 1. \quad (19)$$

Les éléments du noyau de la signification d'un symbole seront appelés les prototypes de ce symbole.

Définition 12 : Soit $\langle E, S, R \rangle$ une ϕ -symbolisation et $s \in S$ un symbole. Les prototypes de s sont les éléments du noyau de $M(s)$, où M est la signification associée à cette symbolisation.

Les partitions floues de Ruspini sont des ϕ -partitions avec $\phi = Id$, Id étant la fonction identité sur l'intervalle unité. Dans nos travaux, nous utilisons donc exclusivement des Id -symbolisations. Néanmoins, afin de cadrer ce travail dans un contexte plus général et de mettre en relief ses extensions possibles, nous présentons le formalisme relatif aux ϕ -symbolisations. Pour chaque résultat important, nous donnerons la forme correspondant au cas $\phi = Id$.

2.3.3. Exemple

Nous reprenons l'exemple du paragraphe précédent, où l'univers de discours est $U = \{u_0, u_1, \dots, u_{255}\}$ et l'ensemble des symboles est $L = \{\text{plié}, \text{mi_plié}, \text{tendu}\}$. La figure 41 donne les significations des termes, ainsi que la description de deux valeurs particulières de U , correspondant à une Id -symbolisation $\langle U, L, R \rangle$. Les significations forment ici une partition floue de Ruspini de U .

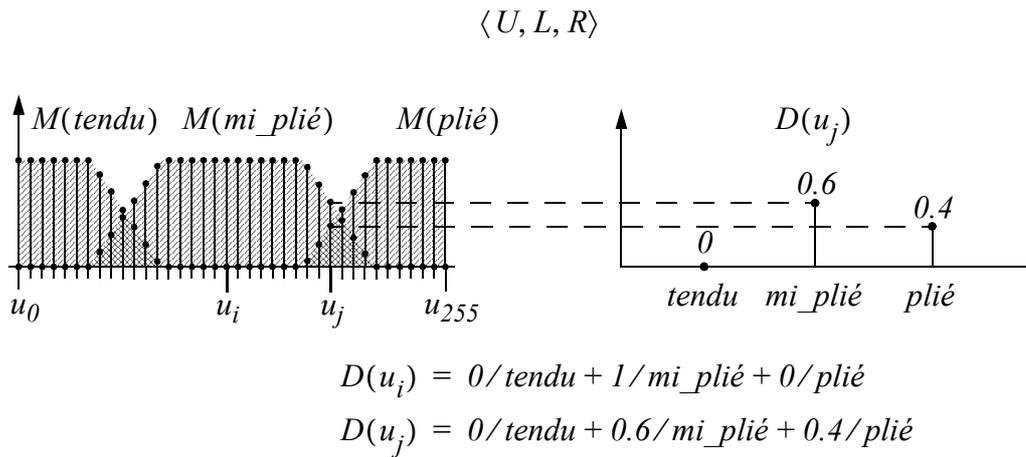


Fig. 41 Une Id -symbolisation de la mesure d'un capteur de flexion.

Pour alléger les notations, on ne précisera que les degrés d'appartenance non nuls. D'autre part, un sous-ensemble flou dont les degrés d'appartenance valent soit 0 soit 1 est également un sous-ensemble net et pourra alors être noté comme tel. Ainsi la description de la mesure u_i dans l'exemple de la figure 41 pourra être noté comme un singleton de L :

$$\begin{aligned}
 D(u_i) &= \{\text{mi_plié}\}, \\
 D(u_j) &= 0.6/\text{mi_plié} + 0.4/\text{plié}.
 \end{aligned}
 \tag{20}$$

2.3.4. Sémantique des fonctions d'appartenance

La fonction d'appartenance d'un sous-ensemble flou peut avoir trois types de sémantique [Dubois 97] : la similarité, la préférence et l'incertitude. Nous considérons, dans ce qui suit, que les significations d'une symbolisation floue ont une sémantique de similarité. Le degré d'appartenance d'une entité e à la signification d'un symbole s représente la similarité de cette entité à un prototype de ce symbole. Comme nous l'avons dit précédemment, si ce degré d'appartenance vaut 1.0, cela signifie que l'entité e peut elle-même être considérée comme un prototype du symbole s .

3. Construction de ϕ -symbolisations

Dans cette section, nous présentons une méthode pour construire des ϕ -symbolisations permettant de convertir sous forme symbolique les mesures données par un ou plusieurs capteurs du Cyberglove. Les résultats présentés ici sont généraux et peuvent être appliqués à la symbolisation d'autres mesures ou d'autres entités que celles qui nous intéressent.

Chaque capteur du Cyberglove prend ses valeurs sur l'intervalle $[0, 2.55]$. Les symbolisations que l'on cherche à définir ont alors comme univers de discours un produit cartésien d'intervalles réels $X = X_1 \times X_2 \times \dots \times X_p$ (voir Fig. 42). Une solution consisterait à construire directement la signification de chaque symbole comme un sous-ensemble flou de X . On pourrait, pour cela, utiliser un algorithme de classification floue [Bezdek 99] en le contraignant afin que la partition floue finale soit Id -partition, voire une ϕ -partition.

Nous utilisons une approche différente qui consiste à transformer, le plus tôt possible dans la chaîne de traitement de l'information, les données numériques en données symboliques, ou plus exactement lexicales. Pour cela, une ϕ -symbolisation est construite sur chaque intervalle X_j . Ces différentes ϕ -symbolisations sont ensuite agrégées en utilisant deux processus de construction que sont le produit cartésien de ϕ -symbolisations et le regroupement de symboles. Au final, les ϕ -symbolisations sont construites par l'application de trois méthodes de constructions qui vont maintenant être présentées.

$$\left. \begin{array}{l} \text{Capteur 1 : } X_1 = [0, 2.55] \\ \text{Capteur 2 : } X_2 = [0, 2.55] \\ \dots \\ \text{Capteur } p : X_p = [0, 2.55] \end{array} \right\} X = X_1 \times X_2 \times \dots \times X_p$$

Fig. 42 Construire une symbolisation sur un ensemble de p capteurs signifie construire une partition sur un produit cartésien d'intervalles réels.

3.1. Construction d'une ϕ -symbolisation sur un intervalle réel

Pour pouvoir construire une ϕ -symbolisation sur un intervalle réel, nous introduisons tout d'abord une méthode permettant de construire des ϕ -partitions sur un tel intervalle. Cette méthode est basée sur l'utilisation d'intervalles flous (L-R).

3.1.1. Construction d'une ϕ -partition sur un intervalle réel par des intervalles flous (L-R)

Définition 13 ([Dubois 87]) : Un intervalle flou (L-R) est un sous-ensemble flou A de \mathbb{R} défini par la fonction d'appartenance suivante :

$$\forall x \in \mathbb{R}, \quad \mu_A(x) = \begin{cases} L\left(\frac{a-x}{\alpha}\right) & \text{si } x \in]a-\alpha, a[, \\ 1 & \text{si } x \in [a, b], \\ R\left(\frac{x-b}{\beta}\right) & \text{si } x \in]b, b+\beta[, \\ 0 & \text{sinon,} \end{cases}$$

où a et b sont deux réels tels que $a \leq b$, α et β sont deux réels positifs et L, R sont deux fonctions de $]0,1[$ dans $[0,1[$, décroissantes, continues à gauche et telles que $L(x) > 0$ et $R(x) > 0$ pour tout x appartenant à $]0,1[$. L'intervalle flou A est alors noté : $(a, b, \alpha, \beta)_{LR}$.

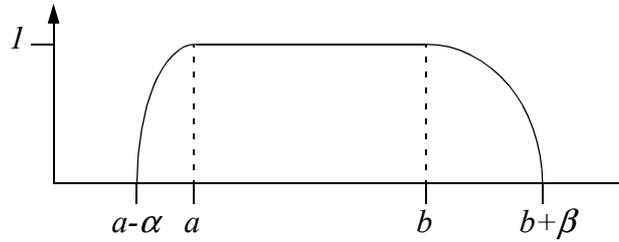


Fig. 43 Exemple d'intervalle flou (L-R).

Les intervalles flous (L-R) peuvent être utilisés pour définir une ϕ -partition sur un intervalle réel [Foulloy 04.b].

Proposition 2 : Soit ϕ une bijection strictement croissante de $[0, 1]$ dans lui-même. Soit $\mathcal{A} = \{A_i, i \in \{1, \dots, n\}\} = \{(a_i, b_i, \alpha_i, \beta_i)_{LR}, i \in \{1, \dots, n\}\}$ une famille de n intervalles flous (L-R), définis sur $[a_1, b_n] \subset \mathbb{R}$, avec $L(u) = R(u) = \phi^{-1}(1-u)$ pour tout u appartenant à $]0,1[$ et tels que $a_{i+1} - b_i = \alpha_{i+1} = \beta_i > 0$, pour tout $i \in \{1, \dots, n-1\}$ et $a_1 = \beta_n = 0$.

Alors, \mathcal{A} est une ϕ -partition de $[a_1, b_n] \subset \mathbb{R}$.

Preuve : Montrons d'abord que pour tout $i \in \{1, \dots, n-1\}$ et tout $x \in]b_i, a_{i+1}[$, on a :

$$L\left(\frac{a_{i+1}-x}{\alpha_{i+1}}\right) = \phi^{-1}\left(1 - \phi\left(R\left(\frac{x-b_i}{\beta_i}\right)\right)\right). \quad (21)$$

Pour tout $i \in \{1, \dots, n-1\}$, on a $a_{i+1} - b_i = \alpha_{i+1} = \beta_i$. Alors, pour tout $i \in \{1, \dots, n-1\}$ et tout $x \in]b_i, a_{i+1}[$, on a :

$$\begin{aligned} \phi^{-1}\left(1 - \phi\left(R\left(\frac{x-b_i}{\beta_i}\right)\right)\right) &= \phi^{-1}\left(1 - \phi\left(\phi^{-1}\left(1 - \frac{x-b_i}{\beta_i}\right)\right)\right) = \phi^{-1}\left(\frac{x-b_i}{\beta_i}\right) = \phi^{-1}\left(1 - \frac{\alpha_{i+1} + b_i - x}{\alpha_{i+1}}\right) \\ &= \phi^{-1}\left(1 - \frac{a_{i+1} - x}{\alpha_{i+1}}\right) = L\left(\frac{a_{i+1} - x}{\alpha_{i+1}}\right). \end{aligned}$$

Pour tout $i \in \{1, \dots, n\}$ et tout $x \in [a_i, b_i]$, on a :

$$\sum_{j=1}^n \phi(\mu_{A_j}(x)) = \phi(\mu_{A_i}(x)) = 1$$

Pour tout $i \in \{1, \dots, n\}$ et tout $x \in]b_i, a_{i+1}[$, on a :

$$\sum_{j=1}^n \phi(\mu_{A_j}(x)) = \phi\left(R\left(\frac{x-b_i}{\beta_i}\right)\right) + \phi\left(L\left(\frac{a_{i+1}-x}{\alpha_{i+1}}\right)\right) = \phi\left(R\left(\frac{x-b_i}{\beta_i}\right)\right) + \phi\left(\phi^{-1}\left(1 - \phi\left(R\left(\frac{x-b_i}{\beta_i}\right)\right)\right)\right) = 1$$

■

3.1.2. Construction d'une ϕ -symbolisation sur un intervalle de \mathbb{R}

Soit $X = [x_0, x_1]$ un intervalle de \mathbb{R} et $S = \{s_1, \dots, s_n\}$ un ensemble fini de symboles. Nous cherchons à définir une relation floue R sur $X \times S$ telle que la symbolisation $\langle X, S, R \rangle$ soit une ϕ -symbolisation. A chaque symbole s_i de l'ensemble lexical S est associé un intervalle $[a_i, b_i] \subset X$ qui est l'ensemble des prototypes de s_i et est noté p_i . Ces « intervalles prototypes » doivent être disjoints. Ils correspondent à des valeurs de X caractéristiques des symboles. On peut, sans perdre la généralité du problème, considérer que les symboles s_i sont indicés de manière à ce que $b_i < a_{i+1}$ pour tout $i \in \{1, \dots, n-1\}$. Afin de recouvrir l'intégralité de l'intervalle X , il faut que ses bornes soient des prototypes de symboles : $x_0 = a_1$ et $x_1 = b_n$.

On pose, pour tout $i \in \{1, \dots, n-1\}$: $\alpha_{i+1} = \beta_i = a_{i+1} - b_i$ et $\alpha_1 = \beta_n = 0$. On peut alors définir la signification de chaque symbole s_i comme l'intervalle flou (L-R) $(a_i, b_i, \alpha_i, \beta_i)_{LR}$, avec $L(u) = R(u) = \phi^{-1}(1 - u)$ pour tout u appartenant à $]0, 1[$. D'après la

proposition 2, l'ensemble des significations forme alors une ϕ -partition de l'intervalle $[a_1, b_n]$ c'est-à-dire de X . Comme les intervalles flous (L-R) sont normés par définition, on a bien défini une ϕ -symbolisation.

3.1.3. Exemple

La figure 44 illustre la construction des significations d'un ensemble de symboles $S = \{s_1, s_2, s_3\}$ sur l'intervalle $[0, 10]$. Les prototypes des symboles sont les intervalles $p_1 = [0, 1]$, $p_2 = [3, 6]$ et $p_3 = [10, 10]$. Deux constructions sont proposées, en utilisant successivement la fonction $\phi = Id$ (identité sur $[0, 1]$) et $\phi : u \mapsto u^2$.

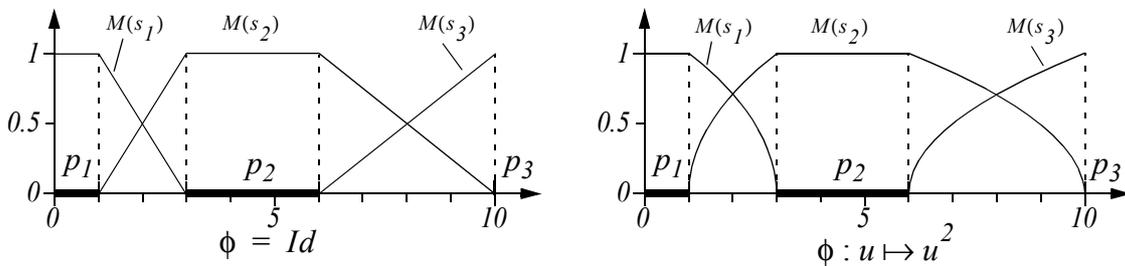


Fig. 44 Construction d'une Id -symbolisation et d'une ϕ -symbolisation ($\phi : u \mapsto u^2$) à partir de prototypes et d'intervalles flous (L-R).

Maintenant que l'on sait construire des ϕ -symbolisations sur des intervalles de \mathbb{R} , nous allons voir comment les agréger pour obtenir une ϕ -symbolisation sur un produit cartésien d'intervalles. Nous introduisons, dans un premier temps, le produit cartésien de ϕ -symbolisations et dans un deuxième temps, une méthode de changement de l'univers symbolique d'une symbolisation par regroupement de symboles.

3.2. Produit cartésien de ϕ -symbolisations

Etant donné plusieurs ϕ -symbolisations, on peut les agréger sous la forme d'un produit cartésien qui sera lui-même une ϕ -symbolisation. Avant de définir formellement un produit cartésien de ϕ -symbolisations, nous rappelons des résultats concernant les produits cartésiens de ϕ -partitions.

3.2.1. Produit cartésien de ϕ -partitions

Proposition 3 ([Foulloy 04.c]) : Soit $\mathcal{A} = \{A_i, i \in I\}$ et $\mathcal{B} = \{B_j, j \in J\}$, deux ϕ -partitions respectivement de deux ensembles E_1 et E_2 . Soit T_P^ϕ , la t -norme isomorphe à la t -norme produit, définie par :

$$\forall (u, v) \in [0, 1]^2, \quad T_P^\phi(u, v) = \phi^{-1}(\phi(u) \cdot \phi(v)). \quad (22)$$

Alors la famille de sous-ensembles flous $(\mathcal{A}, \mathcal{B}) = \{C_{(i,j)}, i \in I, j \in J\}$ définis par :

$$\forall (e_1, e_2) \in E_1 \times E_2, \quad \mu_{C_{(i,j)}}(e_1, e_2) = T_P^\phi(\mu_{A_i}(e_1), \mu_{B_j}(e_2)), \quad (23)$$

est une ϕ -partition de $E_1 \times E_2$.

Preuve : Pour tout couple $(e_1, e_2) \in E_1 \times E_2$, on a :

$$\begin{aligned} \sum_{(i,j) \in I \times J} \phi(\mu_{C_{(i,j)}}(e_1, e_2)) &= \sum_{(i,j) \in I \times J} \phi(\phi^{-1}(\phi(\mu_{A_i}(e_1)) \cdot \phi(\mu_{B_j}(e_2)))) \\ &= \sum_{i \in I} \phi(\mu_{A_i}(e_1)) \sum_{j \in J} \phi(\mu_{B_j}(e_2)) = 1. \end{aligned}$$

■

3.2.2. Produit cartésien de ϕ -symbolisations

a) Définition

Le produit cartésien de deux ϕ -symbolisations est une ϕ -symbolisation ayant le produit cartésien de leur universs de discours comme universs de discours et le produit cartésien de leur universs symbolique comme universs symbolique. La signification du produit cartésien de deux symboles est alors défini comme le produit cartésien de leurs significations, effectué selon la formule (23).

Proposition 4 : Soit $\langle E_1, S_1, R_1 \rangle$ et $\langle E_2, S_2, R_2 \rangle$ deux ϕ -symbolisations. Soit $R_{1 \times 2}$ la relation floue sur $(E_1 \times E_2) \times (S_1 \times S_2)$ définie comme suit :

$$\begin{aligned} \forall((e_1, e_2), (s_1, s_2)) \in (E_1 \times E_2) \times (S_1 \times S_2), \\ \mu_{R_{1 \times 2}}((e_1, e_2), (s_1, s_2)) = T_P^\phi(\mu_{R_1}(e_1, s_1), \mu_{R_2}(e_2, s_2)), \end{aligned} \quad (24)$$

où T_P^ϕ est la t -norme isomorphe à la t -norme produit (22). Alors la symbolisation $\langle E_1 \times E_2, S_1 \times S_2, R_{1 \times 2} \rangle$ est une ϕ -symbolisation appelée produit cartésien des symbolisations $\langle E, S_1, R_1 \rangle$ et $\langle E, S_2, R_2 \rangle$. $R_{1 \times 2}$ sera notée $T_P^\phi(R_1, R_2)$.

Preuve : La preuve découle directement du produit cartésien de deux ϕ -partitions. ■

b) Propriétés

Etant donné que (24) est symétrique, le produit cartésien de deux ϕ -symbolisations est clairement symétrique. D'autre part, la proposition suivante signifie que le produit cartésien de ϕ -symbolisations est associatif. On peut donc l'appliquer à plus de deux ϕ -symbolisations, indifféremment de l'ordre utilisé pour le réaliser.

Proposition 5 : Soit $\langle E_1, S_1, R_1 \rangle$, $\langle E_2, S_2, R_2 \rangle$ et $\langle E_3, S_3, R_3 \rangle$ trois ϕ -symbolisations.

On a :

$$\begin{aligned} \langle (E_1 \times E_2) \times E_3, (S_1 \times S_2) \times S_3, R_{(1 \times 2) \times 3} \rangle &= \langle E_1 \times (E_2 \times E_3), S_1 \times (S_2 \times S_3), R_{1 \times (2 \times 3)} \rangle \\ &= \langle E_1 \times E_2 \times E_3, S_1 \times S_2 \times S_3, R_{1 \times 2 \times 3} \rangle. \end{aligned} \quad (25)$$

Preuve : Il s'agit de montrer que $T_P^\phi(R_1, T_P^\phi(R_2, R_3)) = T_P^\phi(T_P^\phi(R_1, R_2), R_3)$. Pour tout

triplet $(e_1, e_2, e_3) \in (E_1 \times E_2 \times E_3)$ et tout triplet $(s_1, s_2, s_3) \in (S_1 \times S_2 \times S_3)$, on a :

$$\begin{aligned} \mu_{T_P^\phi(R_1, T_P^\phi(R_2, R_3))}((e_1, e_2, e_3), (s_1, s_2, s_3)) &= T_P^\phi(\mu_{R_1}(e_1, s_1), T_P^\phi(\mu_{R_2}(e_2, s_2), \mu_{R_3}(e_3, s_3))) \\ &= \phi^{-1}(\phi(\mu_{R_1}(e_1, s_1)) \cdot \phi(\phi^{-1}(\phi(\mu_{R_2}(e_2, s_2)) \cdot \phi(\mu_{R_3}(e_3, s_3)))))) \\ &= \phi^{-1}(\phi(\mu_{R_1}(e_1, s_1)) \cdot \phi(\mu_{R_2}(e_2, s_2)) \cdot \phi(\mu_{R_3}(e_3, s_3))) \\ &= \phi^{-1}(\phi(\phi^{-1}(\phi(\mu_{R_1}(e_1, s_1)) \cdot \phi(\mu_{R_2}(e_2, s_2)))) \cdot \phi(\mu_{R_3}(e_3, s_3))) \\ &= T_P^\phi(T_P^\phi(\mu_{R_1}(e_1, s_1), \mu_{R_2}(e_2, s_2)), \mu_{R_3}(e_3, s_3)). \end{aligned}$$

■

3.2.3. Exemple

Pour illustrer la construction du produit cartésien de deux ϕ -symbolisations par une base de règles, nous prenons l'exemple des *Id*-symbolisations des deux capteurs de flexion d'un doigt autre que le pouce (ces doigts sont appelés les grands doigts) : *IP* et *MCP* (voir Fig. 45).



Fig. 45 Les deux capteurs de flexion d'un grand doigt.

Ces deux *Id*-symbolisations ont comme univers de discours respectifs les intervalles $X_{IP} = [0, 2.55]$ et $X_{MCP} = [0, 2.55]$ contenant les mesures des deux capteurs correspondant. Elles utilisent respectivement comme univers lexical les ensembles $L_{IP} = \{tendu, mi_plié, plié\}$ et $L_{MCP} = \{tendu, plié\}$. Les significations qui leur sont associées sont données à la figure 46.

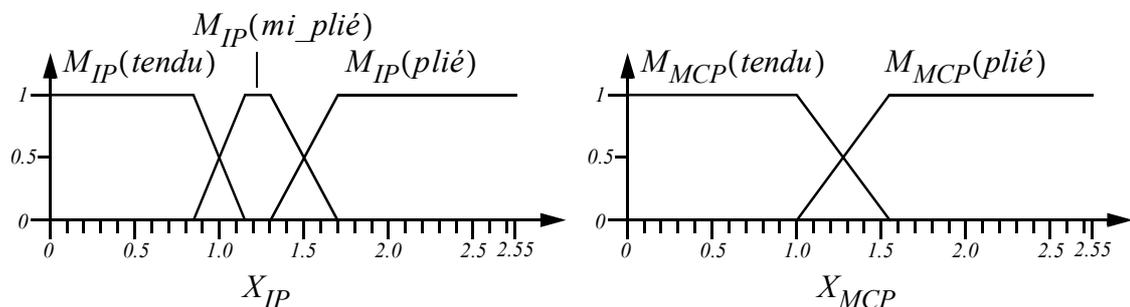


Fig. 46 Significations des symbolisations $\langle X_{IP}, L_{IP}, R_{IP} \rangle$ et $\langle X_{MCP}, L_{MCP}, R_{MCP} \rangle$.

Le produit cartésien de ces deux symbolisations a donc comme univers de discours le plan $X_{IP} \times X_{MCP} = [0, 2.55]^2$ et comme univers lexical l'ensemble $L_{IP} \times L_{MCP}$. Les significations de cette symbolisation produit sont représentées à la figure suivante.

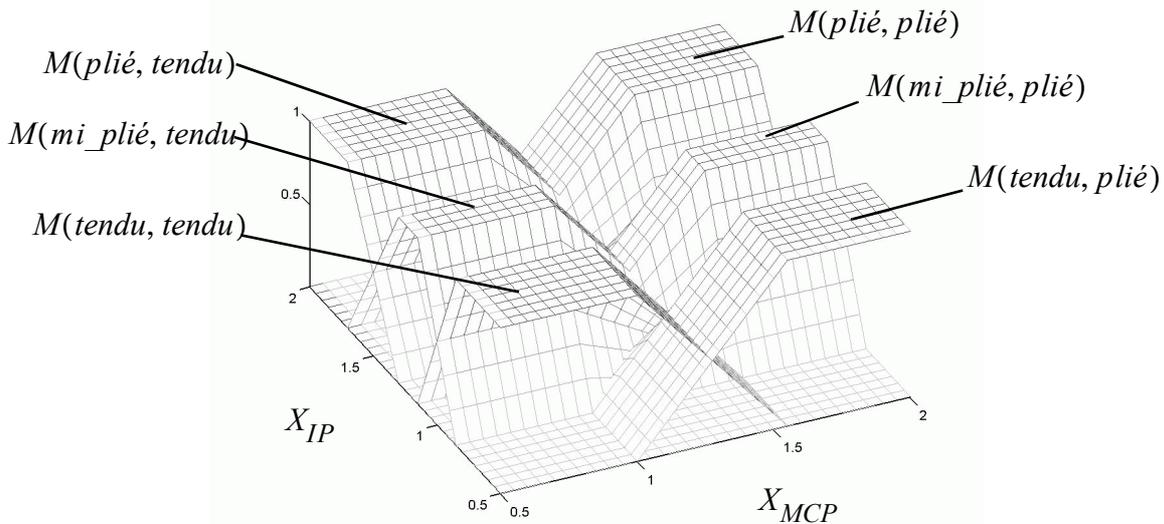


Fig. 47 Significations des éléments de $L_{IP} \times L_{MCP}$ dans la symbolisation produit de $\langle X_{IP}, L_{IP}, R_{IP} \rangle$ et $\langle X_{MCP}, L_{MCP}, R_{MCP} \rangle$ de l'index. L'univers de discours de cette symbolisation est $[0, 2.55]^2$, il est ici tronqué, pour la représentation, à $[0.5, 2.0]^2$.

3.3. Regroupement de symboles d'une ϕ -symbolisation

Avant de présenter comment se construit une ϕ -symbolisation par regroupement de symboles, nous présentons un résultat général concernant la réunion d'éléments d'une ϕ -partition d'un ensemble E .

3.3.1. Réunion d'éléments d'une ϕ -partition

Proposition 6 ([Foulloy 04.c]) : Soit $\mathcal{A} = \{A_i, i \in I\}$ une ϕ -partition d'un ensemble E .

Soit $J = \{J_k, k \in K\}$ une partition (nette) de I . Soit \perp_L^ϕ , la t -conorme isomorphe à la t -conorme de Lukasiewicz \perp_L , définie par :

$$\forall (u, v) \in [0, 1]^2, \quad \perp_L^\phi(u, v) = \phi^{-1}(\perp_L(\phi(u), \phi(v))). \quad (26)$$

Alors la famille de sous-ensembles flous $\mathcal{A}' = \{A'_k, k \in K\}$ définis par :

$$\forall e \in E, \quad \mu_{A'_k}(e) = \perp_{i \in J_k}^\phi \mu_{A_i}(e), \quad (27)$$

est une ϕ -partition de E .

Preuve : On a, pour tout $e \in E$:

$$\sum_{k \in K} \phi(\mu_{A'_k}(x)) = \sum_{k \in K} \phi\left(\perp_{L, J_k}^{\phi} \mu_{A_i}(e)\right) = \sum_{k \in K} \perp_{L, J_k}^{\phi} \phi(\mu_{A_i}(e)) = \sum_{k \in K} \min\left(\sum_{i \in J_k} \phi(\mu_{A_i}(e)), I\right).$$

Comme \mathcal{A} est une ϕ -partition de E , on a $\sum_{i \in J_k} \phi(\mu_{A_i}(e)) \leq I$ et, puisque J est une

$$\text{partition de } I : \sum_{k \in K} \phi(\mu_{A'_k}(x)) = \sum_{k \in K} \sum_{i \in J_k} \phi(\mu_{A_i}(e)) = I. \quad \blacksquare$$

3.3.2. Regroupement de symboles d'une ϕ -symbolisation

Soit $\langle E, S, R \rangle$ une ϕ -symbolisation. Un regroupement de symboles consiste à considérer un ensemble de symboles $s^* \subseteq S$ comme un symbole d'une nouvelle symbolisation sur E . L'univers symbolique \mathfrak{b} de cette nouvelle symbolisation sera alors composé d'ensembles de symboles de S , c'est-à-dire d'éléments de $P(S) : \mathfrak{b} \subset P(S)$. La signification d'un regroupement $s^* \in \mathfrak{b}$ de symboles sera alors définie comme la réunion des significations des symboles le composant, ce dernier étant réalisé suivant la formule (27). Dans ce cas, si \mathfrak{b} est une partition de S , la symbolisation obtenue est une ϕ -symbolisation.

Proposition 7 : Soit $\langle E, S, R \rangle$ une ϕ -symbolisation et \mathfrak{b} une partition de S . Soit $R_{\mathfrak{b}}$ la relation floue sur $E \times \mathfrak{b}$ définie comme suit :

$$\forall (e, s^*) \in E \times \mathfrak{b}, \quad \mu_{R_{\mathfrak{b}}}(e, s^*) = \perp_{L, s^*}^{\phi} \mu_R(e, s). \quad (28)$$

Alors $\langle E, \mathfrak{b}, R_{\mathfrak{b}} \rangle$ est une ϕ -symbolisation.

Preuve : Posons $\mathfrak{b} = \{s_k^*, k \in K\}$. Comme c'est une partition de S et que l'ensemble des significations $\{M(s), s \in S\}$ forme une ϕ -partition de E , alors, d'après la proposition 6, la famille de sous-ensembles flous $\mathcal{A}' = \{A'_k, k \in K\}$ définis par :

$$\forall e \in E, \quad \mu_{A'_k}(e) = \perp_{L, s_k^*}^{\phi} \mu_{M(s)}(e), \quad (29)$$

est une ϕ -partition de E . Or, pour tout $k \in K$, le sous-ensemble flou A'_k n'est autre que la signification de s_k^* dans la symbolisation floue $\langle E, \delta, R_\delta \rangle$. D'autre part, tout $s \in s^*$, possède au moins un prototype, qui est alors également un prototype de s^* . Les significations des symboles dans $\langle E, \delta, R_\delta \rangle$ sont donc normées et forment une ϕ -partition de E . ■

Ainsi, une ϕ -symbolisation $\langle E, S, R \rangle$ utilisant n symboles définit implicitement un ensemble de $B(n)$ ϕ -symbolisations $\langle E, \delta, R_\delta \rangle$ sur E , où $B(n)$ est le nombre de Bell¹. Dans ces ϕ -symbolisations, quelle que soit la partition δ choisie, un sous-ensemble s^* de $P(S)$ aura toujours la même signification sur E , à savoir l'union par l'opérateur \perp_L^ϕ des significations des symboles de s^* . Cette signification sera notée M^* , c'est l'extension de la signification des symboles aux ensembles de symboles :

Définition 14 : Soit $\langle E, S, R \rangle$ une ϕ -symbolisation. La signification associée à cette symbolisation, notée M , est étendue aux ensembles de symboles par la fonction M^* , allant de $P(S)$ dans E :

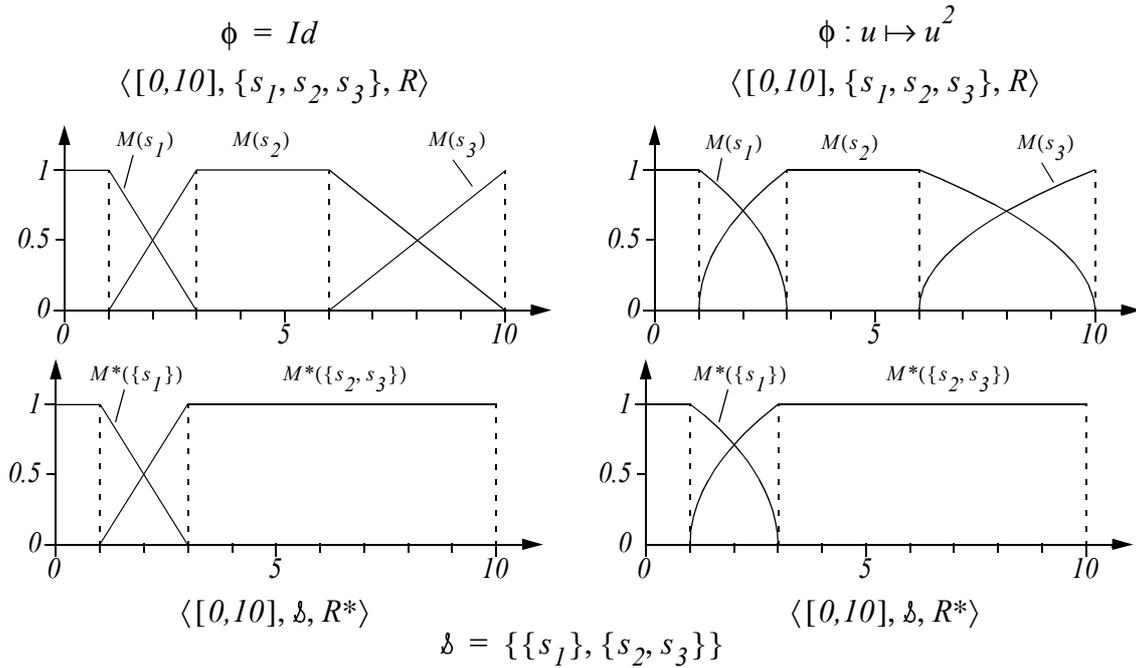
$$\forall s^* \in P(S), \forall e \in E, \quad \mu_{M^*(s^*)}(e) = \perp_{s \in s^*}^\phi \mu_{M(s)}(e). \quad (30)$$

Si s^* est un singleton, i.e. $s^* = \{s\}$, alors $\mu_{M^*(s^*)} = \mu_M(s)$. D'autre part, si δ est une partition de S , la proposition 7 permet de dire que la symbolisation $\langle E, \delta, R^* \rangle$, où R^* est la relation sur $E \times \delta$ issue de M^* , est une ϕ -symbolisation.

Les couples $\langle s^*, M^*(s^*) \rangle$ sont des concepts flous, ayant tous le même univers de discours, E et le même univers lexical, $P(S)$. Une ϕ -symbolisation utilisant n symboles définit donc un ensemble de 2^n concepts différents.

Sur la figure suivante, le regroupement de symboles est illustré pour les symbolisations construites dans la section précédente (voir Fig. 44). L'univers symbolique comportant ici 3 symboles, il y a $B(3) = 5$ ϕ -symbolisations différentes pouvant être réalisées par regroupement de termes. Ces symbolisations sont données en annexe dans le cas $\phi = Id$.

1. Le nombre de Bell $B(n)$ est le nombre de partitions pouvant être construites sur un ensemble à n éléments [Bouvier 96]. Un tableau des nombres de Bell pour $n = 1$ à 15 est donné en annexe.


 Fig. 48 Regroupements de symboles dans des ϕ -symbolisations.

3.3.3. Changement d'univers symbolique d'une ϕ -symbolisation par regroupement de symboles

Soit $\langle E, S_1, R_1 \rangle$ une ϕ -symbolisation et S_2 un ensemble fini de symboles ayant au plus autant d'éléments que S_1 . On peut créer une nouvelle ϕ -symbolisation $\langle E, S_2, R_2 \rangle$ en choisissant une partition δ de S_1 , puis en définissant une fonction bijective f , de δ dans S_2 . On construit alors la ϕ -symbolisation $\langle E, S_2, R_2 \rangle$ en définissant la signification d'un symbole s_2 dans $\langle E, S_2, R_2 \rangle$ comme celle de $f^{-1}(s_2)$ dans $\langle E, \delta, R_1^* \rangle$ qui est, d'après la proposition 7, une ϕ -symbolisation.

$$\forall (e, s_2) \in E \times S_2, \quad \mu_{R_2}(e, s_2) = \mu_{R_1^*}(e, f^{-1}(s_2)). \quad (31)$$

Ce processus est schématisé à la figure suivante. Le choix de la partition δ est équivalent au choix d'une fonction r de regroupement allant de S_1 dans $P(S_1)$ dont l'image est une partition de S_1 . On note alors g la composée de r et de f . Cette fonction g est surjective.

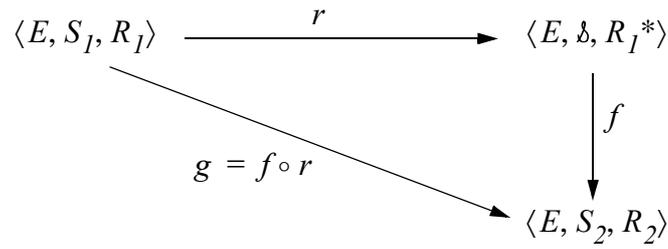


Fig. 49 Principe de création d'une nouvelle symbolisation par regroupement de symboles.

Si on note $g^{-1}(\{s_2\})$ l'ensemble des antécédents de s_2 par g , i.e. $g^{-1}(\{s_2\}) = \{s_1 \in S_1, g(s_1) = s_2\}$, on a :

$$\forall s_2 \in S_2, \quad f^{-1}(s_2) = g^{-1}(\{s_2\}), \quad (32)$$

$$r = f^{-1} \circ g. \quad (33)$$

On peut donc retrouver les fonctions r et f à partir de g et ainsi définir un regroupement de symboles en spécifiant directement cette fonction, en prenant garde qu'elle soit bien surjective. C'est ce que l'on fait généralement, la fonction g étant définie par un ensemble de règles.

3.3.4. Utilisation de règles

a) Règles

La fonction g du schéma de la figure 49 peut être construite par un ensemble de règles de la forme :

$$\text{Si } a \text{ est } s_1^* \text{ Alors } b \text{ est } s_2^*, \quad (34)$$

où $s_1^* \in P(S_1)$ et $s_2^* \in P(S_2)$. « a est s_1^* » est la prémisse de la règle et « b est s_2^* » est la conclusion de la règle. L'ensemble des règles forme une base de règles. On notera $\delta_1 \subset P(S_1)$ l'ensemble des s_1^* des prémisses et $\delta_2 \subset P(S_2)$ l'ensemble des s_2^* des conclusions. On ne considérera que des bases de règles où les éléments de δ_2 sont tous des singletons. La base de règles est dite complète si δ_1 est une partition de S_1 . Elle est dite surjective si tout élément de S_2 apparaît au moins une fois dans la conclusion d'une règle.

On utilise ici une représentation symbolique des règles [Foulloy 95] : a et b , dans (34), sont des variables prenant respectivement comme valeurs des symboles de S_1 et S_2 . L'opérateur « *est* » sera interprété comme une relation d'appartenance :

$$a \text{ est } s_1^* \Leftrightarrow a \in s_1^*. \quad (35)$$

La règle (34) définit alors une relation nette sur $S_1 \times S_2$ qui dépend de l'interprétation que l'on donne à la partie « *Si ... Alors ...* » de la règle. On a le choix entre l'interprétation conjonctive et disjonctive. Ces interprétations sont différentes pour une règle isolée mais identiques pour la base de règles prise dans son ensemble lorsque celle-ci est complète. Dans ce cas, la base de règles définit au final une seule relation sur $S_1 \times S_2$. Les éléments de \mathfrak{S}_2 étant tous des singletons, cette relation est le graphe d'une application g de S_1 dans S_2 . Si $\{s_2\}$ est la conclusion de la règle dont la prémisse est « *a est s_1^** », on a :

$$\forall s_1 \in s_1^*, \quad g(s_1) = s_2, \quad (36)$$

Si la base de règles est surjective, il en est de même de la fonction g qui peut donc être utilisée pour définir une nouvelle symbolisation par regroupement de termes.

Dans le cas où la symbolisation $\langle E, S_1, R_1 \rangle$ dont on veut regrouper les symboles est un produit cartésien de ϕ -symbolisations, l'univers symbolique S_1 est lui-même un produit cartésien d'univers symboliques. La fonction g est alors une fonction à plusieurs variables et les règles la définissant seront des règles à prémisses complexes. En effet, si $\langle E, S_1, R_1 \rangle$ est le produit cartésien de deux ϕ -symbolisations $\langle E_1, A, R_A \rangle$ et $\langle E_2, B, R_B \rangle$, les règles utilisées pour spécifier la fonction g auront la forme suivante :

$$\text{Si } a \text{ est } a^* \text{ et } b \text{ est } b^* \text{ Alors } c \text{ est } s_2^*, \quad (37)$$

où a^* , b^* et s_2^* sont respectivement des éléments de $P(A)$, $P(B)$ et $P(S_2)$. Avec l'interprétation choisie pour l'opérateur *est*, on a l'équivalence suivante :

$$a \text{ est } a^* \text{ et } b \text{ est } b^* \Leftrightarrow (a, b) \text{ est } a^* \times b^*. \quad (38)$$

Ainsi, la règle (37) est bien équivalente à une règle de la forme (34) et elle peut être utilisée pour définir une fonction g de $A \times B$ dans C .

b) Notations

Etant donné que l'on peut étendre la signification des symboles d'une ϕ -symbolisation aux éléments de $P(S)$. Nous allons adopter une notation permettant de rendre explicite le fait que les éléments de $P(S)$ sont non seulement des ensembles de symboles mais sont eux-mêmes des symboles ayant une signification. On peut comparer cela au fait qu'une partie d'un ensemble A est non seulement un ensemble, mais également un élément de $P(A)$.

Les singletons de $P(S)$ seront notés comme l'unique élément qu'ils contiennent, mais en caractère gras :

$$\forall s \in S, \quad \{s\} = \mathbf{s}. \quad (39)$$

L'ensemble S lui-même est un élément particulier de $P(S)$ qui sera défini comme étant le symbole *quelconque* :

$$S = \mathbf{quelconque} \quad (40)$$

Les autres éléments de $P(S)$ seront notés en utilisant l'opérateur *ou*, qui n'est autre que l'union sur $P(S)$:

$$\forall (s_1^*, s_2^*) \in P(S)^2, \quad s_1^* \mathbf{ou} s_2^* = s_1^* \cup s_2^*. \quad (41)$$

Ainsi, si $S = \{s_1, \dots, s_n\}$, on aura :

$$\{s_{i_1}, s_{i_2}, \dots, s_{i_p}\} = \mathbf{s_{i_1} ou s_{i_2} ou \dots ou s_{i_p}}. \quad (42)$$

On a alors les propriétés suivantes :

$$\forall s \in S, \quad \mu_{M^*}(s) = \mu_M(s), \quad (43)$$

$$\mu_{M^*}(\mathbf{quelconque}) = 1, \quad (44)$$

$$\forall (s_1^*, s_2^*) \in P(S)^2, s_1^* \cap s_2^* = \emptyset, \mu_{M^*}(s_1^* \mathbf{ou} s_2^*) = \perp_L^\phi(\mu_{M^*}(s_1^*), \mu_{M^*}(s_2^*)). \quad (45)$$

Cette dernière propriété s'écrit, dans le cas des *Id*-symbolisations :

$$\forall (s_1^*, s_2^*) \in P(S)^2, s_1^* \cap s_2^* = \emptyset \quad \mu_{M^*}(s_1^* \mathbf{ou} s_2^*) = \mu_{M^*}(s_1^*) + \mu_{M^*}(s_2^*). \quad (46)$$

c) Exemple

Introduisons tout d'abord un ensemble lexical particulier qui sera utilisé pour la symbolisation de la flexion d'un grand doigt, c'est-à-dire d'un doigt autre que le pouce :

$$L_{flexionDoigt} = \{tendu, carré, rond, griffe, plié\}. \quad (47)$$

Chacun de ces termes correspond à une configuration particulière d'un grand doigt :

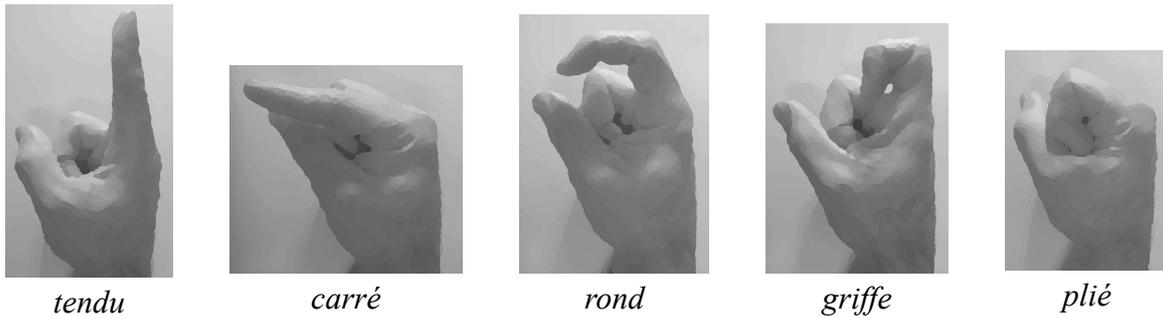


Fig. 50 Les cinq configurations d'un grand doigt.

Une base de règles est utilisée pour construire la symbolisation de la flexion d'un doigt à partir des symbolisations $\langle X_{IP}, L_{IP}, R_{IP} \rangle$ et $\langle X_{MCP}, L_{MCP}, R_{MCP} \rangle$ des deux articulations de ce doigt. La base de règles utilisée pour l'agrégation de ces deux *Id*-symbolisations est la même pour les quatre grands doigts :

- (1) Si *IP* est **tendu** et *MCP* est **tendu** Alors *flexionDoigt* est **tendu**
- (2) Si *IP* est **tendu** et *MCP* est **plié** Alors *flexionDoigt* est **carré**
- (3) Si *IP* est **mi_plié** et *MCP* est **tendu** Alors *flexionDoigt* est **rond**
- (4) Si *IP* est **mi_plié** et *MCP* est **plié** Alors *flexionDoigt* est **rond**
- (5) Si *IP* est **plié** et *MCP* est **tendu** Alors *flexionDoigt* est **griffe**
- (6) Si *IP* est **plié** et *MCP* est **plié** Alors *flexionDoigt* est **plié**

Fig. 51 Base de règles utilisée pour l'agrégation des symbolisations $\langle X_{IP}, L_{IP}, R_{IP} \rangle$ et $\langle X_{MCP}, L_{MCP}, R_{MCP} \rangle$.

Cette base de règles définit une fonction g surjective, de $L_{IP} \times L_{MCP}$ dans $L_{flexionDoigt}$, donnée à la figure 52.

L_{MCP} L_{IP}	<i>plié</i>	<i>plié</i>
<i>plié</i>	<i>plié</i>	<i>griffe</i>
<i>mi_plié</i>	<i>rond</i>	<i>rond</i>
<i>tendue</i>	<i>carré</i>	<i>tendu</i>

Fig. 52 Fonction $g : L_{IP} \times L_{MCP} \rightarrow L_{flexionDoigt}$ utilisée pour la construction de la ϕ -symbolisation $\langle X_{flexionDoigt}, L_{flexionDoigt}, R_{flexionDoigt} \rangle$ à partir de $\langle X_{IP}, L_{IP}, R_{IP} \rangle$ et $\langle X_{MCP}, L_{MCP}, R_{MCP} \rangle$.

Le résultat de l'agrégation de $\langle X_{IP}, L_{IP}, R_{IP} \rangle$ et $\langle X_{MCP}, L_{MCP}, R_{MCP} \rangle$ par cette fonction est la symbolisation $\langle X_{flexionDoigt}, L_{flexionDoigt}, R_{flexionDoigt} \rangle$, avec $X_{flexionDoigt} = X_{IP} \times X_{MCP}$, dont les significations sont données, dans le cas de l'index, à la figure 53.

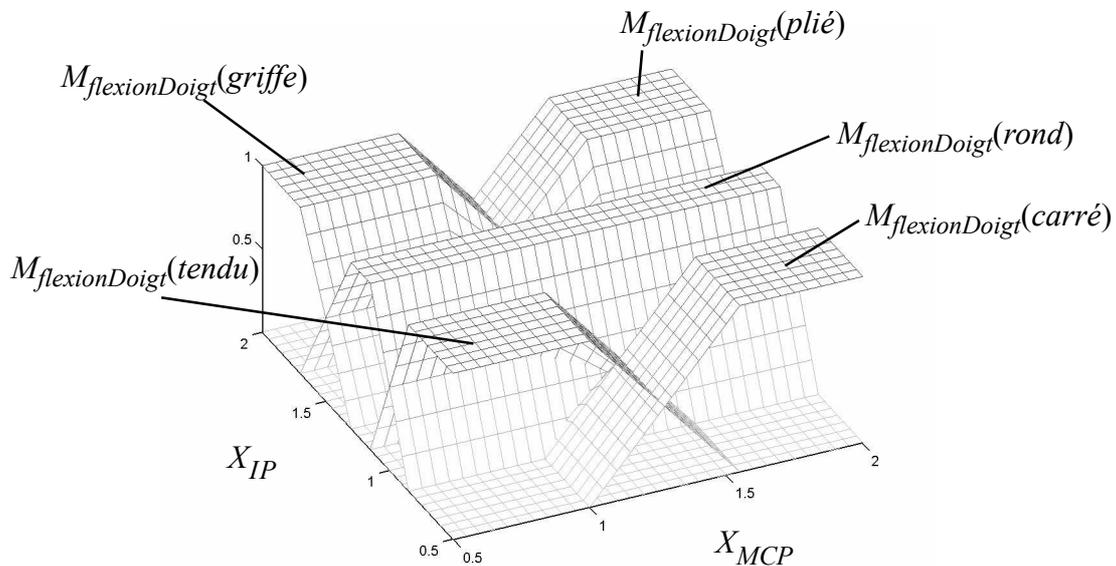


Fig. 53 Significations des termes de $L_{flexionDoigt}$ pour la Id-symbolisation $\langle X_{flexionDoigt}, L_{flexionDoigt}, R_{flexionDoigt} \rangle$ issue de l'agrégation de $\langle X_{IP}, L_{IP}, R_{IP} \rangle$ et $\langle X_{MCP}, L_{MCP}, R_{MCP} \rangle$ dans le cas de l'index. L'univers de discours de cette symbolisation est $[0, 2.55]^2$, il a ici été tronqué, pour la représentation, à $[0.5, 2.0]^2$.

Remarquons que la base de règles de la figure 51 est équivalente à celle, plus compacte, de la figure suivante.

-
- (1) Si IP est **tendu** et MCP est **tendu** Alors $flexionDoigt$ est **tendu**
 - (2) Si IP est **tendu** et MCP est **plié** Alors $flexionDoigt$ est **carré**
 - (3) Si IP est **mi_plié** et MCP est **tendu ou plié** Alors $flexionDoigt$ est **rond**
 - (4) Si IP est **plié** et MCP est **tendu** Alors $flexionDoigt$ est **griffe**
 - (5) Si IP est **plié** et MCP est **plié** Alors $flexionDoigt$ est **plié**

Fig. 54 Base de règles équivalente à la base de règles de la figure 51, mais plus compacte.

La condition « MCP est **tendu ou plié** » de la règle (3) de la figure 54 peut aussi s'écrire « MCP est **quelconque** ». Une telle condition peut être retirée de la base de règles car la règle (3') suivante :

$$(3') \text{ Si } IP \text{ est } \mathbf{mi_plié} \text{ Alors } flexionDoigt \text{ est } \mathbf{rond}, \quad (48)$$

définit exactement la même relation sur $L_{IP} \times L_{MCP} \times L_{flexionDoigt}$.

3.4. Conclusion - forme des significations obtenues

Nous avons introduit des outils permettant de construire des Id -symbolisations et plus généralement des ϕ -symbolisations. Comme cela a été précisé au début de ce paragraphe, les principes de construction présentés ici ne sont pas les uniques moyens de construire une ϕ -symbolisation sur un produit cartésien d'intervalles réels. Ils correspondent à notre choix d'utiliser une approche lexicale pour cette construction. Les bases de règles peuvent être interprétées par une personne non spécialiste. Elles peuvent être modifiées facilement pour améliorer les performances du système ou adapter son comportement à un certain besoin.

Mais cette construction conduit à des significations d'une forme particulière : leurs noyaux sont des produits cartésiens d'intervalles réels. Sur la figure 53 par exemple, les significations des termes de $L_{flexionDoigt}$ ont des noyaux qui sont des rectangles du plan $X_{IP} \times X_{MCP}$. Cette forme des significations n'est pas toujours adaptée à la forme des prototypes des symboles. La figure 55, donne un exemple où quatre ensembles de prototypes doivent être inclus dans les noyaux de quatre significations. Les ensembles de prototypes ont des formes elliptiques et ne sont pas orientés parallèlement aux axes de l'univers de discours. Si l'on essaie de construire une symbolisation sur $X_1 \times X_2$ en agrégeant deux symbolisations intermédiaires préalablement définies sur X_1 et X_2 , le nombre de symboles à utiliser pour ces symbolisations

intermédiaires est très important et les significations obtenues au final correspondent mal aux prototypes. Une projection préalable des données sur les axes Y_1 et Y_2 (voir Fig. 55.b), obtenus par exemple par une analyse en composantes discriminantes [Saporta 90], permet de réduire considérablement le nombre de symboles à utiliser dans les symbolisations intermédiaires et de construire des significations finales dont les noyaux correspondent mieux aux prototypes.

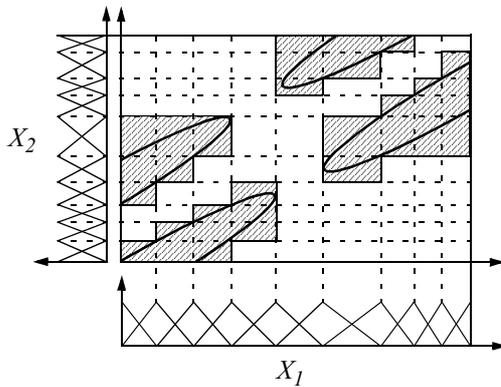


Fig. 55.a Construction des symbolisations intermédiaires sur les données brutes

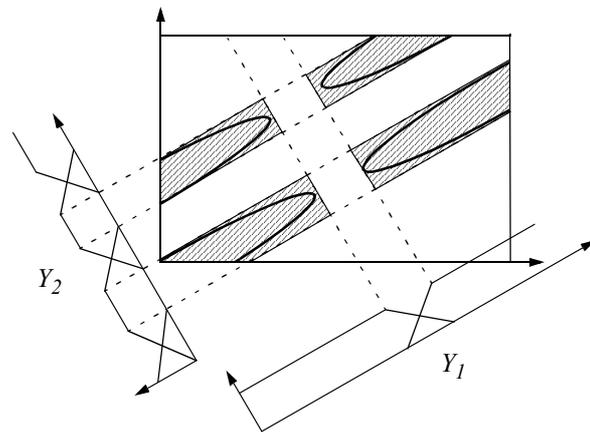


Fig. 55.b Construction des symbolisations intermédiaires sur les données projetées sur les axes discriminants

Fig. 55 Pour pouvoir utiliser les techniques de construction de ϕ -symbolisations décrites dans cette section, il faut parfois projeter préalablement les données sur des axes discriminants.

Nous allons maintenant voir que tout *Id*-symbolisation, quelle que soit la façon dont elle a été construite, est liée à une relation d'équivalence floue permettant de comparer les entités entre elles. Des résultats similaires existent aussi pour certaines ϕ -symbolisations et nous les énoncerons également.

4. Relations d'équivalence floues associées à une symbolisation.

Les ϕ -symbolisations permettent de représenter des données sous la forme de sous-ensembles flous de symboles appelés les descriptions floues. Une fois une ϕ -symbolisation définie, seules ses descriptions floues sont manipulées car ce sont elles qui portent toute l'information utile sur les entités qu'elles représentent. Dans ce chapitre, nous présentons une relation d'équivalence floue définie sur les descriptions d'une *Id*-symbolisation permettant de comparer les descriptions et donc de traiter des données se présentant sous cette forme. Nous allons voir que cette relation d'équivalence floue sur les descriptions induit une relation d'équivalence floue sur l'univers de discours. Cette dernière donne donc un degré de similarité de deux entités selon leur description floue.

Nous avons précisé dans la section précédente que les fonctions d'appartenance des sous-ensembles flous ont ici une sémantique de similarité. Cela signifie que l'appartenance d'un élément à la signification d'un symbole est égale à la similarité de cet élément avec un prototype de ce symbole. Plus formellement, étant donnée une ϕ -symbolisation $\langle E, S, R \rangle$, une relation de similarité *sim* sur l'univers de discours sera en accord avec la sémantique des fonctions d'appartenance des significations si elle vérifie, pour tout symbole s de S et tout prototype p de s :

$$\forall e \in E, \quad \mu_{M(s)}(e) = \mu_{sim}(e, p). \quad (49)$$

Nous montrerons que la relation d'équivalence floue utilisée pour comparer les descriptions floues d'une *Id*-symbolisation vérifie bien (49). Nous nous interrogerons également sur la possibilité de définir une telle relation pour les autres ϕ -symbolisations ou d'utiliser d'autres relations floues pour les *Id*-symbolisations.

Nous commençons par rappeler ce que sont les relations d'équivalences floues, encore appelées *T*-équivalences.

4.1. *T*-équivalences et *T*-égalité

4.1.1. Définitions

Définition 15 ([De Baets 98]) : *Soit une t-norme T et un ensemble U . Une relation floue R sur U est une T -équivalence si et seulement si elle vérifie les trois propriétés suivantes :*

$$(i) - \text{Réflexivité} : \quad \forall u \in U, \quad \mu_R(u, u) = 1. \quad (50)$$

$$(ii) - \text{Symétrie} : \quad \forall (u, v) \in U \times U, \quad \mu_R(u, v) = \mu_R(v, u). \quad (51)$$

$$(iii) - T\text{-transitivité} : \quad \forall (u, v, w) \in U \times U \times U, \quad (52)$$

$$T(\mu_R(u, v), \mu_R(v, w)) \leq \mu_R(u, w)$$

Notons que la propriété de T -transitivité est une version floue de la propriété de transitivité nette :

$$\forall (u, v, w) \in U^3, \quad (u, v) \in R \wedge (v, w) \in R \Rightarrow (u, w) \in R. \quad (53)$$

Les T -équivalences sont donc des généralisations floues des relations d'équivalence nettes. Elles sont aussi appelées relations d'indistinguabilité [Valverde 97] ou encore relations d'équivalence floues [Boixader 00]. Elles ont été introduites, dans le cas de la t -norme *min*, par L.A. Zadeh [Zadeh 71.b].

Parmi les T -équivalences, les T -égalités forment une famille particulièrement intéressante :

Définition 16 ([De Baets 02]) : *Soit une t -norme T et un ensemble U . Une relation floue R sur U est une T -égalité si et seulement si c'est une T -équivalence et qu'elle vérifie :*

$$(iv) \quad \forall (u, v) \in U \times U, \quad \mu_R(u, v) = 1 \Leftrightarrow (u = v). \quad (54)$$

Les T -égalités et T -équivalences sont en rapport très étroit avec les métriques généralisées que nous introduisons maintenant.

4.1.2. Métriques généralisées et T -équivalences

Rappelons tout d'abord la définition des pseudo-métriques :

Définition 17 ([De Baets 02]) : *Soit U un ensemble. Une fonction d , de U sur $[0, +\infty]$ est une pseudo-métrique si et seulement si elle vérifie les trois propriétés suivantes :*

$$(i) \quad \forall u \in U, \quad d(u, u) = 0. \quad (55)$$

$$(ii) - \text{Symétrie} : \quad \forall (u, v) \in U \times U, \quad d(u, v) = d(v, u). \quad (56)$$

$$(iii) - \text{Inégalité Triangulaire} : \quad \forall (u, v, w) \in U \times U \times U, \quad (57)$$

$$d(u, w) \leq d(u, v) + d(v, w)$$

Les pseudo-métriques sont donc des métriques ne vérifiant qu'une partie de l'axiome de séparation. Les métriques sont des pseudo-métriques particulières.

Proposition 8 ([De Baets 02]) : *Soit U un ensemble. Une fonction d , de U sur $[0, +\infty]$ est une métrique si et seulement si c'est une pseudo-métrique et qu'elle vérifie la propriété suivante :*

$$(iv) - \text{Axiome de séparation} \quad \forall (u, v) \in U \times U, \quad d(u, v) = 0 \Leftrightarrow (u = v). \quad (58)$$

Il existe un lien entre T -équivalences et pseudo-métriques, d'une part et T -égalités et métriques, d'autre part. Ce lien se fait par l'intermédiaire des générateurs additifs. Pour une présentation complète des t -normes et notamment de la notion de générateur additif, on pourra se reporter à [Klement 00].

Proposition 9 ([De Baets 02]) : *Soit un ensemble U et T une t -norme. Alors, pour toute t -norme T^* possédant un générateur additif t^* et telle que $T^* \leq T$ et toute T -équivalence R sur U , la fonction $d = t^* \circ R$ est une pseudo-métrique sur U .*

On a alors le corollaire suivant :

Corollaire 1 ([De Baets 02]) : *Soit U un ensemble et T une t -norme ayant un générateur additif t , alors, pour toute T -équivalence R sur U , la fonction $d = t \circ R$ est une pseudo-métrique sur U .*

Dans le cas des T -égalités, on a la proposition suivante :

Proposition 10 ([De Baets 02]) : *Soit un ensemble U et T une t -norme. Alors, pour toute t -norme T^* possédant un générateur additif t^* et telle que $T^* \leq T$ et toute T -égalité R sur U , la fonction $d = t^* \circ R$ est une métrique sur U .*

On a dans ce cas le corollaire suivant :

Corollaire 2 ([De Baets 02]) : *Soit un ensemble U et T une t -norme ayant un générateur additif t , alors, pour toute T -égalité E sur U , la fonction $d = t \circ E$ est une métrique sur U .*

On trouvera une présentation détaillée des T -équivalences dans [Boixader 00]. Nous présentons tout d'abord un résultat permettant de lier les T -équivalences sur les descriptions d'une symbolisation floue aux T -équivalences sur l'univers de discours de cette symbolisation.

4.1.3. T -équivalences et symbolisations floues

Définition 18 : Soit $\langle E, S, R \rangle$ une symbolisation floue, D la description associée à cette symbolisation et R_S une relation floue sur l'ensemble des descriptions de cette symbolisation. Alors la relation $R_S \circ (D \times D)$ sur l'univers de discours définie par :

$$\forall (e_1, e_2) \in E \times E, \quad \mu_{R_S \circ (D \times D)}(e_1, e_2) = \mu_{R_S}(D(e_1), D(e_2)), \quad (59)$$

est la relation induite sur E par la relation R_S .

Lorsque R_S est une relation d'équivalence floue, il en est de même de $R_S \circ (D \times D)$.

Proposition 11 : Soit $\langle E, S, R \rangle$ une symbolisation floue, D la description associée à cette symbolisation et R_S une T -équivalence sur l'ensemble des descriptions de cette symbolisation. Alors la relation $R_S \circ (D \times D)$ induite sur l'univers de discours par R_S est une T -équivalence sur E .

Preuve : La réflexivité, la symétrie et la T -transitivité de la relation R_S se transposent directement à $R_S \circ (D \times D)$. ■

Par contre, si R_S est une T -égalité, il n'en est généralement pas de même de $R_S \circ (D \times D)$ puisque l'on peut avoir $D(e_1) = D(e_2)$ avec $e_1 \neq e_2$.

Nous introduisons maintenant une famille de relations sur des sous-ensembles flous de symboles qui, lorsqu'elles sont appliquées aux descriptions d'une ϕ -symbolisation, induisent une relation sur l'univers de discours vérifiant la propriété (49) : ce sont les relations $(\perp - T^*)$. Nous verrons sous quelles conditions certaines de ces relations sont des T -équivalences.

4.2. Relations $(\perp_L - T^*)$

Si S est un ensemble fini (de symboles), une relation $(\perp - T^*)$ est une relation sur $F(S)$:

Définition 19 : Soit S un ensemble fini de symboles, \perp une t -conorme et T^* une t -norme. La relation $(\perp - T^*)$ est une relation sur $F(S)$ définie comme suit :

$$\forall (A, B) \in F(S) \times F(S), \quad \mu_{(\perp - T^*)}(A, B) = \bigwedge_{s \in S} T^*(\mu_A(s), \mu_B(s)). \quad (60)$$

Si S est l'univers symbolique d'une symbolisation floue $\langle E, S, R \rangle$, la relation $(\perp - T^*)$ induit sur E la relation $(\perp - T^*) \circ (D \times D)$ qui vérifie la propriété (49).

Proposition 12 : Soit $\langle E, S, R \rangle$ une symbolisation floue, \perp une t -conorme et T^* une t -norme. On note $(\perp - T^*) \circ (D \times D)$ la relation induite sur E par $(\perp - T^*)$. Soit s_θ un symbole quelconque de S et p_θ un prototype de ce symbole. On a alors :

$$\forall e \in E, \quad \mu_{M(s_\theta)}(e) = \mu_{(\perp - T^*) \circ (D \times D)}(e, p_\theta) \quad (61)$$

Preuve : L'entité p_θ étant un prototype du symbole s_θ , sa description est un singleton :

$$D(p_\theta) = \{s_\theta\}. \text{ Alors :}$$

$$\begin{aligned} \mu_{(\perp - T^*) \circ (D \times D)}(e, p_\theta) &= \mu_{(\perp - T^*)}(D(e), D(s_\theta)) \\ &= \bigwedge_{s \in S} T^*(\mu_{D(e)}(s), \mu_{\{s_\theta\}}(s)) \\ &= \mu_{D(e)}(s_\theta) = \mu_{M(s_\theta)}(e). \quad \blacksquare \end{aligned}$$

Ainsi, les relations $(\perp - T^*)$ sont compatibles avec la sémantique des fonctions d'appartenance. De plus, puisque une t -norme est symétrique, les relations $(\perp - T^*)$ sont toujours symétriques. Mais rien n'indique qu'elles soient réflexives ou T -transitives (T pouvant être différente de T^*). Des résultats existent cependant pour certaines de ces relations lorsqu'elles sont associées à des ϕ -symbolisations. Un résultat concernant les Id -symbolisations nous concerne plus particulièrement et est présenté au paragraphe suivant.

4.3. Relations $(\perp_L - T_M)$ et Id -symbolisations

4.3.1. Relations $(\perp_L - T_M)$: définition et propriétés

Définition 20 : Soit un ensemble fini de symboles S , \perp_L la t -conorme de Lukasiewicz et T_M la t -norme min. La relation $(\perp_L - T_M)$ est une relation floue sur $F(S)$ définie par la fonction d'appartenance suivante :

$$\forall (A, B) \in F(S) \times F(S), \quad \mu_{(\perp_L - T_M)}(A, B) = \bigwedge_{s \in S} T_M(\mu_A(s), \mu_B(s)). \quad (62)$$

La relation $(\perp_L - T_M)$ est symétrique, comme toute relation $(\perp - T^*)$. Elle est réflexive sur une partie de $F(S)$, qu'on note $F_{Id}(S)$:

$$\forall A \in F(S), \quad A \in F_{Id}(S) \Leftrightarrow \mu_{(\perp_L - T_M)}(A, A) = I. \quad (63)$$

On peut facilement montrer que $F_{Id}(S)$ s'écrit également comme suit :

$$\forall A \in F(S), \quad A \in F_{Id}(S) \Leftrightarrow \sum_{s \in S} \mu_A(s) = I. \quad (64)$$

La notation $F_{Id}(S)$ s'explique par le fait que la somme de l'équation (64) peut s'écrire :

$\sum_{s \in S} Id(\mu_A(s)) = I$. Il a été montré que la relation $(\perp_L - T_M)$ n'est pas seulement réflexive sur $F_{Id}(S)$, mais qu'elle également est T_L -transitive, T_L étant la t-norme de Lukasiewicz et qu'elle vérifie la propriété (iv) des T_L -égalités (équation (54)). C'est donc une T_L -égalité sur $F_{Id}(S)$.

Proposition 13 ([Foulloy 04.a]) : *Soit S un ensemble fini de symboles et $F_{Id}(S)$ la famille des sous-ensembles flous de S définie par (64). La relation $(\perp_L - T_M)$ est une T_L -égalité sur $F_{Id}(S)$, T_L étant la t-norme de Lukasiewicz.*

De plus T_L possède le générateur additif $t_L : u \mapsto I - u$. D'après le corollaire 2 de la proposition 10, l'opérateur $d_{(\perp_L - T_M)}$ suivant est alors une métrique sur $F_{Id}(S)$:

$$\forall (A, B) \in F_{Id}(S) \times F_{Id}(S), \quad d_{(\perp_L - T_M)}(A, B) = I - \mu_{(\perp_L - T_M)}(A, B). \quad (65)$$

Enfin, on peut montrer que la relation $(\perp_L - T_M)$ prend la forme suivante sur $F_{Id}(S)$:

$$\forall (A, B) \in F_{Id}(S) \times F_{Id}(S), \quad \mu_{(\perp_L - T_M)}(A, B) = \sum_{s \in S} \min(\mu_A(s), \mu_B(s)). \quad (66)$$

Les relations $(\perp_L - T_M)$ sont à associer aux Id -symbolisations car si S est l'univers symbolique d'une Id -symbolisation, la description de tout élément de l'univers de discours

appartient à $F_{Id}(S)$. La relation $(\perp_L - T_M)$ sera donc toujours une T_L -égalité sur l'ensemble des descriptions d'une Id -symbolisation.

4.3.2. Relations $(\perp_L - T_M)$ et Id -symbolisations

a) Propriétés

Proposition 14 ([Foulloy 04.a]) : Soit $\langle E, S, R \rangle$ une Id -symbolisation. La relation $(\perp_L - T_M)$ est une T_L -égalité sur l'ensemble des descriptions de $\langle E, S, R \rangle$, T_L étant la t -norme de Lukasiewicz.

La relation $(\perp_L - T_M) \circ (D \times D)$ induite sur l'univers de discours est, elle, une T_L -équivalence. Elle prend la forme suivante :

$$\forall (e_1, e_2) \in E^2, \quad \mu_{(\perp_L - T_M) \circ (D \times D)}(e_1, e_2) = \sum_{s \in S} \min(\mu_{D(e_1)}(s), \mu_{D(e_2)}(s)), \quad (67)$$

où D est la description associée à la symbolisation. L'opérateur $d_{(\perp_L - T_M)}$ défini par (65) est donc bien une métrique sur l'ensemble des descriptions d'une Id -symbolisation. Elle induit une pseudo-métrique sur l'univers de discours.

Finalement, la construction d'une Id -symbolisation permet non seulement de rendre les données plus accessibles et d'utiliser une approche experte pour les manipuler, mais elle fournit également une T_L -équivalence permettant de comparer les entités entre elles. On peut indifféremment utiliser cette T_L -équivalence ou la pseudo-métrique associée pour effectuer cette comparaison.

b) Comparaison d'entités provenant de deux Id -symbolisations

Si deux Id -symbolisations $\langle E_1, S, R_1 \rangle$ et $\langle E_2, S, R_2 \rangle$ ont des univers de discours différents mais utilisent le même univers symbolique, on peut comparer deux entités, provenant des deux univers de discours, en comparant leurs descriptions avec la relation $(\perp_L - T_M)$, on crée ainsi une relation sur $E_1 \times E_2$:

$$\forall (e_1, e_2) \in E_1 \times E_2, \quad \mu_{(\perp_L - T_M) \circ (D_1 \times D_2)}(e_1, e_2) = \mu_{(\perp_L - T_M)}(D_1(e_1), D_2(e_2)), \quad (68)$$

D_1 et D_2 étant les significations respectivement associées aux symbolisations $\langle E_1, S, R_1 \rangle$ et $\langle E_2, S, R_2 \rangle$.

Cette comparaison d'entités provenant d'espaces différents est une nouvelle forme d'agrégation qui est offerte par les symbolisations. Ces dernières ont, dans ce cas, un rôle normatif : elles ramènent les entités sous une même échelle permettant ainsi leur comparaison. Ce type d'agrégation sera utilisé dans nos applications : on a en effet quatre grands doigts dont les descriptions sont définies sur le même espace lexical $L_{flexionDoigt} = \{tendu, carré, rond, griffe, plié\}$. On peut alors comparer leurs descriptions pour, par exemple, vérifier que ces quatre doigts se trouvent dans la même configuration.

On illustre cette agrégation sur la comparaison des descriptions de l'index et du majeur. Les *Id*-symbolisations utilisées pour la description de la flexion de chacun de ces doigts sont respectivement : $\langle X_{flexionIndex}, L_{flexionDoigt}, R_{flexionIndex} \rangle$ et $\langle X_{flexionMajeur}, L_{flexionDoigt}, R_{flexionMajeur} \rangle$. Les univers de discours de ces symbolisations sont les produits cartésiens des deux capteurs mesurant la configuration du doigt considéré : $X_{flexionIndex} = X_{IP_{Index}} \times X_{MCP_{Index}}$ et $X_{flexionMajeur} = X_{IP_{Majeur}} \times X_{MCP_{Majeur}}$. Bien que ces univers de discours soient différents, l'égalité des univers lexicaux permet de comparer la flexion de l'index et du majeur. La figure suivante donne le résultat de cette comparaison pour trois configurations différentes de ces deux doigts.



$$\mu_{(\perp_L - T_M)}(D_{flexionIndex}(x_{flexionIndex}), D_{flexionMajeur}(x_{flexionMajeur})) = 1.0$$



$$\mu_{(\perp_L - T_M)}(D_{flexionIndex}(x_{flexionIndex}), D_{flexionMajeur}(x_{flexionMajeur})) = 0.85$$



$$\mu_{(\perp_L - T_M)}(D_{flexionIndex}(x_{flexionIndex}), D_{flexionMajeur}(x_{flexionMajeur})) = 0.68$$

Fig. 56 Comparaison des descriptions de la flexion de l'index et du majeur. Les deux Id-symbolisations utilisant le même ensemble lexical, on peut comparer leurs descriptions avec la T_L -égalité $(\perp_L - T_M)$.

c) Limites de la relation $(\perp_L - T_M)$

La relation $(\perp_L - T_M)$ permet de mesurer la similarité entre deux entités selon leurs descriptions dans une *Id*-symbolisation. Cette T_L -équivalence reste cependant limitée car elle est toujours égale à 0 si les supports des descriptions que l'on compare sont disjoints :

Proposition 15 : Soit $\langle E, S, R \rangle$ une *Id*-symbolisation et e_1 et e_2 deux entités de E . On a alors :

$$\mu_{(\perp_L - T_M)}(e_1, e_2) = 0 \Leftrightarrow \text{supp}(D(e_1)) \cap \text{supp}(D(e_2)) = \emptyset. \quad (69)$$

Preuve : Si $\text{supp}(D(e_1)) \cap \text{supp}(D(e_2)) = \emptyset$, alors pour tout symbole $s \in S$, $\min(\mu_{D(e_1)}(s), \mu_{D(e_2)}(s)) = 0$, d'où le résultat. ■

Pour un tel couple d'entités, la relation $(\perp_L - T_M)$ ne fournit donc pas plus d'information que le fait de savoir que leurs supports sont disjoints. Par contre, dès que les supports ont une intersection non nulle, elle fournit une information graduelle, comprise entre 0 et 1, sur la similarité des deux entités dans la *Id*-symbolisation. Mais même dans ce cas, elle peut donner des résultats inattendus. Pour illustrer cela, nous prenons un exemple sortant du cadre de la description de la main, mais mettant mieux le paradoxe en relief.

Considérons une *Id*-symbolisation permettant de décrire la température d'une eau et utilisant l'ensemble lexical $L_{temp} = \{froide, tiède, chaude\}$. Supposons qu'une personne dispose de deux capteurs symboliques flous [Benoit 93] utilisant cette symbolisation, l'un placé dans le lac d'Annecy et l'autre dans le lac du Bourget, lui permettant de connaître les descriptions des températures de ces deux lacs. Cette personne est à la recherche d'une eau qui soit mi-tiède, mi-chaude pour se baigner, c'est-à-dire d'une eau dont la description est : $0.5/tiède + 0.5/chaude$. Les descriptions des températures du lac d'Annecy et du lac du Bourget sont respectivement : $\{tiède\}$ et $0.5/froide + 0.5/tiède$. La figure 57 illustre cette situation.

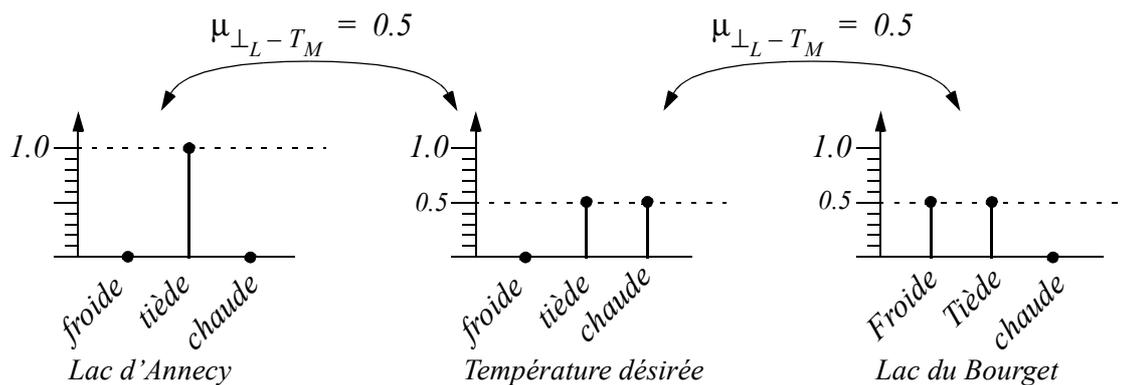


Fig. 57 Exemple où le comportement de $(\perp_L - T_M)$ est contre-intuitif : elle indique que les deux lacs correspondent tous les deux aussi bien à la température désirée alors que le lac du bourget sera manifestement plus froid.

Si elle se fie à son intuition, cette personne ira probablement se baigner à Annecy puisque le lac du Bourget comporte une composante *froide* qui est très éloignée de la composante *chaude* partiellement recherchée. Mais si elle se fie à l'opérateur $(\perp_L - T_M)$, ce dernier ne lui indique aucune préférence pour l'une des deux descriptions. Il n'a en effet aucune information concernant le rapport entre le terme *froide* et le terme *chaude* ou *tiède*. Il ne fait qu'une comparaison point par point des fonctions d'appartenance.

L'opérateur $(\perp_L - T_M)$, comme tout opérateur de similarité, ne considère que la relation d'égalité sur l'ensemble des symboles et ne peut pas intégrer l'existence d'une quelconque métrique sur celui-ci. Dans la section 5 de ce chapitre, nous introduisons un nouvel opérateur de distance permettant de comparer des descriptions même lorsque leurs supports sont disjoints en prenant en compte une métrique définie sur les symboles. Mais avant de définir cette distance, nous présentons brièvement le rapport qui existe entre les autres relations $(\perp - T^*)$ et les ϕ -symbolisations, puis nous mentionnons deux autres types de relations sur $F(S)$ qui pourraient permettre de comparer des descriptions de *Id*-symbolisations et expliquons pourquoi nous ne les utilisons pas.

4.4. Autres relations $(\perp - T^*)$

4.4.1. Relations $(\perp_L - T_P)$

Les relations $(\perp_L - T_P)$, où \perp_L est la t-conorme de Lukasiewicz et T_P est la t-norme produit, ont un comportement similaire aux relations $(\perp_L - T_M)$, à savoir que si S est un ensemble fini de symboles, la relation $(\perp_L - T_P)$ est une $T_{0.5}^Y$ -égalité sur une famille de sous-

ensembles flous de S , où $T_{0.5}^Y$ est la t-norme de Yager de paramètre 0.5 [Foulloy 04.a]. La famille de sous-ensembles flous de symboles sur laquelle $(\perp_L - T_P)$ est une $T_{0.5}^Y$ -égalité est la suivante, notée $F_{Id^2}(S)$:

$$\forall A \in F(S), \quad A \in F_{Id^2}(S) \Leftrightarrow \sum_{s \in S} (\mu_A(s))^2 = 1. \quad (70)$$

Les relations $(\perp_L - T_P)$ s'associent alors aux Id^2 -symbolisations, avec $Id^2 : u \mapsto u^2$, pour former des relations d'égalité floues sur les descriptions.

4.4.2. Relations $(\perp_L - T^*)$

Il a été montré que les relations $(\perp_L - T^*)$, où T^* est une t-norme possédant un générateur additif strictement convexe sont également des T -égalités sous certaines conditions. Plus précisément, si on considère un ensemble fini de symboles S et que l'on définit la famille $F_{\phi_{T^*}}(S)$, de sous-ensembles flous de S , comme suit :

$$\forall A \in F(S), \quad A \in F_{\phi_{T^*}}(S) \Leftrightarrow \sum_{s \in S} T^*(\mu_A(s), \mu_A(s)) = 1, \quad (71)$$

alors $(\perp_L - T^*)$ est une T_D -égalité sur $F_{\phi_{T^*}}(S)$, T_D étant la t-norme drastique (la plus petite des t-normes) [Foulloy 04.a]. De telles relations seraient donc à associer à des ϕ_{T^*} -symbolisations, avec $\phi_{T^*} : u \mapsto T^*(u, u)$. Des résultats expérimentaux permettent de conjecturer que ces relations seraient également des T' -égalités sur $F_{\phi_{T^*}}(S)$, T' étant une t-norme supérieure à T_D et dépendant de T^* . Mais ces résultats ne sont pour l'instant pas démontrés pour les t-normes autres que T_P

4.4.3. Relations $(\perp_L^\phi - T_M)$

Si ϕ est une bijection croissante concave de $[0, 1]$ dans lui-même, nous montrons que la relation $(\perp_L^\phi - T_M)$, où \perp_L^ϕ est la t-conorme isomorphe à la t-conorme de Lukasiewicz (Equation (26)) et T_M est la t-norme *min*, est une T_L -égalité sur une famille $F_\phi(S)$ de sous-ensembles flous d'un ensemble symbolique S :

Proposition 16 : Soit une bijection ϕ croissante concave de $[0, 1]$ dans lui-même. Soit S un ensemble fini de symboles et $F_\phi(S)$ la famille de sous-ensembles flous de S définie comme suit :

$$\forall A \in F(S), \quad A \in F_\phi(S) \Leftrightarrow \sum_{s \in S} \phi(\mu_A(s)) = I. \quad (72)$$

Alors la relation $(\perp_L^\phi - T_M)$ est une T_L -égalité sur $F_\phi(S)$.

Preuve : Une preuve de cette proposition est donnée en annexe. ■

Ainsi, les relations $(\perp_L^\phi - T_M)$ sont à associer aux ϕ -symbolisations lorsque ϕ est concave.

4.5. Autres T -équivalences sur $F(S)$

D'autres T -équivalences sur des sous-ensembles flous de symboles pourraient éventuellement être utilisées comme relations d'équivalence floues entre les descriptions d'une ϕ -symbolisation. Deux relations, principalement, sont le plus souvent citées dans la littérature. Il s'agit des relations Eg_T [De Baets 96] et \equiv_T [Gottwald 93]. La première de ces relations est définie comme suit :

$$\forall (A, B) \in F(S) \times F(S), \quad Eg_T(A, B) = \inf_{s \in S} E_T(\mu_A(s), \mu_B(s)), \quad (73)$$

où E_T est la relation d'équivalence naturelle associée à la t -norme T [Boixader 00]. Dans le cas d'une Id -symbolisation $\langle E, S, R \rangle$, les descriptions appartiennent à la famille $F_{Id}(S)$ des sous-ensembles flous de symboles, sur laquelle Eg_T s'écrit :

$$\forall (A, B) \in F_{Id}(S) \times F_{Id}(S), \quad Eg_T(A, B) = I - \sup_{s \in S} (|\mu_A(s) - \mu_B(s)|). \quad (74)$$

La relation qu'elle induit sur l'univers de discours est alors :

$$\begin{aligned} \forall (e_1, e_2) \in E^2, \quad \mu_{Eg_T \circ (D \times D)}(e_1, e_2) &= I - \sup_{s \in S} (|\mu_{D(e_1)}(s) - \mu_{D(e_2)}(s)|), \\ &= I - \sup_{s \in S} (|\mu_{M(s)}(e_1) - \mu_{M(s)}(e_2)|). \end{aligned} \quad (75)$$

L'utilisation de cette relation pour comparer les éléments de l'univers de discours d'une Id -symbolisation a été envisagée dans [Benoit 93]. Il a été montré que cette relation donne des

valeurs de similarité trop importantes pour certains couples d'entités dont les descriptions sont manifestement très différentes. Il a alors été recommandé d'étendre cette relation comme suit :

$$\forall (e_1, e_2) \in E^2, \quad \mu_{Eg_{T^{\circ}}(D \times D)^*}(e_1, e_2) = 1 - \sup_{s \in \mathcal{P}(S)} (|\mu_{M^*(s)}(e_1) - \mu_{M^*(s)}(e_2)|), \quad (76)$$

où M^* est la signification étendue aux ensembles de symboles (voir § 3.1.2, Définition 14). On peut alors montrer que cette relation n'est autre que la relation induite sur l'univers de discours par $(\perp_L - T_M)$ que nous utilisons dans ce manuscrit.

$$\forall (e_1, e_2) \in E^2, \quad \mu_{Eg_{T^{\circ}}(D \times D)^*}(e_1, e_2) = \mu_{(\perp_L - T_M) \circ D \times D}(e_1, e_2). \quad (77)$$

La relation \equiv_T , est définie comme suit :

$$\forall (A, B) \in F_{Id}(S) \times F_{Id}(S), \quad \mu_{\equiv_T}(A, B) = T(\mu_{\subseteq_T}(A, B), \mu_{\subseteq_T}(B, A)), \quad (78)$$

où \subseteq_T est la relation d'inclusion floue suivante :

$$\forall (A, B) \in F_{Id}(S) \times F_{Id}(S), \quad \mu_{\subseteq_T}(A, B) = \inf_{s \in S} I_T(\mu_A(s), \mu_B(s)),$$

I_T étant l'implication résiduelle associée à la t-norme T [Boixader 00].

On peut montrer que la relation qu'elle induit sur l'univers de discours d'une *Id*-symbolisation ne vérifie pas la propriété (49) et ne respecte donc pas la sémantique des fonctions d'appartenance des significations.

5. Extension d'une métrique définie sur les symboles

Nous avons vu que la relation $(\perp_L - T_M)$, permet de comparer les entités d'une Id -symbolisation ou de deux Id -symbolisations différentes utilisant le même ensemble de symboles. Mais cette relation ne donne une information graduelle que si les supports des descriptions des deux entités comparées ont une intersection non vide. Dans le cas contraire, la relation $(\perp_L - T_M)$ donne toujours 0 . Nous avons également vu qu'elle peut donner des résultats contre-intuitifs car elle ne permet pas de prendre en compte l'existence d'une métrique sur l'ensemble des symboles.

Nous nous intéressons ici au cas où une distance est définie sur l'ensemble des symboles S d'une Id -symbolisation $\langle E, S, R \rangle$. Nous allons voir comment cette distance peut être étendue aux descriptions de cette Id -symbolisation.

5.1. Distances induites sur $F_{Id}(S)$: principe

Soit $\langle E, S, R \rangle$ une Id -symbolisation, où S est un ensemble fini sur lequel est définie une distance d_S . On s'intéresse à la façon dont on peut étendre cette distance sur l'ensemble des descriptions floues, ensemble que nous noterons $D(E)$. Etant donné qu'on ne considère que des Id -symbolisations, on a $D(E) \subset F_{Id}(S)$, on cherche donc une distance $d_{F_{Id}(S)}$ sur $F_{Id}(S)$. Rappelons que $F_{Id}(S)$ est défini comme suit :

$$\forall A \in F(S), \quad A \in F_{Id}(S) \Leftrightarrow \sum_{s \in S} \mu_A(s) = 1. \quad (79)$$

L'ensemble S étant fini, l'ensemble des valeurs prises par d_S est lui-même fini. On peut alors considérer que ce sont des valeurs entières sans perdre la généralité du problème. On impose les quatre contraintes suivantes à $d_{F_{Id}(S)}$:

(i) - *Coincidence sur les singletons*

Cette contrainte est le principe même de la généralisation. Elle signifie que lorsque deux descriptions sont des singletons, leur distance doit être égale à la distance d_S entre les symboles correspondants. Pour tout couple de symboles $(s_1, s_2) \in S^2$, on doit avoir :

$$d_{F_{Id}(S)}(\{s_1\}, \{s_2\}) = d_S(s_1, s_2). \quad (80)$$

(ii) - Continuité

Lorsque l'on passe continûment d'un sous-ensemble flou de symboles F à un autre G , appartenant tous deux à $F_{Id}(S)$ la distance $d_{F_{Id}(S)}$ doit passer continûment de $d_{F_{Id}(S)}(F, F) = 0$ à $d_{F_{Id}(S)}(F, G)$. Ce qui nécessite que les valeurs de $d_{F_{Id}(S)}$ soient réelles. Cette contrainte assure de ne pas perdre tout l'avantage de la gradualité des sous-ensembles flous par une distance qui serait discontinue.

(iii) - Précision

La distance entre deux éléments de $F_{Id}(S)$ doit être un nombre précis et non un sous-ensemble flou de \mathbb{R}^+ . Cette contrainte est, d'une part, pratique car les algorithmes utilisés pour la classification de gestes nécessitent de manipuler des valeurs nettes. Elle est, d'autre part, théorique car les Id -symbolisations fournissent des descriptions précises d'objets ou d'évènements. Ces descriptions sont des sous-ensembles flous car ces objets ou évènements ne correspondent que partiellement aux différents symboles. Les fonctions d'appartenance ont une sémantique de similarité et non d'incertitude. Il est donc normal que la distance entre deux descriptions soit elle aussi précise.

(iv) - Cohérence

La contrainte (i) assure un bon comportement de la distance $d_{F_{Id}(S)}$ pour les singletons. Mais elle ne prévient en rien des comportements incohérents pour les autres éléments de $F_{Id}(S)$. Quand on a à faire à une distance entre sous-ensembles nets d'un espace métrique, celle-ci vérifie généralement la propriété suivante (voir Fig. 58) :

Soit A, B, C et D quatre sous-ensembles nets d'un espace métrique muni de la distance d . Si D est une distance entre sous-ensembles nets généralisant d , elle vérifie :

$$\sup_{(u, v) \in A \times B} d(u, v) \leq \inf_{(x, y) \in C \times D} d(x, y) \Rightarrow D(A, B) \leq D(C, D). \quad (81)$$

C'est le cas de la distance de Hausdorff ainsi que des principales distances définies entre sous-ensembles nets (voir [Eiter 97] pour une présentation de telles distances).

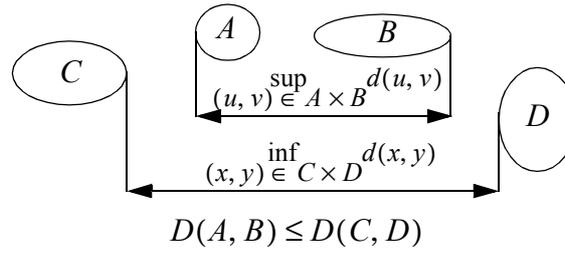


Fig. 58 Propriété généralement vérifiée par une distance D entre sous-ensembles nets d'un espace métrique.

Il parait raisonnable d'imposer à une distance entre sous-ensembles flous d'avoir un comportement similaire. Pour cela, on lui impose de vérifier la propriété suivante :

Soit F, G, H et I quatre éléments de $F_{Id}(S)$. Si $d_{F_{Id}(S)}$ est une distance sur $F_{Id}(S)$ généralisant la distance d_S sur S , elle doit vérifier :

$$\sup_{(s_1, s_2) \in \text{supp}(F) \times \text{supp}(G)} d_S(s_1, s_2) \leq \inf_{(s_3, s_4) \in \text{supp}(H) \times \text{supp}(I)} d_S(s_3, s_4) \Rightarrow d_{F_{Id}(S)}(F, G) \leq d_{F_{Id}(S)}(H, I). \quad (82)$$

où $\text{supp}(F)$ est le support de F . Le choix du support permet d'avoir la condition la moins restrictive possible. Pour éclaircir la signification de cette contrainte nous prenons l'exemple suivant. Soit l'ensemble de symboles $S = \{s_1, s_2, s_3, s_4, s_5\}$, équipé de la métrique d_S donnée à la figure suivante.

d_S	s_1	s_2	s_3	s_4	s_5
s_1	0	1	2	3	4
s_2	1	0	1	2	3
s_3	2	1	0	1	2
s_4	3	2	1	0	1
s_5	4	3	2	1	0

Fig. 59 Distance d_S sur l'ensemble symbolique S .

Considérons alors les quatre éléments de $F_{Id}(S)$ de la figure 60.

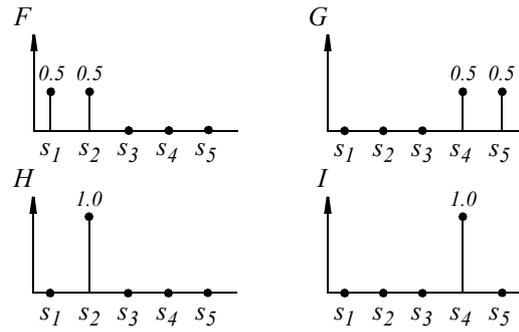


Fig. 60 . Quatre éléments de $F_{Id}(S)$.

Dans un tel exemple, la condition (iv) impose qu'on ait :

$$d_{F_{Id}(S)}(F, G) \geq d_{F_{Id}(S)}(H, I). \quad (83)$$

Nous cherchons finalement une distance sur $F_{Id}(S)$ respectant ces quatre contraintes. Dans un premier temps, nous examinons l'application des distances entre sous-ensembles flous utilisées dans la littérature au cas des descriptions flous.

5.2. Application des distances définies dans la littérature à $F_{Id}(S)$

Le problème de l'extension d'une distance d_S , définie sur un espace fini S , à une distance $d_{F(S)}$, définie sur les sous-ensembles flous de cet espace, a fait l'objet de nombreuses études. Quatre types de généralisation sont proposés dans la littérature [Bloch 99] :

5.2.1. Approche géométrique

Cette première solution consiste à considérer les sous-ensembles flous d'un espace de dimension n comme des sous-ensembles nets d'un espace de dimension $n+1$. Ceci impose que les distances entre les points de l'espace de dimension n et les degrés d'appartenance aient la même sémantique. C'est une condition très forte et il n'y a aucune raison pour qu'elle soit valide pour $F_{Id}(S)$. Ces généralisations ne seront donc pas utilisées ici.

5.2.2. Fuzzification

Une autre approche est de fuzzifier une distance D_S entre sous-ensembles nets, elle-même définie à partir de d_S . Les trois fuzzifications de la distance de Hausdorff étudiées dans [Zwick 87] sont envisagées :

$$H_{F_{Id}(S)}^l(F, G) = \int_0^l H_S(F_\alpha, G_\alpha) d\alpha, \quad (84)$$

$$H_{F_{Id}(S)}^\infty(F, G) = \sup_{\alpha \geq 0} H_S(F_\alpha, G_\alpha), \quad (85)$$

$$H_{F_{Id}(S)}^*(F, G) = H_S(F_{1.0}, G_{1.0}), \quad (86)$$

où F_α et G_α sont les α -coupes de F et G et H_S est la distance de Hausdorff¹. Puisque nous avons postulé que d_S prend des valeurs entières, il en est de même de H_S ainsi que de $H_{F_{Id}(S)}^\infty$ et $H_{F_{Id}(S)}^*$. Seule $H_{F_{Id}(S)}^l$ satisfait la contrainte de continuité et peut être utilisée ici.

La plupart des éléments de $F_{Id}(S)$ n'étant pas normés, il existe généralement un degré α_{12} au delà duquel F_α ou G_α est vide et $H_S(F_\alpha, G_\alpha)$ n'est plus définie. On doit donc modifier la définition de $H_{F_{Id}(S)}^l$ de la façon suivante :

$$H_{F_{Id}(S)}^l(F, G) = \int_0^{\alpha_{12}} H_S(F_\alpha, G_\alpha) d\alpha. \quad (87)$$

Cette distance respecte les deux premières contraintes, mais pas celle de cohérence. En effet, si on reprend l'exemple de la section précédente (Fig. 60), on a :

$$\begin{aligned} H_{F_{Id}(S)}^l(F, G) &= 1.5, \\ H_{F_{Id}(S)}^l(H, I) &= 2. \end{aligned} \quad (88)$$

Cette distance ne vérifie donc pas l'inégalité (83) et ne peut pas être appliquée.

D'autres fuzzifications utilisent le principe d'extension qui conduit à des distances floues. De telles distances ne respectent pas la contrainte de précision et ne sont pas applicables ici.

1. La distance de Hausdorff entre deux sous-ensembles A et B de S est égale à :

$$H_S(A, B) = \max \left(\max_{s_1 \in A} \min_{s_2 \in B} d_S(s_1, s_2), \max_{s_1 \in B} \min_{s_2 \in A} d_S(s_1, s_2) \right).$$

5.2.3. Pondération

Il est possible de généraliser la distance d_S en la pondérant par les degrés d'appartenance :

$$d_T(F, G) = \sum_{s_1 \in S} \sum_{s_2 \in S} T(\mu_F(s_1), \mu_G(s_2)) \cdot d_S(s_1, s_2), \quad (89)$$

où T est une t-norme qui, compte tenu de la contrainte de continuité, doit être continue. L'opérateur d_S étant une distance, (89) peut s'écrire :

$$d_T(F, G) = \sum_{s_1 \neq s_2} T(\mu_F(s_1), \mu_G(s_2)) \cdot d_S(s_1, s_2). \quad (90)$$

Mais l'opérateur obtenu n'est pas une distance au sens strict du terme car il ne vérifie pas l'axiome de séparation comme cela est montré ci-dessous.

■ Rappelons que cet axiome de séparation s'énonce comme suit :

$$d_T(F, G) = 0 \Leftrightarrow F = G. \quad (91)$$

Considérons un sous-ensemble flou F quelconque. L'axiome de séparation impose qu'on aie $d_T(F, F) = 0$, c'est-à-dire :

$$\begin{aligned} \sum_{s_1 \neq s_2} T(\mu_F(s_1), \mu_F(s_2)) \cdot d_S(s_1, s_2) &= 0 \\ \Leftrightarrow \forall (s_1, s_2) \in S^2 / s_1 \neq s_2, \quad T(\mu_F(s_1), \mu_F(s_2)) \cdot d_S(s_1, s_2) &= 0. \end{aligned} \quad (92)$$

Et comme d_S est une distance, si $s_1 \neq s_2$, on a $d_S(s_1, s_2) \neq 0$. Ainsi, (92) est équivalent à :

$$\forall (s_1, s_2) \in S^2 / s_1 \neq s_2, \quad T(\mu_F(s_1), \mu_F(s_2)) = 0. \quad (93)$$

Ce qui signifie que la t-norme T doit posséder des diviseurs de zéros. Supposons donc que ce soit le cas et que T vérifie (93) (on peut vérifier que c'est le cas de la t-norme de Lukasiewicz). Soit alors les deux sous-ensembles flous représentés à la figure 61 où l'ensemble des symboles est $S_L = \{A, B, C, D\}$. Ces deux sous-ensembles flous sont des éléments de $F_{Id}(S_L)$ et sont clairement différents l'un de l'autre. Voyons ce que vaut la distance d_T pour ce couple de sous-ensembles flous.

Pour tout couple $(s_1, s_2) \in S_L^2$ tel que $s_1 \neq s_2$, on a soit $\mu_F(s_1) = 0$, soit $\mu_G(s_2) = 0$, dans quels cas $T(\mu_F(s_1), \mu_G(s_2)) = 0$, soit enfin $\mu_F(s_1) = \mu_G(s_2) = 0.5$. Dans ce dernier cas, $T(\mu_F(s_1), \mu_G(s_2)) = T(0.5, 0.5)$. Or $T(0.5, 0.5)$ est égal à $T(\mu_F(A), \mu_F(B))$, qui est nul puisque la t-norme T est choisie de manière à vérifier (93). Finalement, $T(\mu_F(s_1), \mu_G(s_2))$ est nul pour tout couple de symboles (s_1, s_2) tels que $s_1 \neq s_2$ et $d_T(F, G) = 0$. Ainsi, on ne pourra pas trouver de t-norme vérifiant (93) et telle $d_T(F, G) \neq 0$. Il n'existe donc pas de t-norme telle que l'axiome de séparation soit vérifié. ■

Ainsi l'opérateur d_T n'est pas un bon candidat pour la généralisation de la distance d_S .

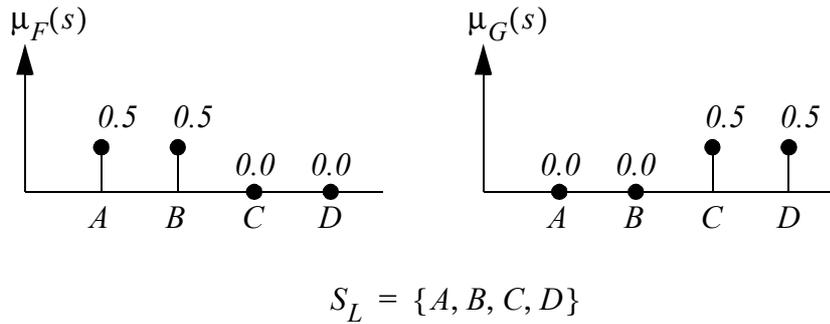


Fig. 61 Un couple de sous-ensembles flous qui mène à une contradiction si l'équation (93) est vérifiée par d_T .

5.2.4. Approche morphologique

La dernière approche de la littérature est l'approche morphologique. La distance de Hausdorff, par exemple, peut s'exprimer en terme d'opérations morphologiques. Ces opérations peuvent être généralisées à des opérations de morphologie floue. La plupart de ces généralisations conduisent cependant à des distances floues et ne sont pas étudiées ici. Une généralisation présentée dans [Dubois 83] donne, quant à elle, un résultat net. Appliqué à $F_{Id}(S)$, elle s'écrit :

$$\tilde{H}(F, G) = \max(\sigma_f(F, G), \sigma_f(G, F)), \quad (94)$$

avec $\sigma_f(F, G) = \inf\{\lambda, F \circ P_{f, \lambda} \supseteq G\}$, où $F \circ P_{f, \lambda}$ est une dilatation de F . Mais le résultat de cette dilatation est un sous-ensemble flou dont la hauteur ne peut dépasser la hauteur de F . Ainsi, si G a une hauteur supérieure à F , il n'existe pas de λ tel que $F \circ P_{f, \lambda} \supseteq G$.

Cette distance n'est donc pas applicable non plus à $F_{Id}(S)$ dont les éléments ne sont généralement pas normés.

5.3. Distance du transport

Aucune des généralisations de distance étudiées dans la littérature ne s'appliquant correctement aux éléments de $F_{Id}(S)$, nous introduisons une nouvelle généralisation spécifique à ces sous-ensembles flous. La distance obtenue est appelée distance du transport et notée d_{tp} car son calcul se ramène à la résolution d'un problème de transport de masses [Rachev 98]. Cette distance est, en cela, l'analogue des distances de Wasserstein utilisées en théorie des probabilités [Rachev 85]. Elle peut également être vue comme une version floue des distances d'édition, ou distances de Levenshtein, utilisées pour comparer des chaînes de caractères [Wagner 74].

Pour définir la distance du transport, nous introduisons une famille de fonctions de $F_{Id}(S)$ dans $F_{Id}(S)$, les transformations $T_{s_i, s_j, x}$.

5.3.1. Transformations $T_{s_i, s_j, x}$

Définition 21 : Soit $S = \{s_1, \dots, s_n\}$ un ensemble fini de symboles. Soit F un sous-ensemble flou de S appartenant à $F_{Id}(S)$. Soit s_i et s_j deux symboles de S et x un réel compris entre 0 et 1. L'image de F par la transformation $T_{s_i, s_j, x}$ est le sous-ensemble flou G tel que :

$$\begin{aligned}\mu_G(s_i) &= \mu_F(s_i) - x, \\ \mu_G(s_j) &= \mu_F(s_j) + x.\end{aligned}\tag{95}$$

Le sous-ensemble flou G appartient clairement lui aussi à $F_{Id}(S)$.

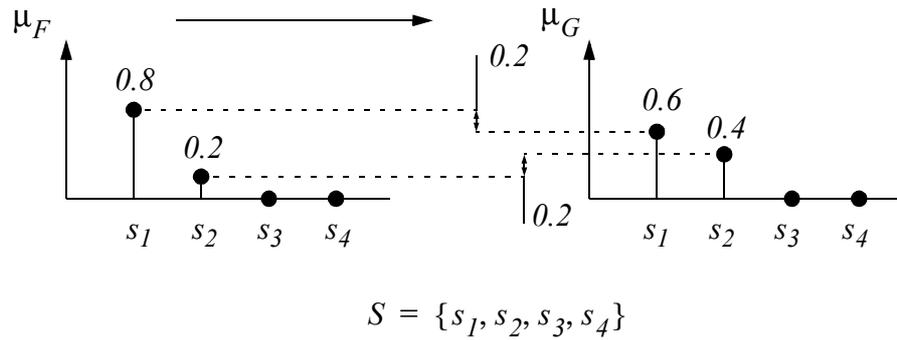


Fig. 62 Un exemple : la transformation $T_{s_1, s_2, 0.2}$.

5.3.2. Distance du transport d_{tp}

Soit F et G , deux éléments quelconques de $F_{Id}(S)$, nous allons montrer qu'il est possible de passer de F à G par une succession de transformations $T_{s_i, s_j, x}$. Notons $S_{F > G}$ et $S_{F = G}$ les ensembles de termes définis comme suit :

$$\begin{aligned} S_{F > G} &= \{s \in S, \mu_F(s) > \mu_G(s)\}, \\ S_{F = G} &= \{s \in S, \mu_F(s) = \mu_G(s)\}. \end{aligned} \quad (96)$$

Par exemple, en utilisant les éléments F et G de $F_{Id}(S)$ représentés à la figure 63, on obtient :

$$\begin{aligned} S_{F > G} &= \{s_1, s_2\}, \\ S_{G > F} &= \{s_4, s_5\}, \\ S_{F = G} &= \{s_3\}. \end{aligned} \quad (97)$$

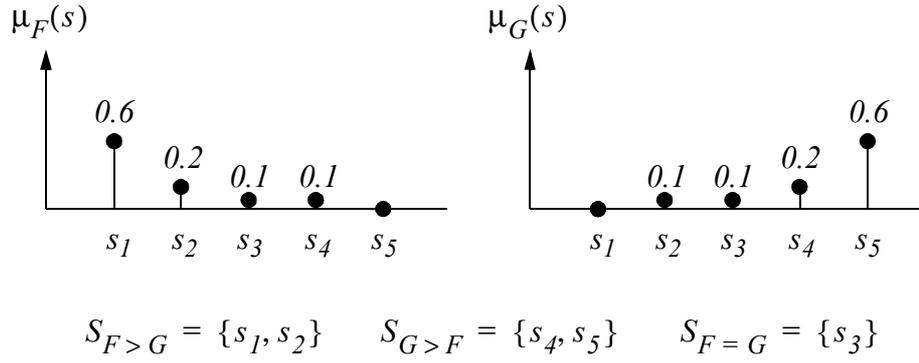


Fig. 63 . Les ensembles $S_{F>G}$, $S_{G>F}$ et $S_{F=G}$.

On a alors la proposition suivante :

Proposition 17 : Soit S un ensemble fini. Soit F et G deux éléments de $F_{Id}(S)$. Pour tout $s \in S$, on note Δ_s la différence $\mu_F(s) - \mu_G(s)$. On a l'égalité suivante :

$$\sum_{s \in S_{F>G}} \Delta_s = - \sum_{s \in S_{G>F}} \Delta_s. \quad (98)$$

Preuve : $S = S_{F>G} \cup S_{G>F} \cup S_{F=G}$. Or, par définition de $F_{Id}(S)$, $\sum_{s \in S} \mu_F(s) =$

$\sum_{s \in S} \mu_G(s) = 1$. Ainsi, on a :

$$\begin{aligned} \sum_{S_{F>G}} \mu_F(s) + \sum_{S_{G>F}} \mu_F(s) + \sum_{S_{F=G}} \mu_F(s) &= \sum_{S_{F>G}} \mu_G(s) + \sum_{S_{G>F}} \mu_G(s) + \sum_{S_{F=G}} \mu_G(s) \\ \Leftrightarrow \sum_{S_{F>G}} \Delta_s + \sum_{S_{G>F}} \Delta_s + \sum_{S_{F=G}} \Delta_s &= 0. \end{aligned}$$

Or $\Delta_s = 0$ sur $S_{F=G}$. On en déduit donc (98). ■

Le problème du passage de F à G par une succession de transformations $T_{s_p, s_p, x}$ est alors un problème de programmation linéaire appelé problème du transport [de Werra 03]. Il s'agit, dans ce dernier, de transporter un produit d'un ensemble de sources notées so contenant chacune

une quantité q_{so} de ce produit vers des destinations notées des devant chacune en recevoir $-q_{des}$. La quantité totale à fournir devant être égale à celle à recevoir, on a :

$$\sum_{so} q_{so} = -\sum_{des} q_{des}. \quad (99)$$

Une solution admissible de ce problème s'énonce en donnant pour chaque couple (so, des) , la quantité $x_{so \rightarrow des}$ transportée de la source so à la destination des . Un coût unitaire $c_{so \rightarrow des}$, dépendant généralement de la distance entre so et des , est attribué à chacun de ces déplacements et le but est de trouver la solution minimisant le coût total $\sum_{so, des} x_{so \rightarrow des} \cdot c_{so \rightarrow des}$.

Les symboles $s \in S_{F > G}$ correspondent alors à des sources, contenant chacune la quantité Δ_s et les symboles $s \in S_{F < G}$ correspondent à des destinations devant chacune recevoir la quantité $-\Delta_s$. L'équation (99) est l'analogue de (98) et une transformation $T_{s_i, s_j, x}$ correspond au choix $x_{s_i \rightarrow s_j} = x$. Il ne reste plus qu'à définir les coûts $c_{s_i \rightarrow s_j}$ que l'on choisit être $d_S(s_i, s_j)$. Notre problème est donc bien un problème du transport (voir Tableau 1). Il possède toujours une solution optimale qui correspond au coût minimal du problème du transport et qui est ici une distance minimale. Cette solution n'est pas forcément unique en terme de séquences de transformations $T_{s_i, s_j, x}$, mais son coût est, lui, bien unique. Il définit alors un opérateur qui ne dépend que de F et G et est noté $d_{tp}(F, G)$.

<i>Problème du transport</i>	<i>Distance du transport</i>
Sources.	Symboles de $S_{F > G}$.
Destinations.	Symboles de $S_{G > F}$.
La quantité totale de produit aux sources est égale à la quantité totale demandée aux destinations : $\sum_{so} q_{so} = \sum_{des} q_{des}.$	$\sum_{s \in S_{F > G}} \Delta_s = \sum_{s \in S_{G > F}} -\Delta_s$

Tableau 1 : Analogie entre la distance du transport et le problème du même nom.

<i>Problème du transport</i>	<i>Distance du transport</i>
Quantité x de produit transporté de la source s_i à la destination s_j .	Transformation $T_{s_i, s_j, x}$ avec $s_i \in S_{F > G}$ et $s_j \in S_{G > F}$.
Coût unitaire du transport du produit de la source s_i vers la destination s_j : $c_{s_i \rightarrow s_j}$.	Distance entre les symboles s_i et s_j : $d_S(s_i, s_j)$.

Tableau 1 : Analogie entre la distance du transport et le problème du même nom.

Le calcul de $d_{tp}(F, G)$ se fait donc en résolvant le problème du transport associé. Pour cela, on peut utiliser, par exemple, l'algorithme fini du simplexe [de Werra 03], mais il n'existe pas de solution analytique applicable dans tous les cas.

Proposition 18 : *L'opérateur d_{tp} est une distance sur $F_{Id}(S)$.*

Preuve : Il s'agit de montrer que pour tout $F, G, H \in F_{Id}(S)$, on a :

$$d_{tp}(F, G) = 0 \Leftrightarrow F = G, \quad (100)$$

$$d_{tp}(F, G) = d_{tp}(G, F), \quad (101)$$

$$d_{tp}(F, G) + d_{tp}(G, H) \geq d_{tp}(F, H). \quad (102)$$

On a $F = G$ ssi $S_{F > G} = S_{F < G} = \emptyset$ qui est vrai ssi $d_{tp}(F, G) = 0$ d'où (100). La symétrie (101) de d_{tp} découle de la symétrie du problème du transport. Enfin, $d_{tp}(F, H)$ est, par définition, la distance correspondant à une séquence optimale de transformations $T_{s_i, s_j, x}$ permettant passer de F à H . Elle est donc inférieure à la distance correspondant à une séquence passant par G et (102) vient naturellement. ■

Proposition 19 : *La distance du transport respecte les quatre contraintes fixées au paragraphe 5.1.*

Preuve : Si on calcule la distance du transport entre deux singletons $\{s_i\}$ et $\{s_j\}$, le problème du transport associé est de trouver la façon la plus économique de faire passer une quantité unitaire de la source s_i à la destination s_j et la solution de ce problème est

clairement de ne faire aucun détour, de transporter directement cette quantité unitaire de s_i à s_j . Le coût de cette solution est alors $c_{s_i \rightarrow s_j}$ qui correspond à la distance $d_S(s_i, s_j)$ (voir Tableau 1). La distance du transport vérifie donc la contrainte de coïncidence sur les singletons. Elle donne un résultat précis et est clairement continue. Elle respecte enfin la contrainte de cohérence, comme cela est prouvé ci-dessous.

■ Soit F, G, H, I quatre éléments de $F_{Id}(S)$ et $(s_1, s_2, s_3, s_4) \in S^4$ un quadruplet de termes tels que :

$$(s_1, s_2) \in \supp(F) \times \supp(G) \quad d_S(s_1, s_2) \leq (s_3, s_4) \in \supp(H) \times \supp(I) \quad d_S(s_3, s_4). \quad (103)$$

On pose :

$$\begin{aligned} \underline{d}_{HI} &= \inf_{(s_3, s_4) \in \supp(H) \times \supp(I)} d_S(s_3, s_4), \\ \overline{d}_{FG} &= \sup_{(s_1, s_2) \in \supp(F) \times \supp(G)} d_S(s_1, s_2). \end{aligned} \quad (104)$$

(103) s'écrit alors :

$$\overline{d}_{FG} \leq \underline{d}_{HI}. \quad (105)$$

Il faut montrer qu'on a alors $d_{tp}(F, G) \leq d_{tp}(H, I)$.

Si $H \cap I \neq \emptyset$, alors $\underline{d}_{HI} = 0$ et donc $\overline{d}_{FG} = 0$. F et G sont alors un seul et même singleton et $d_{tp}(F, G) = 0$. Dans ce cas, on a bien $d_{tp}(F, G) \leq d_{tp}(H, I)$.

Si maintenant $H \cap I = \emptyset$, la quantité transportée dans le problème du transport associé au calcul de $d_{tp}(H, I)$ est égale à 1 et dans ce cas $d_{tp}(H, I) \geq \underline{d}_{HI}$ car la distance du transport est alors une moyenne pondérée de distances toutes supérieures ou égales à \underline{d}_{HI} . D'autre part, la distance du transport $d_{tp}(F, G)$ est une moyenne pondérée de distances toutes inférieures ou égales à \overline{d}_{FG} , d'où $d_{tp}(F, G) \leq \overline{d}_{FG}$. Alors, avec (105), $d_{tp}(F, G) \leq d_{tp}(H, I)$ et la distance du transport respecte bien la condition de cohérence. ■

La distance du transport peut donc être utilisée pour généraliser à $F_{Id}(S)$ une distance d_S définie sur un ensemble de symboles S . Elle est construite en utilisant la propriété spécifique de $F_{Id}(S)$ qui est que la somme des degrés d'appartenance est égale à 1. L'absence de formule analytique générale permettant de calculer cette distance en toute circonstance est son principal inconvénient.

5.3.3. Illustration

Nous illustrons le comportement de cette nouvelle distance sur l'exemple de la *Id*-symbolisation de la flexion d'un grand doigt $\langle X_{flexionIndex}, L_{flexionDoigt}, R_{flexionIndex} \rangle$. La métrique suivante est définie, arbitrairement, sur $L_{flexionDoigt}$:

$d_{L_{flexionDoigt}}$	<i>plié</i>	<i>griffe</i>	<i>rond</i>	<i>carré</i>	<i>tendu</i>
<i>plié</i>	0	1	2	3	4
<i>griffe</i>	1	0	1	2	3
<i>rond</i>	2	1	0	1	2
<i>carré</i>	3	2	1	0	1
<i>rendu</i>	4	3	2	1	0

Fig. 64 Distance $d_{L_{flexionDoigt}}$ utilisée dans notre exemple.

Les figures 66 et 67 donnent le tracé de $d_{tp}(D(x_{ref}), D(x))$, avec $(x_{ref}, x) \in X_{flexionDoigt}^2$. Rappelons que $X_{flexionDoigt} = X_{IP} \times X_{MCP} = [0, 2.5]^2$. On ne représente que la partie $[0.5, 2.0]^2$.

Sur la figure 66, $x_{ref} = x_{ref_1} = (0.5, 0.5)$ ($mcp = ip = 0.5$) et x parcourt l'ensemble du plan $[0.5, 2.0]^2$. Sur la figure 67, $x_{ref} = x_{ref_2} = (1.2, 1.0)$ ($mcp = 1.2$ et $ip = 1.0$), x parcourt également l'ensemble du plan $[0.5, 2.0]^2$.

Les descriptions $D_{flexionDoigt}(x_{ref_1})$ et $D_{flexionDoigt}(x_{ref_2})$ sont données à la figure 65.

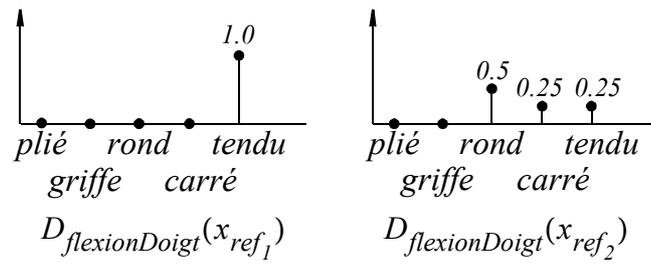


Fig. 65 Descriptions $D_{flexionDoigt}(x_{ref_1})$ et $D_{flexionDoigt}(x_{ref_2})$.

Sur les deux tracés, on observe cinq plateaux correspondant aux cinq termes de $L_{flexionDoigt}$. Les hauteurs de ces plateaux dépendent directement du choix des distances entre les termes. On remarque aussi que sur la figure 67, les plateaux correspondant aux termes *carré* et *rond* sont à la même hauteur, ce qui signifie que :

$$d(D_{flexionDoigt}(x_{ref_2}), \{carré\}) = d(D_{flexionDoigt}(x_{ref_2}), \{rond\}) . \quad (106)$$

Pourtant $\mu_{D_{flexionDoigt}(x_{ref_2})}(Rond) \geq \mu_{D_{flexionDoigt}(x_{ref_2})}(carré)$. Ce résultat provient du fait que $\mu_{D_{flexionDoigt}(x_{ref_2})}(tendu) = 0.25$. La distance du transport prend en compte cette valeur en même temps que la distance du terme *tendu* aux termes *rond* et *carré*.

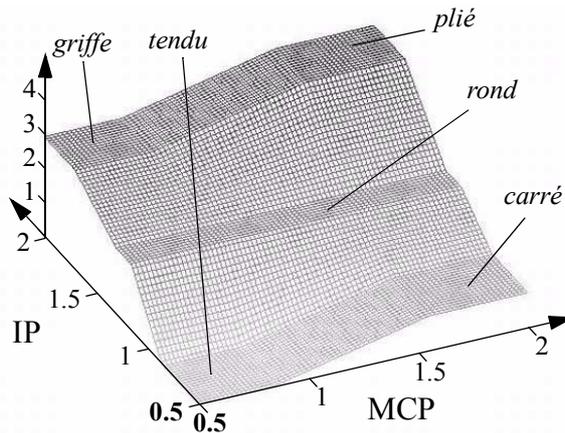


Fig. 66 $d_{tp}(D_{flexionDoigt}(x_{ref_1}), D_{flexionDoigt}(x))$, $x_{ref_1} = (0.5, 0.5)$ et $x \in [0.5, 2.0]^2$.

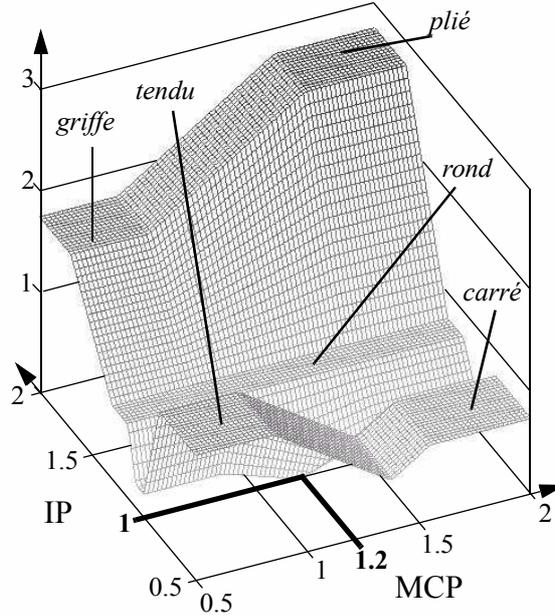


Fig. 67 $d_{tp}(D_{flexionDoigt}(x_{ref_2}), D_{flexionDoigt}(x))$, $x_{ref_2} = (1.2, 1.0)$ et $x \in [0.5, 2.0]^2$.

5.3.4. Rapport avec la distance $d_{(\perp_L - T_M)}$

La distance du transport a un lien direct avec la distance $d_{(\perp_L - T_M)}$ introduite dans la section 4. En effet, s'il n'existe aucune relation sur les symboles, il est toujours possible de choisir comme distance symbolique la distance discrète, notée δ_I , définie comme suit :

$$\forall (s_1, s_2) \in S^2, \quad \delta_I(s_1, s_2) = \begin{cases} 1 & \text{si } s_1 \neq s_2, \\ 0 & \text{si } s_1 = s_2. \end{cases} \quad (107)$$

Dans ce cas, la distance du transport est égale à la distance $d_{(\perp_L - T_M)}$

Proposition 20 : Lorsque la distance définie sur un ensemble de symboles S est la distance discrète δ_I définie par (107), la distance du transport sur $F_{Id}(S)$ est égale à la distance $d_{(\perp_L - T_M)}$.

Preuve : Soit F et G , deux éléments de $F_{Id}(S)$. La distance $d_{(\perp_L - T_M)}$ entre F et G est égale à :

$$d_{(\perp_L - T_M)}(F, G) = 1 - \sum_{s \in S} \min(\mu_F(s), \mu_G(s)). \quad (108)$$

En définissant les ensembles $S_{F > G}$ et $S_{F = G}$ de la même façon qu'à la section 5.3.2 (Equation (96)) et en notant Δ_s la différence $\mu_F(s) - \mu_G(s)$, l'égalité (108) peut s'écrire :

$$\begin{aligned} d_{(\perp_L - T_M)}(F, G) &= \sum_{s \in S} \mu_F(s) - \left(\sum_{s \in S_{G > F}} \mu_F(s) + \sum_{s \in S_{F = G}} \mu_F(s) + \sum_{s \in S_{F > G}} \mu_G(s) \right) \\ &= \sum_{s \in S_{F > G}} (\mu_F(s) - \mu_G(s)) = \sum_{s \in S_{F > G}} \Delta_s. \end{aligned} \quad (109)$$

Or, si l'on se réfère au tableau 1, $\sum_{s \in S_{F > G}} \Delta_s$ est l'analogie de la quantité totale

de produit devant être transportée dans le problème du transport associé au calcul de la distance d_{tp} . Dans ce problème du transport, pour toute source so et toute destination des , le coût d'un transport unitaire de produit de so vers des est égal à 1 puisque la distance définie sur les symboles est la distance discrète. Quel que soit le choix fait pour les différents transports $x_{so \rightarrow des}$, le coût total d'une solution admissible est donc

toujours égal à $\sum_{so, des} x_{so \rightarrow des} \cdot c_{so \rightarrow des} = \sum_{so, des} x_{so \rightarrow des}$ qui est donc la solution de

ce problème du transport et qui est également égale à la quantité totale de produit à transporter. Ainsi, avec (109), la distance $d_{(\perp_L - T_M)}$ est bien égale à la distance du transport associée à la distance symbolique discrète. ■

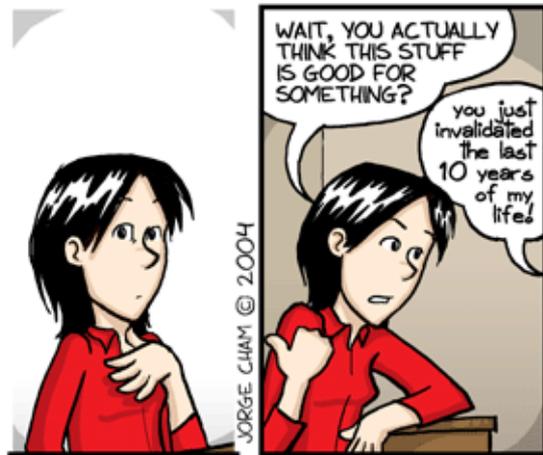
6. Conclusion

Les symbolisations permettent d'établir un lien entre un univers numérique et un univers symbolique plus facilement compréhensible par l'homme. Elles fournissent également un cadre théorique permettant de représenter le lien qu'un capteur classique établit entre l'univers empirique qu'est le monde réel et un univers numérique.

Nous avons ici introduit un nouveau type de symbolisation, les ϕ -symbolisations, rassemblant dans un cadre cohérent plusieurs travaux concernant la conversion numérique-symbolique. Nous avons présenté une méthode de construction de ces symbolisations, présentant les bases de règles comme une façon élégante de définir un regroupement de symboles. Nous avons également défini des notations propres à ce formalisme, permettant entre autre de simplifier l'énonciation des bases de règles tout en restant cohérent avec les définitions. Dans la définition des processus de regroupement de symboles, on a pu remarquer qu'une ϕ -symbolisation n'est pas qu'une collection de n concepts, n étant le nombre de symboles utilisés, mais définit implicitement un ensemble de 2^n concepts différents, ce qui renforce son pouvoir de représentation. Il serait intéressant dans de futurs travaux de pouvoir identifier un symbole s avec le singleton $\{s\}$ étant donné que ces deux objets ont la même signification dans une symbolisation. On pourrait, pour cela, s'appuyer sur la méréologie - *meros* signifiant *partie*, il s'agit littéralement de l'étude des parties - théorie mathématique alternative à la théorie des ensembles dans laquelle on ne différencie justement pas un singleton de l'unique élément qu'il contient.

Des outils permettant de traiter les informations converties sous forme symbolique par une ϕ -symbolisation ont été introduits dans les sections 4 et 5. Dans la section 4, nous avons décrit en détail un certain type de relations, les relations $(\perp - T^*)$. Nous avons présenté plusieurs résultats concernant ces relations et leurs rapports aux ϕ -symbolisations, apportant notamment notre contribution concernant une famille des ces relations, les relations $(\perp_L^\phi - T_M)$, ϕ étant une bijection croissante concave de $[0, 1]$ dans lui-même. Dans la section 5, nous avons introduit un nouvelle distance entre descriptions floues associées à une *Id*-symbolisation, distance basée sur des transports de masses, permettant d'intégrer le fait qu'une métrique peut éventuellement être définie sur les symboles. Lorsque la métrique discrète est utilisée, cette distance du transport se confond avec la relation $(\perp_L - T_M)$, qui est donc un cas particulier, le cas discret, de cette dernière.

Nous allons maintenant voir comment ce formalisme peut être appliqué pour la représentation lexicale de la configuration de la main à partir des données numériques issues du Cyberglove.



www.phdcomics.com

CHAPITRE 4

*Description lexicale de la
configuration de la main*

1. Introduction

Nous choisissons de décrire une posture de la main en donnant la configuration de chaque doigt dans cette posture. L'ensemble des configurations pouvant être prises par un doigt dépend des degrés de liberté de ce doigt. Les grands doigts ont principalement deux degrés de liberté, correspondant à deux types de mouvements : les mouvements de flexion/extension et ceux d'abduction/adduction ou plus simplement d'écartement. Le pouce possède cinq degrés de liberté. La flexion de l'articulation interphalangienne du pouce se traduit par un mouvement de flexion du pouce. Par contre, les mouvements de flexion des articulations MCP_1 et CMC (voir Fig. 70) ont principalement comme effet un changement de l'orientation du pouce par rapport au reste de la main.

Au final, une posture de la main sera donnée par un ensemble de neuf descriptions linguistiques :

- une description pour l'orientation du pouce,
- une description pour la flexion du pouce,
- quatre descriptions correspondant aux flexions des grands doigts,
- trois descriptions correspondant aux écartements des grands doigts.

Il faut donc définir neuf ϕ -symbolisations permettant, à partir des mesures fournies par le gant numérique, d'obtenir ces neuf descriptions lexicales. Ces symbolisations seront des triplets $\langle X, L, R \rangle$, où X sera la plage de variation d'un ensemble de capteurs, L l'ensemble des termes utilisés pour la description et R une relation floue sur $X \times L$. Comme la distance du transport a été définie uniquement pour les Id -symbolisations, nous nous limiterons à ce type de symbolisations. Dans un premier temps, nous choisissons les ensembles lexicaux de ces différentes symbolisations puis nous les construisons en utilisant les techniques introduites dans le chapitre précédent.

2. Description lexicale de la configuration de la main

2.1. Choix des ensembles lexicaux et définition des symbolisations

2.1.1. Écartement des doigts

L'écartement des doigts donne deux configurations d'aspects visuels différents : deux doigts adjacents sont soit écartés, soit serrés. Les trois symbolisations des écartements des grands doigts auront donc comme univers lexical l'ensemble $L_{\text{écart}} = \{\text{écartés}, \text{serrés}\}$. Ces symbolisations seront notées $\langle X_{\text{écart}_{i,j}}, L_{\text{écart}}, R_{\text{écart}_{i,j}} \rangle$. Pour plus de clarté, on utilisera aussi les noms suivant : *écartIndexMajeur*, *écartAnnulaireMajeur*, *écartAuriculaireAnnulaire*.

2.1.2. Flexion des grands doigts

Nous avons distingué cinq configurations visuellement différentes pour la flexion d'un grand doigt (voir Fig. 68). Ces configurations se retrouvent dans les systèmes de notation des signes HamNoSys [Prillwitz 89] et SignWriting [www.signwriting.org]. Elles seront nommées *tendu*, *carré*, *rond*, *griffe* et *plié*.

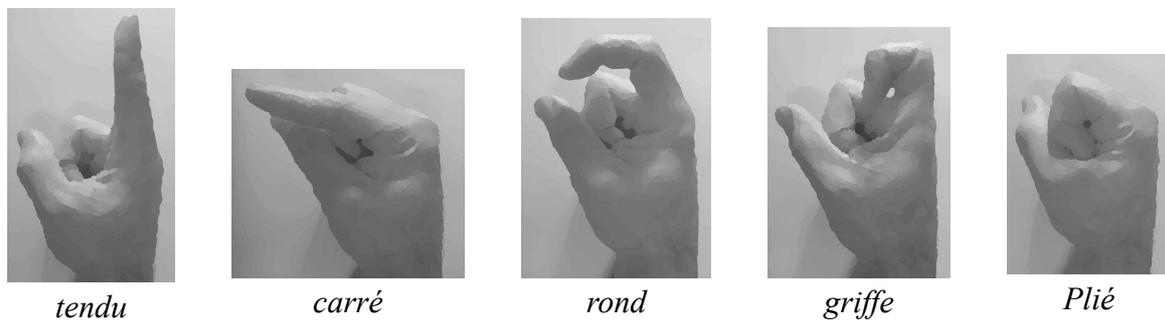


Fig. 68 Les cinq configurations d'un grand doigt.

Les quatre symbolisations seront donc les triplets : $\langle X_{\text{flexionDoigt}_i}, L_{\text{flexionDoigt}}, R_{\text{flexionDoigt}_i} \rangle$, avec $i = 2, 3, 4$ ou 5 . Elles auront le même univers lexical, $L_{\text{flexionDoigt}} = \{\text{tendu}, \text{carré}, \text{rond}, \text{griffe}, \text{plié}\}$. Pour plus de clarté, on pourra utiliser les noms suivants : *flexionIndex*, *flexionMajeur*, *flexionAnnulaire* et *flexionAuriculaire*.

2.1.3. Description du pouce

a) Flexion

La flexion du pouce est décrite très simplement à partir de deux configurations : *plié* et *tendu*. Ainsi, la symbolisation de cette flexion aura un univers lexical beaucoup plus réduit que celles des flexions des grands doigts : $L_{flexionPouce} = \{plié, tendu\}$. Elle sera notée $\langle X_{flexionPouce}, L_{flexionPouce}, R_{flexionPouce} \rangle$.

b) Orientation

Le pouce peut être orienté vers l'extérieur de la main, vers le haut de la main (il est alors collé aux autres doigts), vers l'avant de la main ou vers l'intérieur de la main (voir Fig. 69). Dans ce dernier cas, on distingue le fait qu'il soit collé à la paume de la main et sous les autres doigts si ceux-ci sont repliés ou bien décollé de la paume et au-dessus des autres doigts si ceux-ci sont repliés. On a finalement cinq orientations différentes possibles et la symbolisation de l'orientation du pouce aura l'univers lexical $L_{orientPouce} = \{externe, en_avant, interne_dessus, interne_dessous, de_coté\}$. Cette symbolisation sera notée : $\langle X_{orientPouce}, L_{orientPouce}, R_{orientPouce} \rangle$.

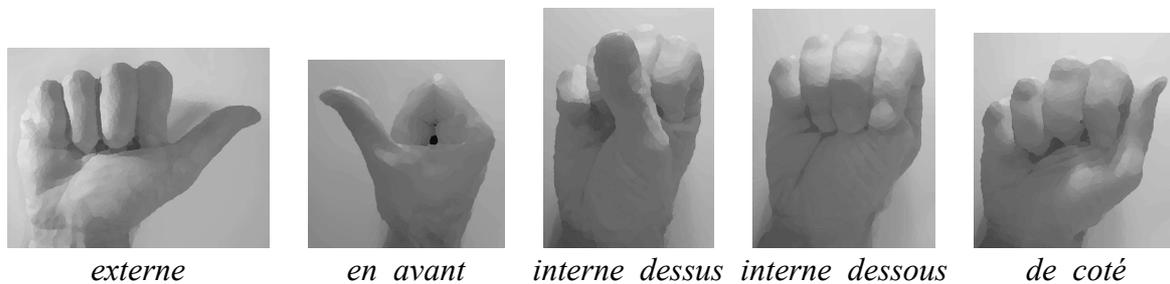


Fig. 69 Les orientations possibles du pouce

2.2. Construction des symbolisations

Nous utilisons le Cyberglove pour mesurer la configuration de la main. Ce gant possède dix-huit capteurs de flexion, représentés à la figure 70. Notre description de la configuration de la main n'intégrant pas une description du poignet, nous n'utilisons pas les deux capteurs mesurant la position de celui-ci. Nous n'utilisons pas non plus le capteur mesurant la fermeture de la paume. Finalement, nous utilisons les cinq capteurs IP_i mesurant les angles des articulations interphalangiennes $IPDs$ et $IPPs$ (voir Fig. 70), les cinq capteurs MCP_i mesurant les angles des articulations métacarpo-phalangiennes, les quatre capteurs d'abduction $Abd_{i,j}$ et

enfin le capteur *CMC* mesurant l'angle de l'articulation carpo-métacarpienne du pouce soit, en tout, quinze capteurs.

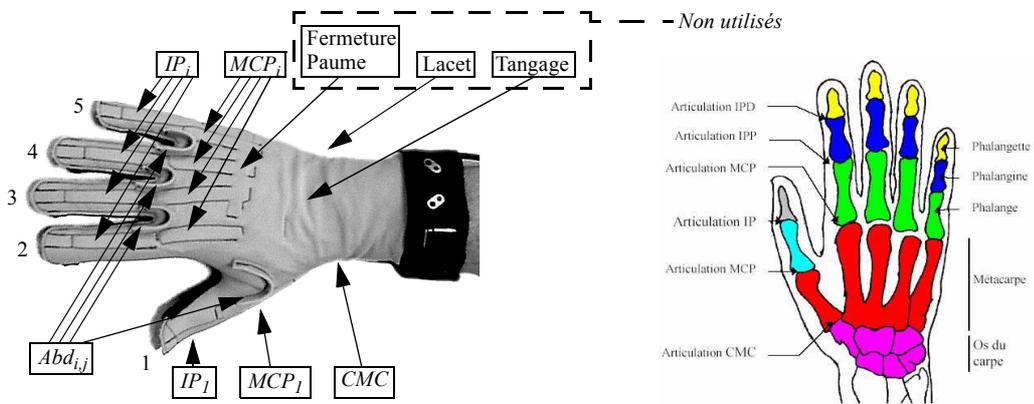


Fig. 70 Les dix-huit capteurs du Cyberglove, la numérotation des doigts et les articulations de la main (tiré de [Ouhaddi 98]).

Nous allons maintenant voir quels capteurs sont associés aux différentes symbolisations et comment sont construites ces symbolisations.

2.2.1. Ecartement des grands doigts

L'écartement de chaque paire (i, j) de grands doigts est mesuré par un capteur d'abduction/adduction, appelé $Abd_{i,j}$, prenant ses valeurs sur $X_{Abd_{i,j}}$. L'univers de discours de la symbolisation $\langle X_{écart_{i,j}}, L_{écart_{i,j}}, R_{écart_{i,j}} \rangle$, avec $(i, j) = (2, 3), (3, 4)$ ou $(4, 5)$, est donc $X_{Abd_{i,j}}$. Les univers de discours de ces trois symbolisations sont alors des intervalles réels et on peut construire directement les significations des termes de l'univers lexical $L_{écart}$ à partir de valeurs prototypes et d'intervalles flous (L-R), comme cela a été décrit au chapitre 3. Ces valeurs prototypes correspondent aux mesures pour lesquelles les doigts sont considérés comme complètement serrés ou complètement écartés. Elles sont déterminées expérimentalement et peuvent être ajustées de manière à rendre les descriptions plus ou moins floues. La figure 71 illustre la construction des significations à partir de ces valeurs.

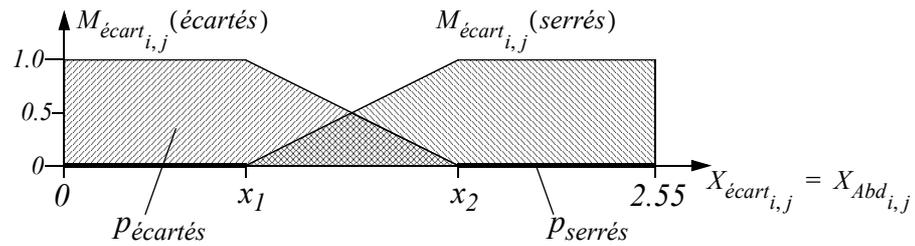


Fig. 71 Significations des termes écartés et serrés d'une symbolisation $\langle X_{écart_{i,j}}, L_{écart}, R_{écart_{i,j}} \rangle$. Les valeurs prototypes des termes sont les intervalles

$$p_{écartés} = [0, x_1] \text{ et } p_{serrés} = [x_2, 2.55].$$

- **Remarque :** La description de l'écartement de deux grands doigts adjacents n'a de sens que si les articulations *MCPs* de ces deux doigts sont toutes deux tendues. Si l'une des deux seulement est tendue, l'autre étant donc pliée, alors les premières phalanges de ces deux doigts ne sont plus dans un même plan et l'écartement latéral n'a plus vraiment de sens (voir Fig. 72.a). Si les deux articulations *MCPs* des doigts sont pliées, une contrainte cinématique de la main impose que ces deux doigts soient serrés (voir Fig. 72.b).



72.a - Dans cette configuration, quelle que soit la mesure du capteur $Abd_{2,3}$, l'index et le majeur sont écartés



72.b - Dans cette configuration (*MCPs* pliés) il est impossible d'écarter les grands doigts. Ils sont donc forcément serrés, quelles que soient les mesures fournies par les capteurs $Abd_{i,j}$.

Fig. 72 Configurations où la description de l'écartement de deux grands doigts est imposée et n'a pas besoin d'être mesurée.

2.2.2. Flexion des grands doigts

La flexion d'un grand doigt i , est mesurée par les deux capteurs de flexion IP_i et MCP_i (voir Fig. 70). L'univers de discours de la symbolisation $\langle X_{flexionDoigt_i}, L_{flexionDoigt_i}, R_{flexionDoigt_i} \rangle$, avec $i = 2, 3, 4$ ou 5 , sera donc $X_{IP_i} \times X_{MCP_i}$. C'est un produit cartésien d'intervalles réels. Pour construire une symbolisation sur cet univers de discours, nous définissons d'abord deux symbolisations, ayant comme univers de discours respectifs les intervalles X_{IP_i} et X_{MCP_i} puis agrégeons ces deux symbolisations par une base de règles.

Tous les grands doigts étant décrits de la même façon, les quatre symbolisations intermédiaires $\langle X_{IP_i}, L_{IP_i}, R_{IP_i} \rangle$, avec $i = 2, 3, 4$ ou 5 , auront le même univers lexical : $L_{IP} = \{tendu, mi_plié, plié\}$. De même, les quatre symbolisations $\langle X_{MCP_i}, L_{MCP_i}, R_{MCP_i} \rangle$ auront l'univers lexical $L_{MCP} = \{tendu, plié\}$. Il y a donc un niveau de description supplémentaire pour les articulations IPs . Ce choix vient du fait que l'angle de flexion des articulations interphalangiennes a un domaine de variation deux fois plus grand (environ 180°) que celui des métacarpo-phalangiennes (environ 90°).

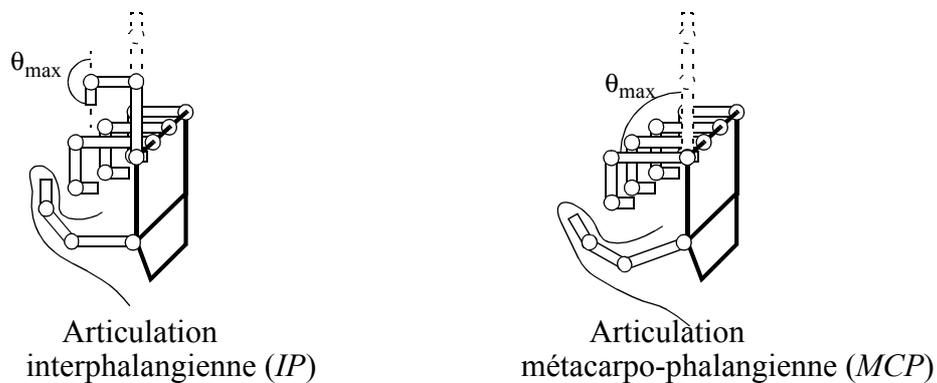


Fig. 73 Domaines de variation des angles des articulations IPs et $MCPs$ des grands doigts.

Les significations des termes de L_{IP} et L_{MCP} dans les huit symbolisations intermédiaires sont construites à partir de valeurs prototypes de ces termes, choisies expérimentalement, et d'intervalles flous (L-R), selon le processus décrit au chapitre 3. Chaque couple de symbolisation $\langle X_{IP_i}, L_{IP_i}, R_{IP_i} \rangle$, $\langle X_{MCP_i}, L_{MCP_i}, R_{MCP_i} \rangle$ est agrégé par la base de règle de la figure 74.

- (1) Si IP est **tendu** et MCP est **tendu** Alors flexionDoigt est **tendu**
- (2) Si IP est **tendu** et MCP est **plié** Alors flexionDoigt est **carré**
- (3) Si IP est **mi_plié** et MCP est **tendu ou plié** Alors flexionDoigt est **rond**
- (4) Si IP est **plié** et MCP est **tendu** Alors flexionDoigt est **griffe**
- (5) Si IP est **plié** et MCP est **plié** Alors flexionDoigt est **plié**

Fig. 74 Base de règles utilisée pour l'agrégation des couples de symbolisations $\langle X_{IP_i}, L_{IP_i}, R_{IP_i} \rangle, \langle X_{MCP_i}, L_{MCP_i}, R_{MCP_i} \rangle$, avec $i = 2, 3, 4$ ou 5 .

Les significations des termes de $L_{flexionDoigt}$ obtenues par cette agrégation sont données, dans le cas de l'index, à la figure suivante. Ces significations sont des sous-ensembles flous du plan $X_{IP_2} \times X_{MCP_2}$. Ce plan est normalement égal à $[0, 2.55]^2$, il est ici tronqué à $[0.5, 2]$ pour les besoins de la représentation.

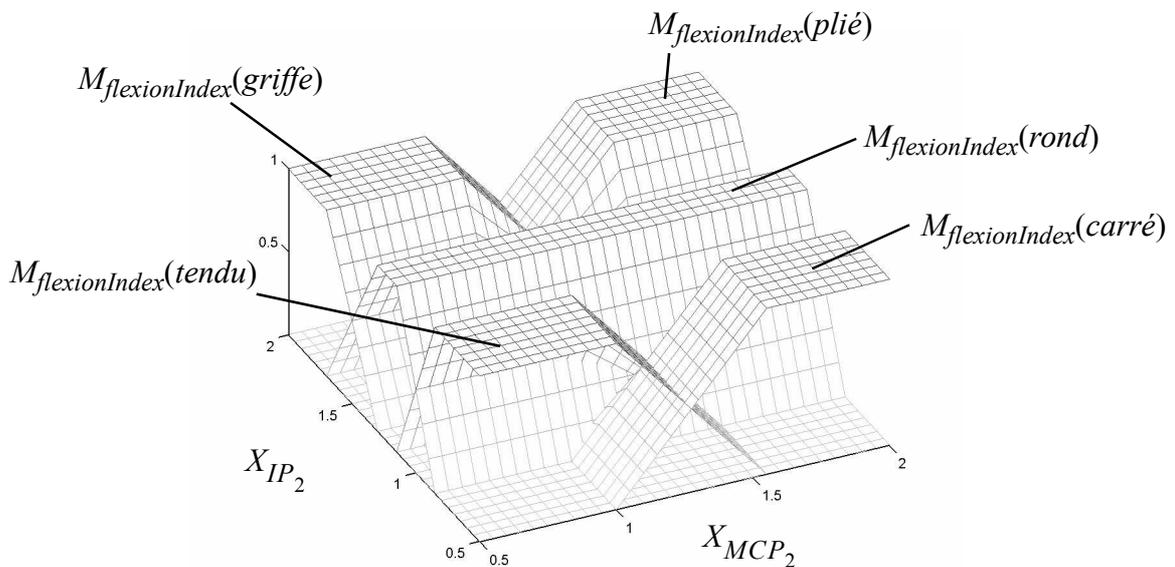


Fig. 75 Significations des termes de $L_{flexionDoigt}$ dans le cas de l'index.

2.2.3. Description du pouce

a) Flexion du pouce

La flexion du pouce est mesurée par le capteur IP_1 , on a donc $X_{flexionPouce} = X_{IP_1}$ et la symbolisation $\langle X_{flexionPouce}, L_{flexionPouce}, R_{flexionPouce} \rangle$ est construite à partir de prototypes des termes *plié* et *tendu* pour cette symbolisation.

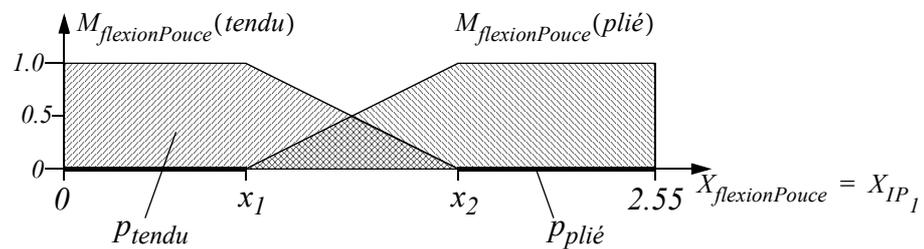


Fig. 76 Significations des termes *plié* et *tendu* de la symbolisation $\langle X_{flexionPouce}, L_{flexionPouce}, R_{flexionPouce} \rangle$. Les valeurs prototypes des termes sont les intervalles $p_{tendu} = [0, x_1]$ et $p_{plié} = [x_2, 2.55]$.

b) Orientation du pouce

L'orientation du pouce est mesurée par trois capteurs : CMC , MCP_1 et $Abd_{1,2}$ (voir Fig. 70). On a donc $X_{orientPouce} = X_{CMC} \times X_{MCP_1} \times X_{Abd_{1,2}}$. La dimension de l'univers de discours est supérieure à 2 et il n'existe pas de règles évidentes permettant d'agréger des symbolisations qui seraient construites sur les trois ensembles X_{CMC} , X_{MCP_1} et $X_{Abd_{1,2}}$. Nous avons alors cherché à réduire la dimension de cet univers de discours en menant une analyse en composantes discriminantes.

Un ensemble de postures contenant divers exemples des cinq orientations du pouce (voir Fig. 69) a été enregistré, formant ainsi cinq classes de postures correspondant aux cinq configurations. L'analyse en composantes discriminantes a montré que deux axes permettaient de séparer correctement ces cinq classes (voir Fig. 78). D'autre part, on peut donner une sémantique particulière à chacun de ces axes : le premier axe, Y_1 , correspond à l'orientation latérale du pouce, c'est-à-dire à son orientation dans le plan de la paume de la main. Le deuxième axe, Y_2 , mesure l'orientation transversale du pouce, c'est-à-dire à son orientation

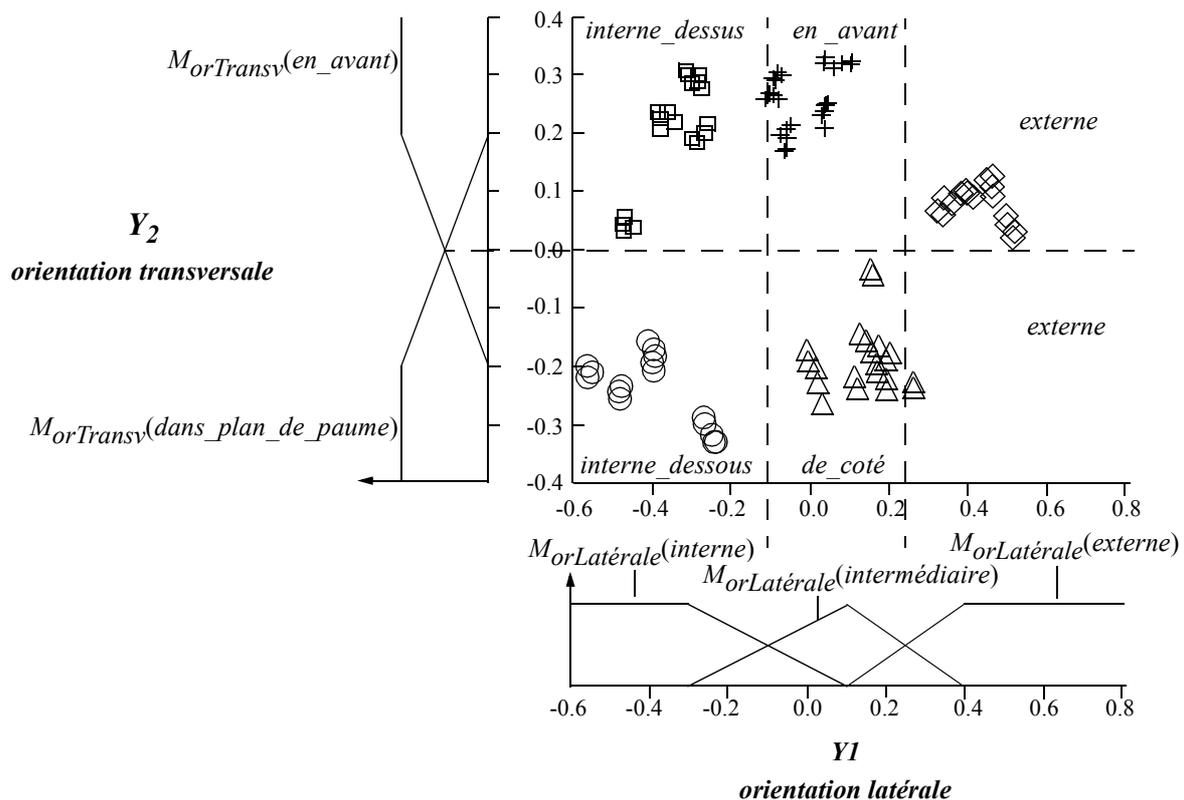


Fig. 78 Ensemble d'apprentissage de l'orientation du pouce projeté sur les deux axes discriminants. Les partitions de ces deux axes sont construites à partir de cet ensemble d'apprentissage.

2.3. Schéma général

Avant de récapituler le principe de construction des différentes symbolisations, nous introduisons l'ensemble des postures P , univers empirique qui sera largement utilisé par la suite.

2.3.1. L'ensemble des postures P

Dans la suite de ce manuscrit, on notera P l'ensemble des postures de la main. A chacune de ces postures correspond une mesure du Cyberglove. Mais nombre de mesures que pourrait fournir le gant numérique correspondent à des postures qui sont irréalisables par la main du fait des contraintes cinématiques de celle-ci. Cependant, le fait qu'une chose soit irréalisable ne signifie pas qu'elle n'existe pas. Un triple salto arrière, départ à l'arrêt, n'est pas réalisable. Mais cette figure, au moins en tant que concept, existe pourtant bien. Nous considérerons donc que toutes les mesures du gant numérique correspondent à des postures, pas forcément réalisables, de P .

Soit $\langle X, L, R \rangle$ l'une des neuf *Id*-symbolisations que nous allons définir. Notons C_X l'ensemble des capteurs dont X est la plage de variation. Ces capteurs C_X permettent de passer

d'une posture $p \in P$ de la main, entité du monde physique, à une mesure numérique $C_X(p) \in X$. Chaque symbolisation $\langle X, L, R \rangle$ induira alors une symbolisation $\langle P, L, R' \rangle$ dont l'univers de discours est l'ensemble des postures de la main.

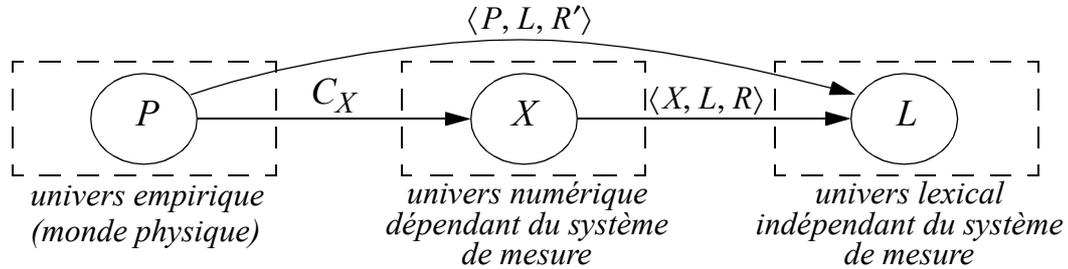


Fig. 79 Relations entre une symbolisation $\langle X, L, R \rangle$ et la symbolisation $\langle P, L, R' \rangle$ qu'elle induit.

La relation R' se déduira de la relation R comme suit :

$$\forall (p, l) \in P \times L, \quad \mu_{R'}(p, l) = \mu_R(C_X(p), l). \quad (110)$$

On peut vérifier que $\langle P, L, R' \rangle$ est bien une *Id*-symbolisation. On a en effet :

$$\forall p \in P, \quad \sum_{l \in L} \mu_{R'}(p, l) = \sum_{l \in L} \mu_R(C_X(p), l) = 1. \quad (111)$$

De plus, puisque toutes les significations des termes de L dans $\langle X, L, R \rangle$ sont normées et que toute mesure du Cyberglove correspond à une posture p , les significations des termes de L dans $\langle P, L, R' \rangle$ sont elles-même normées.

Les symbolisations $\langle P, L, R' \rangle$ sont plus intéressantes que les symbolisations $\langle X, L, R \rangle$ car elles permettent de passer directement de l'univers empirique à l'univers lexical qui sont tous les deux indépendants du système de mesure et on peut imaginer de les construire avec un autre gant numérique que le Cyberglove ou avec un système de vision. Dans les chapitres suivants, nous ne considérerons pratiquement plus que les symbolisations $\langle P, L, R' \rangle$ et ne ferons que peu référence aux symbolisations $\langle X, L, R \rangle$. Bien que nous ayons ici noté différemment la relation R , qui est une relation sur $X \times L$ et la relation R' , qui est une relation sur $P \times L$, le contexte permettra toujours de différencier ces deux relations si bien que nous pourrons les noter toutes les deux avec le même symbole. C'est ce qui est fait dans le schéma général suivant.

2.3.2. Schéma général

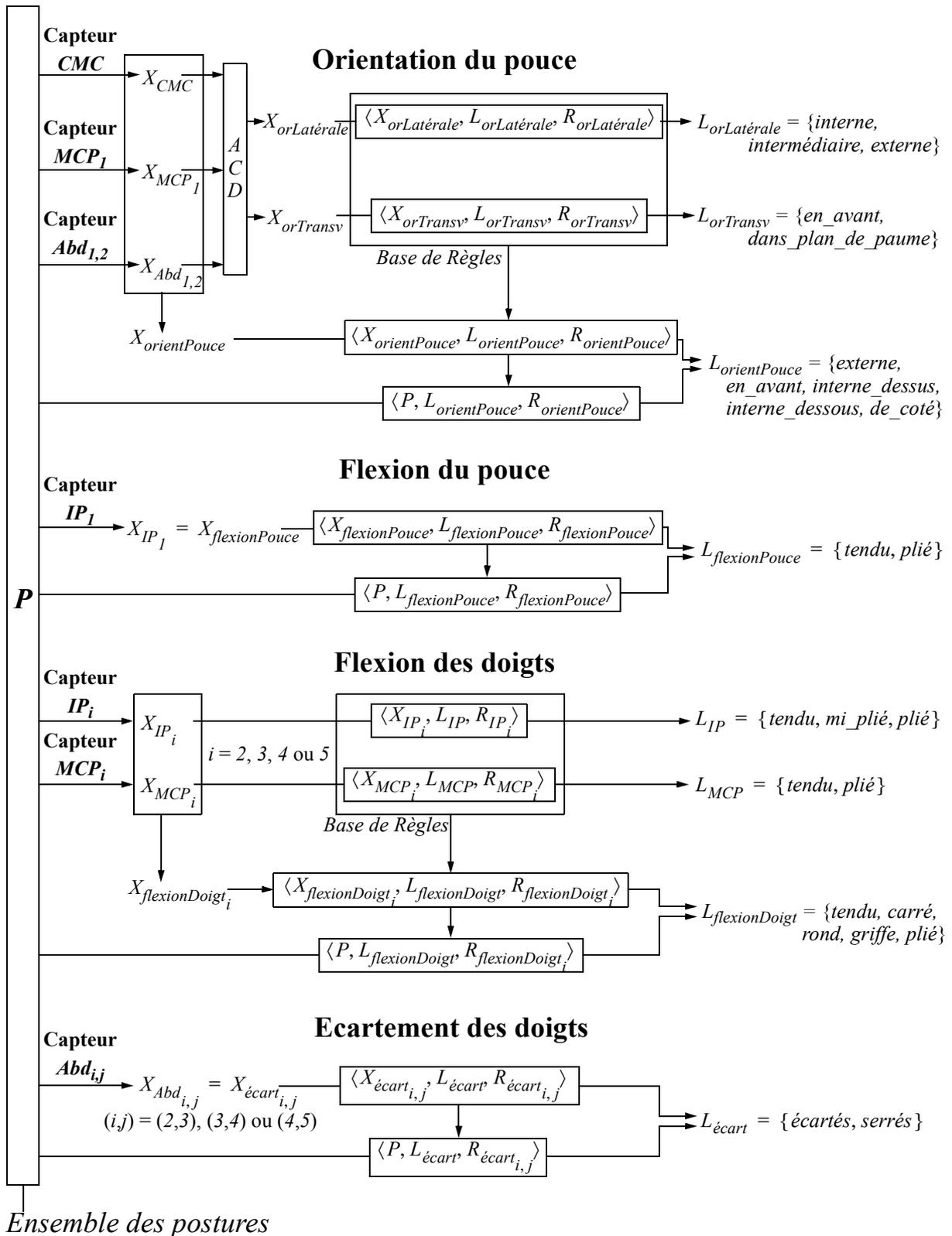


Fig. 80 Construction des neuf symbolisations.

2.3.3. La symbolisation $\langle P, L_P, R_P \rangle$

Nous pouvons faire le produit cartésien des neuf symbolisations que nous venons de construire. Nous obtenons alors une symbolisation globale de la configuration de la main, dont les symboles correspondent à des configurations particulières de la main qu'il nous sera possible de définir. L'ensemble de ces configurations, est noté L_P et est donc défini comme suit :

$$L_P = L_{orientPouce} \times L_{flexionPouce} \times L_{flexionDoigt}^4 \times L_{écart}^3 \quad (112)$$

Cet ensemble contient 50 000 éléments et nous pouvons donc définir autant de configurations particulières de la main, même si beaucoup de ces configurations ne sont pas réalisables. Considérons par exemple, les deux 9-uplets suivants :

$$\begin{aligned} c_1 &= (externe, tendu, plié, plié, tendu, tendu, serrés, serrés, serrés), \\ c_2 &= (interne_dessus, plié, tendu, plié, plié, plié, serrés, serrés, serrés). \end{aligned} \quad (113)$$

Ce sont tous les deux des éléments de L_P . c_1 est une configuration très difficilement réalisable de la main. c_2 est réalisable très facilement et correspond à la configuration généralement utilisée pour montrer un objet ou une direction.

2.4. Exemple

L'ensemble des neuf symbolisations qui viennent d'être construites fournit donc une représentation de la configuration de la main sous la forme de neuf descriptions floues, c'est-à-dire de neuf sous-ensembles flous de termes. La symbolisation $\langle P, L_P, R_P \rangle$ fournit la même information mais sous la forme d'un sous-ensemble flou de 50 000 termes. Elle ne sera pas utilisée pour décrire une posture particulière, mais pour la reconnaissance de signes dans le chapitre suivant.

Etant donné que nous avons construit des *Id*-symbolisations, les descriptions floues que nous obtenons ont la somme de leurs degrés d'appartenance égale à 1. Nous considérons maintenant une posture p_0 de la main. Cette posture de la main correspond à la mesure suivante du Cyberglove, déjà donnée en exemple au début du chapitre 3.

[1.58, 1.17, 1.65, 0.32, 1.22, 0.91, 1.51, 1.76, 1.64, 1.58, 1.73, 1.01, 1.84, 1.42, 1.38]

A quelle configuration de la main correspond cette mesure du Cyberglove ?

Fig. 81 Mesure fournie par le Cyberglove de la posture p_0 .

Comme on l'a déjà remarqué, ces mesures numériques sont difficilement interprétables. Il n'est pas évident de répondre à la question posée à la figure 81 sans s'aider de la photo de la posture p_0 , donnée à la figure 83. Nous allons voir que les descriptions lexicales de la même posture, permettent, quant à elles, d'y répondre simplement. La figure 82 donne les valeurs des neuf descriptions de la posture p_0 .

Les trois grands doigts que sont le majeur, l'annulaire et l'auriculaire sont majoritairement décrits par le terme *plié*. Ils sont donc serrés, du fait de la contrainte cinématique de la main dont nous avons parlé dans la section 2.2.1 et indépendamment des descriptions $D_{\text{écart}_{i,j}}(p_0)$. Le pouce est majoritairement *interne_dessus*, ce qui signifie qu'il vient recouvrir ces trois grands doigts. Il est également *plié*, ce qui est généralement le cas lorsqu'il est *interne_dessus*. Il ne s'agit pas là d'une contrainte cinématique de la main, mais plus d'une tendance naturelle à plier le pouce lorsqu'on l'utilise pour recouvrir les grands doigts. L'index enfin se trouve entre les trois configurations *rond*, *carré* et *tendu*. Il n'est donc pas *plié* ni *griffe* et se distingue ainsi des autres grands doigts. Finalement, la posture de la main correspondant à ces descriptions se rapproche de la configuration de la main utilisée pour montrer un objet ou une direction, c'est-à-dire de la configuration c_2 définie précédemment (Equation (113)), ce que l'on peut vérifier en se reportant à la figure 83.

Mais cette posture n'est réalisée qu'approximativement. L'index, par exemple, n'est pas parfaitement tendu. Alors, la valeur de $\mu_{M_p(c_2)}(p_0)$ est très faible :

$$\mu_{M_p(c_2)}(p_0) = 0.04. \quad (114)$$

Cet exemple permet donc de comprendre que les configurations telles que c_2 , que nous utiliserons par la suite, correspondent à des classes de postures très étroites. Dans les chapitres suivants, nous manipulerons plutôt des classes de configurations et pourrons ainsi représenter efficacement des signes statiques.

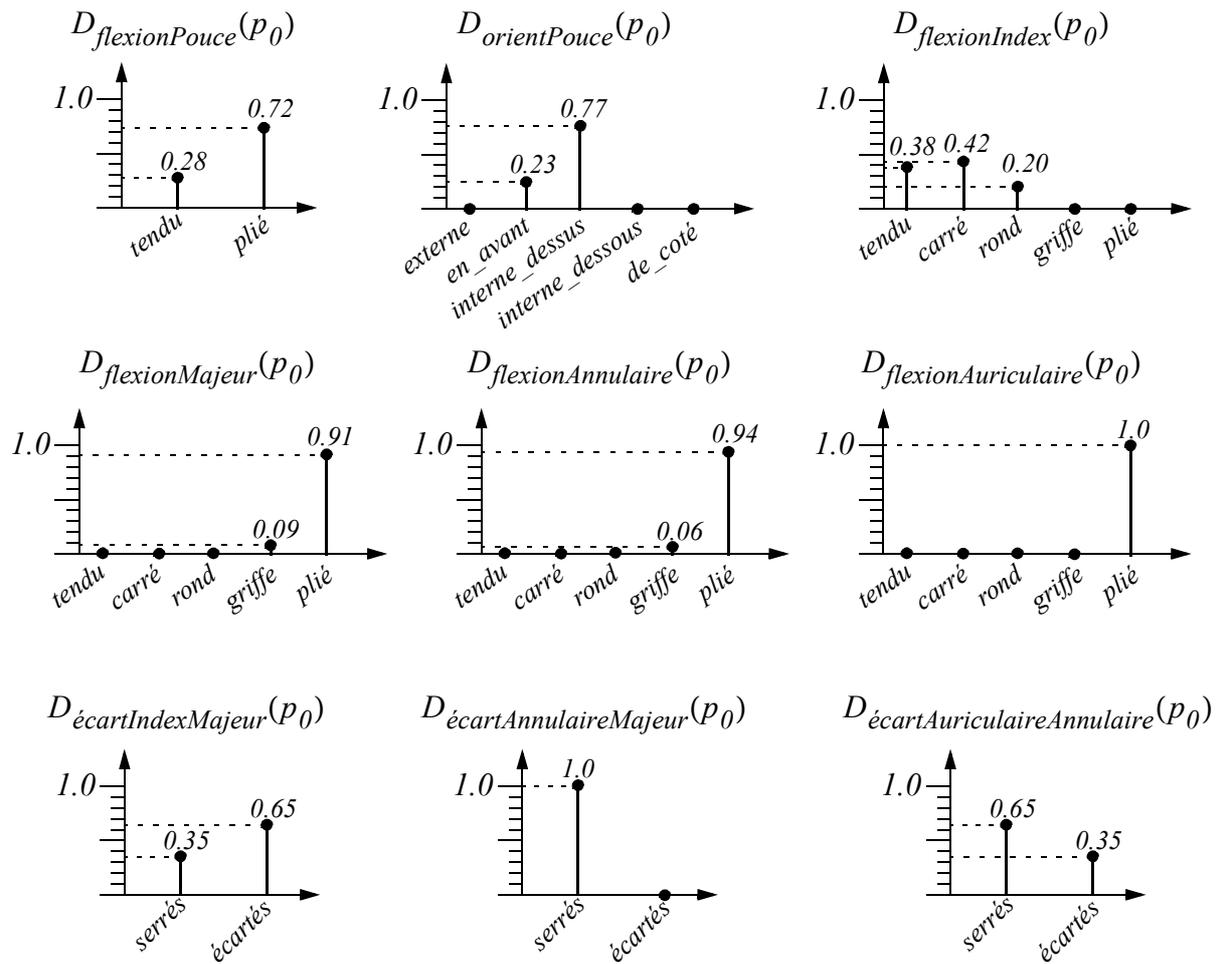


Fig. 82 Descriptions issues des neuf symbolisations, pour la posture p_0 .



Fig. 83 Photo de la posture p_0 correspondant aux mesures de la figure 81 et aux descriptions de la figure 82.

3. Conclusion

Nous avons défini neuf symbolisations permettant de décrire la configuration de la main par des sous-ensembles flous de termes. Ces symbolisations ont été choisies de façon à décrire les différentes configurations de chacun des doigts. Seul l'aspect visuel de ces configurations a été pris en compte. Cette description a donc un caractère très général et peut être utilisée quels que soient les signes étudiés. Par contre, chaque doigt est décrit indépendamment des autres. Il serait intéressant dans de futurs travaux de travailler sur une description des différentes combinaisons possibles des doigts, de pouvoir par exemple décrire précisément le fait que l'extrémité du pouce est en contact avec celle de l'index. Une autre orientation possible serait de travailler sur une description inspirée directement de systèmes existant de description de gestes de la main tels que les systèmes de notation HamNoSys [Prillwitz 89] et SignWriting [www.signwriting.org]. Enfin, il serait utile de travailler à la génération automatique de ces neuf symbolisations, afin que le système puisse s'adapter rapidement à tout nouvel utilisateur.

L'avantage de notre approche est que le nombre de termes finalement utilisés est très réduit. Cinq termes sont utilisés pour décrire les flexions des grands doigts, deux pour celle du pouce, cinq pour son orientation et deux pour l'écartement des grands doigts. Au final, notre système de description de la main, pour être compréhensible par une personne nécessite de ne retenir qu'au plus quatorze termes, mais il permet de décrire plusieurs dizaines de milliers de configurations différentes de la main.

Dans les chapitres suivants, nous allons voir que cette description de la configuration de la main peut être utilisée pour construire des systèmes de reconnaissance efficaces de signes statiques et dynamiques ainsi que pour le contrôle par le geste d'un robot mobile.



THANKS TO MIGUEL ...

JORGE CHAM © THE STANFORD DAILY

CHAPITRE 5

Reconnaissance de signes

1. Introduction

Nous étudions maintenant la possibilité de reconnaître des signes statiques et dynamiques à partir de la description lexicale de la configuration de la main, introduite au chapitre précédent.

La section 2 de ce chapitre traite de la reconnaissance des signes statiques. Ces signes ont été définis comme des classes de postures de la main. Leur reconnaissance est particulièrement simplifiée puisque les signes peuvent être efficacement modélisés à l'aide de l'ensemble lexical L_P . Nous verrons que notre système permet de reconnaître efficacement l'alphabet de la Langue des Signes Française.

Dans la section 3, nous nous intéressons à la reconnaissance de signes dynamiques. Ceux-ci sont des classes de gestes ayant une certaine durée dans le temps. Ils ont alors une variabilité temporelle qui rend leur reconnaissance plus délicate que celle des signes statiques. Deux approches sont ici envisagées. Elles sont toutes deux basées sur une normalisation temporelle des gestes, ne nécessitant, comme opération, que le calcul de distances entre postures qui seront définies à partir des opérateurs de distances introduits dans le chapitre 3. La première approche nécessite, dans tous les cas, de disposer d'un ensemble d'apprentissage des signes dynamiques alors que la deuxième approche permet éventuellement de se passer de cet ensemble d'apprentissage. Elle compare, en effet, dans un premier temps des composantes statiques des signes dynamiques, telles que le signe statique de départ ou d'arrivée, qui permettent le plus souvent de discriminer efficacement les différents signes, évitant ainsi d'avoir à effectuer la comparaison de la partie dynamique des signes.

2. Reconnaissance de signes statiques

2.1. Formalisation de la notion de signe statique

2.1.1. Introduction

Les éléments de l'ensemble lexical L_P défini dans le chapitre précédent correspondent à des configurations particulières de la main. Il est alors possible d'utiliser cet ensemble lexical pour formaliser la notion de signes statiques. Mais il serait très restrictif de se limiter aux seuls éléments de L_P . Rappelons qu'une ϕ -symbolisation $\langle E, S, R \rangle$ dont l'univers symbolique contient n symboles définit un ensemble de 2^n concepts ayant tous le même univers de discours, E , et le même univers lexical, $P(S)$. Ces concepts sont les couples $\langle s^*, M^*(s^*) \rangle$, où $s^* \subset S$ et M^* est la signification étendue aux ensembles de symboles associée à $\langle E, S, R \rangle$:

$$\forall s^* \in P(S), \forall e \in E, \quad \mu_{M^*(s^*)}(e) = \bigoplus_{s \in s^*}^{\phi} \mu_{M(s)}(e). \quad (115)$$

On appellera *classes de configurations*, les sous-ensembles c^* de L_P . Toute classe de configurations c^* a donc une signification $M^*(c^*)$ qui est un sous-ensemble flou de postures.

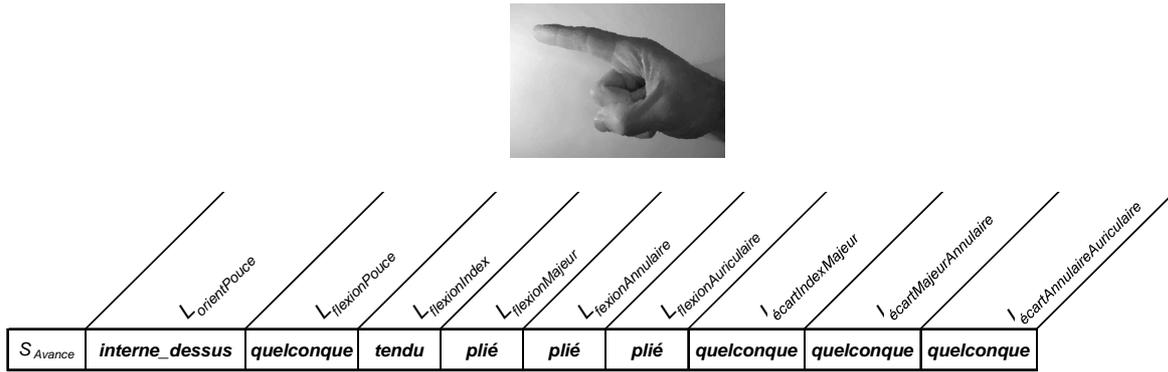
Un signe statique S sera alors défini comme une classe de configurations particulière, c'est-à-dire comme un sous-ensemble de L_P : $S \subset L_P$. Pour une posture p donnée, on définit le taux de reconnaissance du signe S comme étant le degré d'appartenance de cette posture à la signification du signe S dans la symbolisation $\langle P, L_P, R_P \rangle$:

$$rec_S(p) = \mu_{M^*(S)}(p). \quad (116)$$

2.1.2. Exemples

Pour illustrer cette définition des signes statiques, nous prenons l'exemple du signe S_{Avance} ¹. Un prototype de ce signe est représenté à la figure 84. Sa définition est également donnée sur la même figure. Elle reprend les notations introduites au chapitre 3 consistant à noter les singletons d'un ensemble lexical en *italique gras* et à utiliser l'opérateur *ou* pour l'union de ces ensembles et le terme *quelconque* pour représenter l'ensemble lexical lui-même.

1. Le nom de ce signe se réfère à l'application de contrôle d'un robot mobile, qui est présentée dans le chapitre 6.

Fig. 84 Définition du signe S_{Avance} .

Examinons comment se calcule l'appartenance d'une posture p à ce signe. On peut facilement montrer que lorsqu'une classe de configurations c^* a la forme d'un produit cartésien :

$$c^* = c^*_{orientPouce} \times c^*_{flexionPouce} \times \dots \times c^*_{écart_{4,5}}, \quad (117)$$

avec $c^*_{orientPouce} \in P(L_{orientPouce})$, $c^*_{flexionPouce} \in P(L_{flexionPouce})$, ..., $c^*_{écart_{4,5}} \in P(L_{écart})$, alors la fonction d'appartenance de $M^*(c^*)$ peut se calculer avec la formule suivante (cf chapitre 3, paragraphe 3.2.1, avec $\phi = Id$) :

$$\begin{aligned} \forall p \in P, \quad \mu_{M^*(c^*)}(p) &= \mu_{M^*_{orientPouce}(c^*_{orientPouce})}(p) \\ &\cdot \mu_{M^*_{flexionPouce}(c^*_{flexionPouce})}(p) \\ &\cdot \dots \\ &\cdot \mu_{M^*_{écart_{4,5}}(c^*_{écart_{4,5}})}(p). \end{aligned} \quad (118)$$

D'après le tableau de la figure 84, le signe S_{Avance} est une classe de configurations qui a la forme (117) d'un produit cartésien. Le taux de reconnaissance de ce signe pour une posture p s'écrira alors :

$$\begin{aligned} \mu_{M^*_{P(S_{Avance})}}(p) &= \mu_{M^*_{orientPouce}(interne_dessus)}(p) \cdot \mu_{M^*_{flexionPouce}(quelconque)}(p) \\ &\cdot \mu_{M^*_{flexionIndex}(tendu)}(p) \cdot \mu_{M^*_{flexionMajeur}(plié)}(p) \\ &\cdot \mu_{M^*_{flexionAnnulaire}(plié)}(p) \cdot \mu_{M^*_{flexionAuriculaire}(plié)}(p) \\ &\cdot \mu_{M^*_{écart_{2,3}}(quelconque)}(p) \cdot \mu_{M^*_{écart_{3,4}}(quelconque)}(p) \cdot \mu_{M^*_{écart_{4,5}}(quelconque)}(p). \end{aligned} \quad (119)$$

Or, pour toute ϕ -symbolisation $\langle E, S, R \rangle$, le terme *quelconque* correspondant par définition à l'univers symbolique S , on a :

$$\forall e \in E, \quad \mu_{M^*(quelconque)}(e) = \mu_{M^*(S)}(e) = \bigoplus_{s \in S}^{\phi} \mu_{M(s)}(e) = 1. \quad (120)$$

D'autre part, on a également, pour tout symbole s de S , en se rappelant que s correspond à $\{s\}$:

$$\forall e \in E, \quad \mu_{M^*(s)}(e) = \mu_{M^*(\{s\})}(e) = \mu_{M(s)}(e). \quad (121)$$

Finalement, l'appartenance d'une posture p à la signification du signe S_{Avance} est égale à :

$$\begin{aligned} \mu_{M^*P(S_{Avance})}(p) &= \mu_{M_{orientPouce}(interne_dessus)}(p) \cdot \mu_{M_{flexionIndex}(tendu)}(p) \\ &\quad \cdot \mu_{M_{flexionMajeur}(plié)}(p) \cdot \mu_{M_{flexionAnnulaire}(plié)}(p) \cdot \mu_{M_{flexionAuriculaire}(plié)}(p). \\ &= \mu_{D_{orientPouce}(p)}(interne_dessus) \cdot \mu_{D_{flexionIndex}(p)}(tendu) \\ &\quad \cdot \mu_{D_{flexionMajeur}(p)}(plié) \cdot \mu_{D_{flexionAnnulaire}(p)}(plié) \cdot \mu_{D_{flexionAuriculaire}(p)}(plié). \end{aligned} \quad (122)$$

Le fait qu'une posture p corresponde ou non au signe S_{Avance} dépendra donc uniquement de l'orientation du pouce et de la flexion des grands doigts dans cette posture. Aucune restriction n'est donnée pour la flexion du pouce qui peut donc être plié ou tendu. Il n'y a pas non plus de restriction concernant l'écartement des grands doigts puisque lorsque ceux-ci sont pliés, ils sont forcément serrés (contrainte cinématique de la main), quelles que soient les valeurs des descriptions fournies par les symbolisations $\langle P, L_{écart}, R_{écart_{i,j}} \rangle$.

La définition d'un signe ne se présente pas toujours sous la forme d'un produit cartésien. Par exemple, le signe correspondant à la lettre M de la Langue des Signes Française (LSF), représenté à la figure 85, sera défini comme l'union de deux produits cartésiens.



			L_orientPouce	L_flexionPouce	L_flexionIndex	L_flexionMajeur	L_flexionAnnulaire	L_flexionAuriculaire	l_écartIndexMajeur	l_écartMajeurAnnulaire	l_écartAnnulaireAuriculaire
M $= c^*_{M1} \cup c^*_{M2}$	c^*_{M1}	interne_dessus ou interne_dessous	quelconque	tendu	tendu	tendu	tendu	serrés	serrés	serrés	
	c^*_{M2}	interne_dessus ou interne_dessous	quelconque	carré	carré	carré	carré	quelconque	quelconque	quelconque	

Fig. 85 Le signe « M » et la classe de configurations qui lui est associée.

Dans les deux classes c^*_{M1} et c^*_{M2} , les quatre grands doigts se trouvent dans la même configuration : *tendu* pour c^*_{M1} et *carré* pour c^*_{M2} . Dans le cas de c^*_{M2} , il n'est donc pas nécessaire de préciser l'écartement des grands doigts puisque ceux-ci sont forcément serrés. Finalement le signe M correspond à des postures où les quatre grands doigts sont soit tous tendus, soit tous carrés, toujours serrés et où le pouce est rentré à l'intérieur de la main.

Les ensembles c^*_{M1} et c^*_{M2} sont des produits cartésiens et ils sont disjoints. Or on a vu au chapitre 3 que, pour toute *Id*-symbolisation $\langle E, S, R \rangle$ et tout couple (s^*_1, s^*_2) de parties de S tel que $s^*_1 \cap s^*_2 = \emptyset$, on a :

$$\forall e \in E, \quad \mu_{M^*(s^*_1 \cup s^*_2)}(e) = \mu_{M^*(s^*_1)}(e) + \mu_{M^*(s^*_2)}(e). \quad (123)$$

Le taux de reconnaissance du signe statique M pour une posture p est donc la somme des degrés d'appartenance de cette posture aux significations de c^*_{M1} et c^*_{M2} , qui se calculent chacun avec l'équation (118).

Cette méthode de définition des signes statiques va maintenant être utilisée pour définir, puis reconnaître, les signes de l'alphabet de la LSF.

2.2.Reconnaissance de l'alphabet de la Langue des Signes Française (LSF)

2.2.1. L'alphabet de la Langue des Signes Française

Les signes de l'alphabet de la LSF sont présentés à la figure 86.

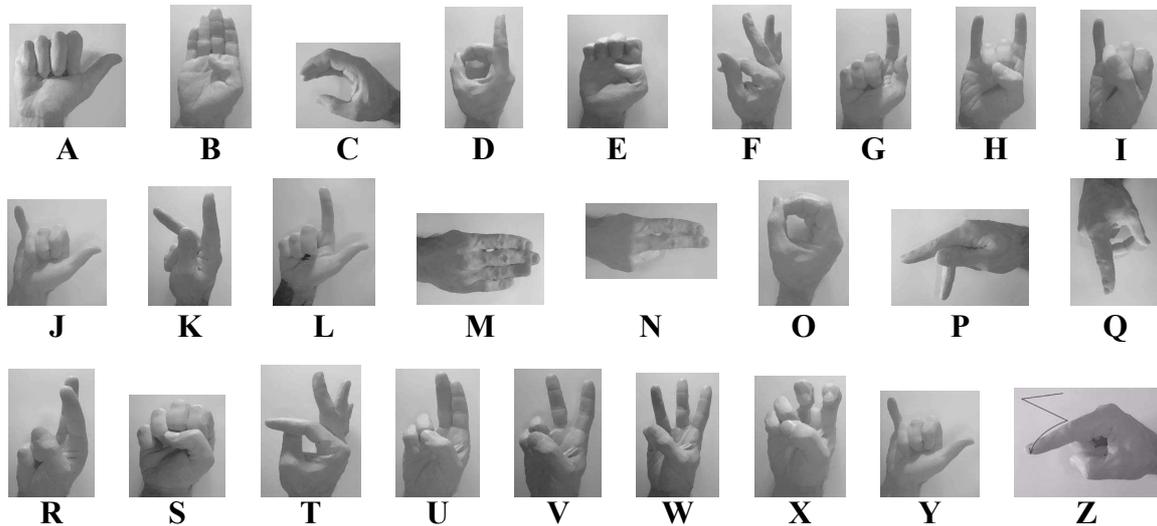


Fig. 86 Les signes de l'alphabet de la LSF.

Chaque signe a été défini comme une classe de configurations de la main, c'est-à-dire comme un sous-ensemble de L_P . Ces définitions sont données dans le tableau de la figure 87. La dernière colonne de ce tableau indique qu'il existe des liens entre les signes ainsi définis : ils ne forment pas une famille d'ensembles disjoints de L_P . La première raison à cela est que les signes de l'alphabet de la LSF ne sont en réalité pas tous des signes statiques. Certains d'entre eux sont accompagnés d'un mouvement global de la main. C'est le cas des signes *K*, *J*, *Y* et *Z*. D'autre part, chaque signe est associé à une orientation particulière de la main. Etant donné que nous ne nous sommes attachés qu'à la description de la configuration de la main sans prendre en compte son orientation, son mouvement, ni sa position, plusieurs de ces signes deviennent indistinguables les uns des autres.

	$L_{\text{orientPouce}}$	$L_{\text{flexionPouce}}$	$L_{\text{flexionIndex}}$	$L_{\text{flexionMajeur}}$	$L_{\text{flexionAnnulaire}}$	$L_{\text{flexionAuriculaire}}$	$I_{\text{ecartIndexMajeur}}$	$I_{\text{ecartMajeurAnnulaire}}$	$I_{\text{ecartAnnulaireAuriculaire}}$	Remarques
A	externe	tendu	plié	plié	plié	plié	quelconque	quelconque	quelconque	
B	interne_dessus ou interne_dessous	quelconque	tendu	tendu	tendu	tendu	serrés	serrés	serrés	$B \subset M$
C	en_avant	quelconque	rond	rond	rond	rond	serrés	serrés	serrés	
D	interne_dessus	quelconque	tendu	rond	rond	rond	quelconque	quelconque	quelconque	
E	interne_dessus ou interne_dessous	plié	griffe	griffe	griffe	griffe	serrés	serrés	serrés	
F	de_coté ou en_avant	tendu	carré	tendu ou carré	tendu	tendu	quelconque	quelconque	quelconque	
G	de_coté	tendu	tendu	plié	plié	plié ou rond	quelconque	quelconque	quelconque	$G \subset Q$
H	interne_dessus ou interne_dessous ou de_coté	quelconque	tendu	plié	plié	tendu ou carré	quelconque	quelconque	quelconque	
I	interne_dessus ou de_coté	quelconque	plié	plié	plié	tendu ou carré	quelconque	quelconque	quelconque	
J	externe ou en_avant	tendu	plié ou rond	plié ou rond	plié ou rond	tendu ou carré	quelconque	quelconque	quelconque	$J = Y$
K	pas tendu	tendu	tendu	carré	plié ou rond	plié ou rond	quelconque	quelconque	quelconque	$K = P$
L	externe	tendu	tendu	plié	plié	plié ou rond	quelconque	quelconque	quelconque	
M	interne_dessus ou interne_dessous	quelconque	tendu	tendu	tendu	tendu	serrés	serrés	serrés	$B = M$
	interne_dessus ou interne_dessous	quelconque	carré	carré	carré	carré	quelconque	quelconque	quelconque	
N	interne_dessus ou interne_dessous	quelconque	tendu	tendu	plié ou rond ou griffe	plié ou rond ou griffe	serrés	quelconque	quelconque	$N = R = U$
O	interne_dessus	quelconque	rond	rond	rond	rond	quelconque	quelconque	quelconque	
P	pas tendu	tendu	tendu	carré	plié ou rond	plié ou rond	quelconque	quelconque	quelconque	$K = P$
Q	interne_dessus ou de_coté	quelconque	tendu ou carré	plié	plié	plié ou rond	quelconque	quelconque	quelconque	$Q \cap Z \neq \emptyset$ $G \subset Q$
R	interne_dessus ou interne_dessous	quelconque	tendu	tendu	plié ou rond ou griffe	plié ou rond ou griffe	serrés	quelconque	quelconque	$N = R = U$
S	interne_dessus	quelconque	plié	plié	plié	plié	quelconque	quelconque	quelconque	
T	interne_dessus ou interne_dessous	tendu	carré ou rond	tendu ou carré	tendu	tendu	quelconque	quelconque	quelconque	
U	interne_dessus ou interne_dessous	quelconque	tendu	tendu	plié ou rond ou griffe	plié ou rond ou griffe	serrés	quelconque	quelconque	$N = R = U$
V	interne_dessus ou interne_dessous	quelconque	tendu	tendu	plié ou rond ou griffe	plié ou rond ou griffe	écartés	quelconque	quelconque	
W	interne_dessus ou interne_dessous	quelconque	tendu	tendu	tendu	plié ou rond ou griffe	écartés	écartés	quelconque	
X	interne_dessus ou interne_dessous	quelconque	griffe	griffe	plié	plié	écartés	quelconque	quelconque	
Y	externe ou en_avant	tendu	plié ou rond	plié ou rond	plié ou rond	tendu ou carré	quelconque	quelconque	quelconque	$J = Y$
Z	interne_dessus ou interne_dessous	quelconque	tendu ou carré	plié	plié	plié ou rond	quelconque	quelconque	quelconque	$Q \cap Z \neq \emptyset$

Fig. 87 Définitions des signes de l'alphabet de la LSF comme des classes de configurations, i.e. des sous-ensembles de L_P

Les signes J et Y ont exactement la même configuration et se différencient normalement par les mouvements qui leur sont associés. Les signes K et P ont également la même configuration, mais le premier s'effectue avec un mouvement global de la main alors que le deuxième est entièrement statique. Les couples de signes (B, M) et (N, U) ne se distinguent l'un de l'autre que par leur orientation.

La confusion entre les signes est amplifiée par le fait que, étant facilement identifiables si l'on prend en compte toutes les composantes du geste de la main, ils ne sont pas réalisés aussi précisément que si leur configuration seule avait du sens. Les définitions que nous avons choisies prennent en compte cette imprécision dans l'exécution des signes et peuvent être considérées comme des définitions « élargies » des signes, correspondant à leur réalisation pratique plutôt qu'à leur stricte définition. Par exemple, le signe M devrait, en toute rigueur, être exécuté avec les doigts tendus (voir Fig. 86). Mais, étant donné qu'il peut facilement être identifié du fait de son orientation, il est généralement réalisé moins précisément, les doigts pouvant se trouver dans la configuration *carré* (voir Fig. 85). D'autre part, les signes Z et Q ont normalement des configurations différentes, mais comme le signe Z est dynamique et le signe Q statique, on peut les distinguer très facilement si on prend en compte l'information de mouvement. Leur configuration n'est alors pas réalisée très précisément et leurs définitions élargies ont une intersection non nulle. C'est également le cas des signes G et Q qui deviennent liés par une relation d'inclusion : $G \subset Q$. Une posture pourra être reconnue comme étant un signe Q et non un signe G , mais l'inverse ne sera pas possible.

Enfin, le signe R s'effectue en croisant l'index et le majeur et ce croisement de doigts ne peut pas être détecté par le Cyberglove. Ce dernier a en effet été conçu pour la reconnaissance de l'alphabet de la Langue des Signes Américaine et cet alphabet ne comporte pas de tels croisements. Le signe R ne peut donc pas être différencié des signes N et U .

Afin de pouvoir plus facilement interpréter les résultats, nous ne travaillerons que sur des classes de configurations deux à deux disjointes. Pour cela, nous regroupons les signes ayant la même définition dans un signe unique. Par exemple, les signes J et Y sont supprimés du tableau et remplacés par le signe $J \cup Y$. Remarquons que l'usage de l'opérateur \cup est ici autorisé puisque les signes sont définis comme des sous-ensembles de l'univers lexical L_P .

Les signes B et M sont supprimés et remplacés par les signes $B \cap M$ et $M \setminus B$. Les signes G , Q et Z sont également supprimés et remplacés par les signes $Q \setminus (G \cup Z)$, $Z \setminus Q$, $Q \cap Z$ et $Q \cap G$. Les définitions de ces signes sont données à la figure 88 où est présenté le nouvel ensemble de signes à reconnaître.

	$L_{orientPouce}$	$L_{flexionPouce}$	$L_{flexionIndex}$	$L_{flexionMajeur}$	$L_{flexionAnnulaire}$	$L_{flexionAuriculaire}$	$I_{ecartIndexMajeur}$	$I_{ecartMajeurAnnulaire}$	$I_{ecartAnnulaireAuriculaire}$
A	externe	tendu	plié	plié	plié	plié	quelconque	quelconque	quelconque
C	en_avant	quelconque	rond	rond	rond	rond	serrés	serrés	serrés
D	interne_dessus	quelconque	tendu	rond	rond	rond	quelconque	quelconque	quelconque
E	interne_dessus ou interne_dessous	plié	griffe	griffe	griffe	griffe	serrés	serrés	serrés
F	de_coté ou en_avant	tendu	carré	tendu ou carré	tendu	tendu	quelconque	quelconque	quelconque
H	interne_dessus ou interne_dessous ou de_coté	quelconque	tendu	plié	plié	tendu ou carré	quelconque	quelconque	quelconque
I	interne_dessus ou de_coté	quelconque	plié	plié	plié	tendu ou carré	quelconque	quelconque	quelconque
L	externe	tendu	tendu	plié	plié	plié ou rond	quelconque	quelconque	quelconque
O	interne_dessus	quelconque	rond	rond	rond	rond	quelconque	quelconque	quelconque
S	interne_dessus	quelconque	plié	plié	plié	plié	quelconque	quelconque	quelconque
T	interne_dessus ou interne_dessous	tendu	carré ou rond	tendu ou carré	tendu	tendu	quelconque	quelconque	quelconque
V	interne_dessus ou interne_dessous	quelconque	tendu	tendu	plié ou rond ou griffe	plié ou rond ou griffe	écartés	quelconque	quelconque
W	interne_dessus ou interne_dessous	quelconque	tendu	tendu	tendu	plié ou rond ou griffe	écartés	écartés	quelconque
X	interne_dessus ou interne_dessous	quelconque	griffe	griffe	plié	plié	écartés	quelconque	quelconque
$B \cap M (= B)$	interne_dessus ou interne_dessous	quelconque	tendu	tendu	tendu	tendu	serrés	serrés	serrés
$M \setminus B$	interne_dessus ou interne_dessous	quelconque	carré	carré	carré	carré	quelconque	quelconque	quelconque
$J \cup Y$	externe ou en_avant	tendu	plié ou rond	plié ou rond	plié ou rond	tendu ou carré	quelconque	quelconque	quelconque
$K \cup P$	pas tendu	tendu	tendu	carré	plié ou rond	plié ou rond	quelconque	quelconque	quelconque
$N \cup R \cup U$	interne_dessus ou interne_dessous	quelconque	tendu	tendu	plié ou rond ou griffe	plié ou rond ou griffe	serrés	quelconque	quelconque
$G \cap Q (= G)$	de_coté	tendu	tendu	plié	plié	plié ou rond	quelconque	quelconque	quelconque
$Q \setminus (G \cup Z)$	de_coté	plié	tendu ou carré	plié	plié	plié ou rond	quelconque	quelconque	quelconque
	de_coté	tendu	carré	plié	plié	plié ou rond	quelconque	quelconque	quelconque
$Z \cap Q$	interne_dessus	quelconque	tendu ou carré	plié	plié	plié ou rond	quelconque	quelconque	quelconque
$Z \setminus Q$	interne_dessous	quelconque	tendu ou carré	plié	plié	plié ou rond	quelconque	quelconque	quelconque

Fig. 88 Définitions des signes que l'on cherche finalement à reconnaître.

2.2.2. Résultats expérimentaux

Les vingt six lettres de l'alphabet ont été épelées par une personne sourde signant couramment. Après une phase de calibrage du gant, chaque signe a été réalisé de façon isolée.

Les taux de reconnaissance de ces signes isolés sont donnés dans le tableau de la figure suivante. Pour chaque posture, le taux de reconnaissance maximum est toujours celui du signe effectivement réalisé. La posture correspondant au signe M a un taux de reconnaissance particulièrement faible. Nous verrons en quoi cela était prévisible et quelle amélioration nous pourrions apporter au formalisme utilisé pour donner une définition plus adaptée à ce signe.

Signes Postures	A	C	D	E	F	H	I	L	O	S	T	V	W	X	$B \cap M$ (=B)	$M \setminus B$	$J \cup Y$	$K \cup P$	$N \cup R$ $\cup U$	$G \cap Q$ (=G)	$Q \setminus (G \cup Z)$	$Z \cap Q$	$Z \setminus Q$
A	0,64	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
B	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,93	-	-	-	-	-	-	-	-
C	-	0,39	-	-	-	-	-	-	0,04	-	-	-	-	-	-	-	0,10	-	-	-	-	-	-
D	-	-	0,92	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
E	-	-	-	1,00	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
F	-	-	-	-	0,67	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
G	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,94	-	-	-
H	-	-	-	-	-	0,98	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,01	-	-
I	-	-	-	-	-	-	0,40	-	-	-	-	-	-	-	-	-	0,11	-	-	-	-	-	-
J	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,87	-	-	-	-	-	-
K	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,92	-	-	-	-	-
L	-	-	-	-	-	-	-	0,62	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
M	-	-	-	-	-	-	-	-	-	0,01	-	-	-	-	-	0,10	-	-	-	-	-	-	-
N	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,09	0,68	-	-	-	-
O	-	-	-	-	-	-	-	-	0,98	-	-	-	-	-	-	-	-	-	-	-	-	-	-
P	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,73	-	-	-	-	-
Q	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,03	0,46	-	-
R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	-	-	-	-
S	-	-	-	-	-	-	-	-	-	0,61	-	-	-	-	-	-	-	-	-	-	-	-	-
T	-	-	-	-	0,01	-	-	-	-	-	0,65	-	-	-	-	-	-	-	-	-	-	-	-
U	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
V	-	-	-	-	-	-	-	-	-	-	-	1,00	-	-	-	-	-	-	0,98	-	-	-	-
W	-	-	-	-	-	-	-	-	-	-	-	-	0,69	-	-	-	-	-	-	-	-	-	-
X	-	-	-	-	-	-	-	-	-	-	-	-	-	0,35	-	-	-	-	-	-	-	-	-
Y	-	-	-	-	-	-	0,03	-	-	-	-	-	-	-	-	-	0,77	-	-	-	-	-	-
Z	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,06	0,27	-	-	0,68

Fig. 89 Reconnaissance des signes isolés. En colonne se trouvent les postures effectuées, en ligne les signes à reconnaître.

Dans un deuxième temps, les signes ont été réalisés à une vitesse normale : ils sont alors connectés. Nous les avons segmentés manuellement pour pouvoir les présenter un par un au système de reconnaissance. Notons que l'on aurait pu utiliser une méthode de segmentation automatique basée sur le fait que les signes, même lorsqu'ils sont connectés, sont marqués par une pause dans la configuration de la main. Si un signe est reconnu durant un certain laps de temps, c'est qu'il correspond bien à une lettre de l'alphabet. Les taux de reconnaissance de ces signes connectés sont donnés dans le tableau de la figure 90. Là encore, pour chaque posture, le taux de reconnaissance maximal est toujours celui du signe effectivement réalisé. On retrouve le problème du signe M qui a un taux de reconnaissance particulièrement bas.

Postures \ Signes	A	C	D	E	F	H	I	L	O	S	T	V	W	X	$B \cap M (=B)$	$M \cap B$	$J \cup Y$	$K \cup P$	$N \cup R \cup U$	$G \cap Q (=G)$	$Q \cap (G \cup Z)$	$Z \cap Q$	$Z \cap Q$
A	0,65	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
B	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	-	-	-	-	-	-	-	-
C	-	0,35	-	-	-	-	-	-	0,07	-	-	-	-	-	-	-	0,25	-	-	-	-	-	-
D	-	-	0,96	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
E	-	-	-	0,81	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
F	-	-	-	-	0,57	-	-	-	-	-	0,08	-	-	-	-	-	-	-	-	-	-	-	-
G	-	-	-	-	-	-	-	0,09	-	-	-	-	-	-	-	-	-	-	-	0,65	-	-	-
H	-	-	-	-	-	1,00	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
I	-	-	-	-	-	-	0,60	-	-	-	-	-	-	-	-	-	0,19	-	-	-	-	-	-
J	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,97	-	-	-	-	-	-
K	-	-	-	-	-	-	-	-	-	-	-	0,23	-	-	-	-	-	0,37	-	-	-	-	-
L	-	-	-	-	-	-	-	0,80	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
M	-	-	-	-	-	-	-	-	-	-	0,03	-	-	-	0,04	0,04	-	-	-	-	-	-	-
N	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,05	0,64	-	-	-	-	-
O	-	-	-	-	-	-	-	-	1,00	-	-	-	-	-	-	-	-	-	-	-	-	-	-
P	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,42	-	-	-	-	-
Q	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,70	0,26
R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1,00	-	-	-	-
S	-	-	-	-	-	-	-	-	-	1,00	-	-	-	-	-	-	-	-	-	-	-	-	-
T	-	-	-	-	0,09	-	-	-	-	-	0,52	-	-	-	-	-	-	-	-	-	-	-	-
U	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,80	-	-	-	-
V	-	-	-	-	-	-	-	-	-	-	-	1,00	-	-	-	-	-	-	-	-	-	-	-
W	-	-	-	-	-	-	-	-	-	-	-	-	1,00	-	-	-	-	-	-	-	-	-	-
X	-	-	-	-	-	-	-	-	-	-	-	-	-	0,12	-	-	-	-	-	-	-	-	-
Y	-	-	-	-	-	-	0,01	-	-	-	-	-	-	-	-	-	0,84	-	-	-	-	-	-
Z	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0,24	0,22	-	-	0,52

Fig. 90 Reconnaissance des signes connectés.

2.2.3. Amélioration possible : le cas du signe M

Dans les deux cas traités, la posture correspondant à un signe M est bien reconnue comme tel puisque le signe ayant le taux de reconnaissance maximal pour cette posture est le signe M . Néanmoins, le taux de reconnaissance de ce signe est particulièrement bas. Tel qu'il a été défini, un signe M correspond en effet à une configuration de la main où les grands doigts sont soit tous tendus, soit tous pliés. Mais la posture de la main représentée à la figure 91 sera mal reconnue comme signe M .



$$\mu_{M^*P(M)}(p_0) = 0.06$$

Fig. 91 Exemple de posture dont l'appartenance à la signification du signe M est particulièrement faible.

Pour comprendre pourquoi cette posture est mal reconnue, nous allons voir le détail du calcul de son appartenance à la signification du signe M . Les descriptions de la flexion des grands doigts dans cette posture sont toutes à peu près égales :

$$D_{flexionIndex}(p_0) \approx D_{flexionMajeur}(p_0) \approx D_{flexionAnnulaire}(p_0) \approx D_{flexionAuriculaire}(p_0) \approx 0.5 / tendu + 0.5 / carré. \quad (124)$$

Cette posture vérifie d'autre part :

$$\begin{aligned} \mu_{M^*_{orientPouce}(interne_dessus\ ou\ interne_dessous)}(p_0) &= \mu_{M^*_{flexionPouce}(quelconque)}(p_0) \\ &= \mu_{M^*_{écart_{2,3}}(serrés)}(p_0) = \mu_{M^*_{écart_{3,4}}(serrés)}(p_0) = \mu_{M^*_{écart_{4,5}}(serrés)}(p_0) = 1.0. \end{aligned} \quad (125)$$

L'appartenance de cette posture à la signification du signe M , qui se calcule à partir des équations (118) et (123), ne dépend donc que de la configuration des grands doigts et vaut environ :

$$\begin{aligned} \mu_{M^*_{p(M)}}(p) &= \mu_{M^*_{flexionIndex}(tendu)}(p) \cdot \mu_{M^*_{flexionMajeur}(tendu)}(p) \\ &\quad \cdot \mu_{M^*_{flexionAnnulaire}(tendu)}(p) \cdot \mu_{M^*_{flexionAuriculaire}(tendu)}(p) \\ &\quad + \mu_{M^*_{flexionIndex}(carré)}(p) \cdot \mu_{M^*_{flexionMajeur}(carré)}(p) \\ &\quad \cdot \mu_{M^*_{flexionAnnulaire}(carré)}(p) \cdot \mu_{M^*_{flexionAuriculaire}(carré)}(p) \\ &\approx 0.5^4 + 0.5^4 = 0.125. \end{aligned} \quad (126)$$

Pour que cette posture p_0 soit reconnue comme un signe M , il faudrait définir celui-ci comme suit :

	L _{orientPouce}		L _{flexionPouce}		L _{flexionIndex}		L _{flexionMajeur}		L _{flexionAnnulaire}		L _{flexionAuriculaire}		L _{écartIndexMajeur}		L _{écartMajeurAnnulaire}		L _{écartAnnulaireAuriculaire}
M	interne_dessus ou interne_dessous	quelconque	tendu ou carré	tendu ou carré	tendu ou carré	tendu ou carré	serrés	serrés	serrés								

Fig. 92 Autre définition possible pour le signe M .

Mais dans ce cas, la posture représentée à la figure 93 serait également reconnue comme un signe M , car ses grands doigts se trouvent tous dans la configuration **plié ou tendu**. Pourtant cette configuration n'est manifestement pas un signe M . Le formalisme introduit jusqu'ici ne permet donc pas de représenter le signe M correctement.



Fig. 93 Cette posture, bien que très différente d'un signe M , serait reconnue comme tel avec la définition de la figure 92.

Une solution serait finalement de définir le signe M en utilisant une contrainte de similarité, imposant que le majeur, l'annulaire et l'auriculaire se trouvent dans une configuration similaire à celle de l'index. La définition du signe M pourrait alors prendre la forme donnée à la figure 94, où la flexion du majeur est spécifiée comme devant être égale à celle de l'index, celle de l'annulaire devant être égale à celle du majeur et celle de l'auriculaire à celle de l'annulaire.

		$L_{orientPouce}$	$L_{flexionPouce}$	$L_{flexionIndex}$	$L_{flexionMajeur}$	$L_{flexionAnnulaire}$	$L_{flexionAuriculaire}$	$L_{ecartIndexM}$	$L_{ecartMajeur}$	$L_{ecartAnnulaireAuriculaire}$
M	interne_dessus ou interne_dessous	quelconque	tendu ou carré	(index)	(majeur)	(annulaire)	serrés	serrés	serrés	

Fig. 94 Définition qu'il serait intéressant de pouvoir donner pour le signe M , permettant de spécifier que les quatre grands doigts doivent se trouver dans la même configuration.

Les contraintes de similarité pourraient être formalisées par la relation d'égalité floue ($\perp_L - T_M$) introduite au chapitre 3. Une orientation intéressante pour de futurs travaux serait donc d'intégrer la possibilité de telles définitions dans le formalisme des symbolisations.

2.3. Conclusion

Les résultats obtenus pour la reconnaissance de signes statiques sont globalement comparables avec ceux obtenus par d'autres systèmes de reconnaissance (voir chapitre 2). Il faut cependant insister sur le fait qu'il est possible de reconnaître un nombre beaucoup plus important de signes à condition qu'ils aient tous des configurations différentes. Nous avons fait le choix de travailler sur l'alphabet de la langue des signes afin d'utiliser un corpus classique de signes et non dans l'objectif d'obtenir un fort taux de reconnaissance sur un grand nombre de signes. D'autre part, une caractéristique très importante de notre approche est qu'une fois la symbolisation $\langle P, L_P, R_P \rangle$ construite, il n'y a pas d'apprentissage des signes, il suffit de donner leur définition sous forme de classe de configurations pour pouvoir ensuite les reconnaître.

Ces résultats sont encourageants et confirment la pertinence de la symbolisation $\langle P, L_P, R_P \rangle$ qui a été définie au chapitre précédent. Dans la section suivante, nous présentons une étude sur la capacité des descriptions linguistiques à différencier des signes dynamiques.

3. Reconnaissance de signes dynamiques

3.1. Introduction

Un geste est une trajectoire dans l'espace des postures P . La mesure d'un geste se fait en mesurant les postures à intervalles de temps réguliers, donnant au final une séquence de postures. La taille de cette séquence varie avec la vitesse d'exécution du geste : plus un geste est effectué lentement, plus la séquence obtenue est longue. Un signe dynamique est une classe de gestes, c'est-à-dire une classe de trajectoires. Le fait qu'un geste G donné soit ou ne soit pas un signe S dépend de la trajectoire de ce geste et non de la vitesse à laquelle il est effectué.

Un signe dynamique S sera représenté par un geste prototype G_S . Pour déterminer si un geste G donné est ou n'est pas le signe S , on le comparera au prototype de S . Cette comparaison doit se faire indépendamment de la vitesse d'exécution du geste. Seulement, si G est effectivement le signe S , mais que sa dynamique d'exécution est différente de celle du prototype G_S , la séquence de postures obtenue pour G sera sensiblement différente de celle de G_S . S'il est effectué plus lentement, elle sera plus longue et s'il est effectué plus rapidement elle sera plus courte. Il se peut également que le geste G soit plus rapide dans sa première partie puis plus lent dans sa seconde partie, donnant au final une séquence de taille identique à celle de G_S , mais sans correspondance avec celle-ci. Ceci est illustré sur la figure suivante où les postures sont représentées comme des éléments d'un espace à une dimension.

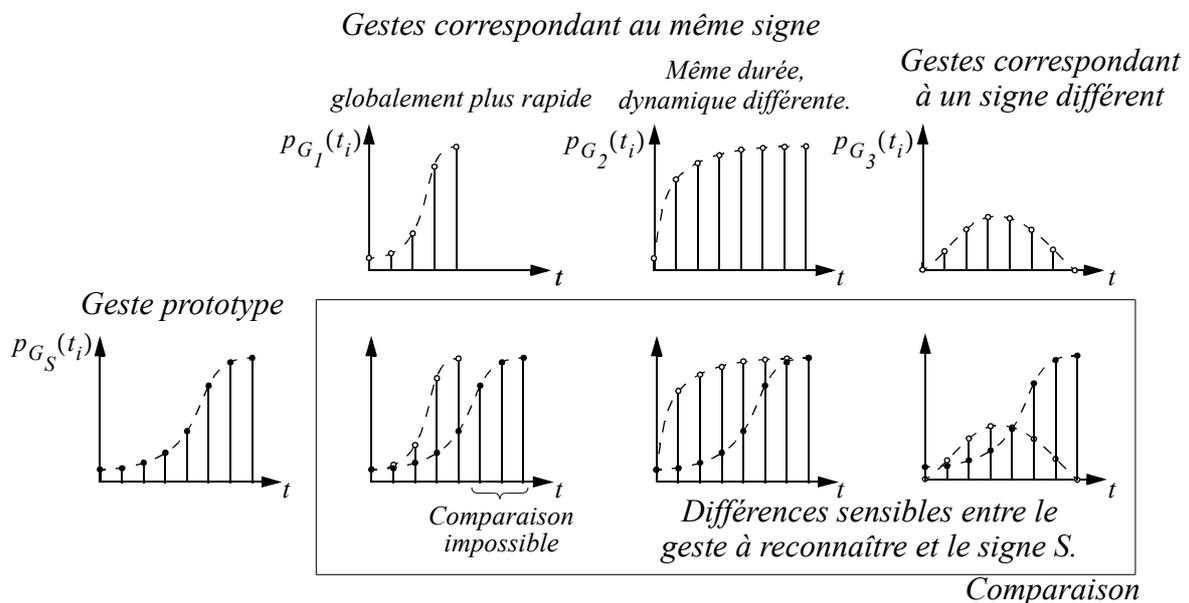


Fig. 95 Le problème de la variabilité temporelle des gestes.

Dans cette figure, trois gestes sont comparés au geste prototype G_S d'un signe S . Le premier de ces gestes, G_1 , correspond au même signe mais est effectué plus rapidement donnant

au final une séquence plus petite et ne pouvant pas être comparée point par point à la séquence prototype. Le deuxième geste, G_2 , correspond également au même signe. Il donne cette fois une séquence de taille identique, mais la différence de la dynamique interne du geste induit une différence sensible dans les deux séquences. Cette différence est alors comparable à celle que l'on a pour le geste G_3 , correspondant à un autre signe.

Il existe plusieurs approches permettant de s'affranchir de cette variabilité temporelle (voir Chapitre 2), nous appliquerons une normalisation des gestes : les séquences seront toutes ramenées à une taille normalisée pour leur comparaison. Il a été montré dans le chapitre 2 en quoi les normalisations utilisées dans la littérature ([Lamar 00] et [Ishikawa 02]) présentent des lacunes qui les rendent peu efficaces si la variation de la vitesse d'exécution des signes n'est pas globale. Ces méthodes ne permettraient pas de reconnaître un geste tel que G_2 dans la figure 95. Nous présentons donc une nouvelle méthode de normalisation qui a été employée avec succès pour la reconnaissance de mouvements [Benoit 03].

La méthode de normalisation, ainsi que la comparaison des gestes normalisés, nécessitent tout d'abord le choix d'un opérateur de distance entre postures. Dans un premier temps, nous présentons les distances entre postures que nous utiliserons. Nous expliquons ensuite comment sont comparés les gestes dynamiques, le principe de la méthode de normalisation et l'opérateur utilisé pour la comparaison de deux gestes normalisés. Enfin, nous appliquons cet opérateur pour la reconnaissance de huit signes dynamiques suivant deux approches qui sont détaillées dans la section 3.4.

3.2. Distances entre postures

Dans la première approche que nous utiliserons pour la reconnaissance, les distances entre postures se calculeront à partir de leurs descriptions dans la symbolisation $\langle P, L_P, R_P \rangle$. Dans la deuxième approche, les distances entre postures ne feront intervenir qu'une partie des configurations des doigts. On comparera les descriptions de ces postures non pas dans la symbolisation $\langle P, L_P, R_P \rangle$, qui est, rappelons-le, le produit cartésien des neuf symbolisations des configurations des doigts, mais dans le produit cartésien d'une partie seulement de ces neuf symbolisations. Par exemple, on pourra ne s'intéresser qu'au mouvement du pouce, les distances entre postures se calculeront alors à partir de leurs descriptions dans la symbolisation suivante :

$$\langle P, L_{Pouce}, R_{Pouce} \rangle = \langle P, L_{flexionPouce}, R_{flexionPouce} \rangle \times \langle P, L_{orientPouce}, R_{orientPouce} \rangle. \quad (127)$$

Deux distances sur les descriptions d'une *Id*-symbolisation ont été définies au chapitre 3. Il s'agit de la distance $d_{(\perp_L - T_M)}$ et la distance du transport, d_{tp} .

3.2.1. Distance $d_{(\perp_L - T_M)}$

a) Utilisation de la symbolisation $\langle P, L_P, R_P \rangle$

Une première solution pour calculer la distance entre deux postures est d'utiliser la distance $d_{(\perp_L - T_M)}$, définie au chapitre 3, sur les descriptions de ces postures dans la symbolisation $\langle P, L_P, R_P \rangle$:

$$\forall (p_1, p_2) \in P^2, \quad d_{(\perp_L - T_M) \circ (D_P \times D_P)}(p_1, p_2) = d_{(\perp_L - T_M)}(D_P(p_1), D_P(p_2)). \quad (128)$$

Rappelons que cette distance se calcule comme suit :

$$\forall (p_1, p_2) \in P^2, \quad d_{(\perp_L - T_M)}(D_P(p_1), D_P(p_2)) = 1 - \sum_{l \in L_P} \min(\mu_{D_P(p_1)}(l), \mu_{D_P(p_2)}(l)). \quad (129)$$

Cette distance est comprise entre 0 et 1. Elle vaut 1 lorsque les descriptions des deux postures comparées sont entièrement disjointes. C'est le cas dès qu'elles ont au moins un doigt qui n'est pas dans la même configuration. Cette distance ne différencie donc pas plus deux postures dont les configurations ne diffèrent que d'un doigt et deux postures dont les configurations sont entièrement différentes. Ceci est illustré à la figure 96. Trois postures sont représentées et correspondent respectivement aux signes statiques *B*, *G* et *L*. Les postures *G* et *L* ne diffèrent l'une de l'autre que par la configuration du pouce qui est *de_coté* dans la première et *externe* dans la seconde. La posture *B* est très différente des deux autres postures, tant par la configuration du pouce qui est *interne_dessous* que par les configurations de trois des grands doigts. Pourtant ces trois postures sont disjointes et la distance $d_{(\perp_L - T_M) \circ (D_P \times D_P)}$ entre toutes ces postures vaut toujours 1.

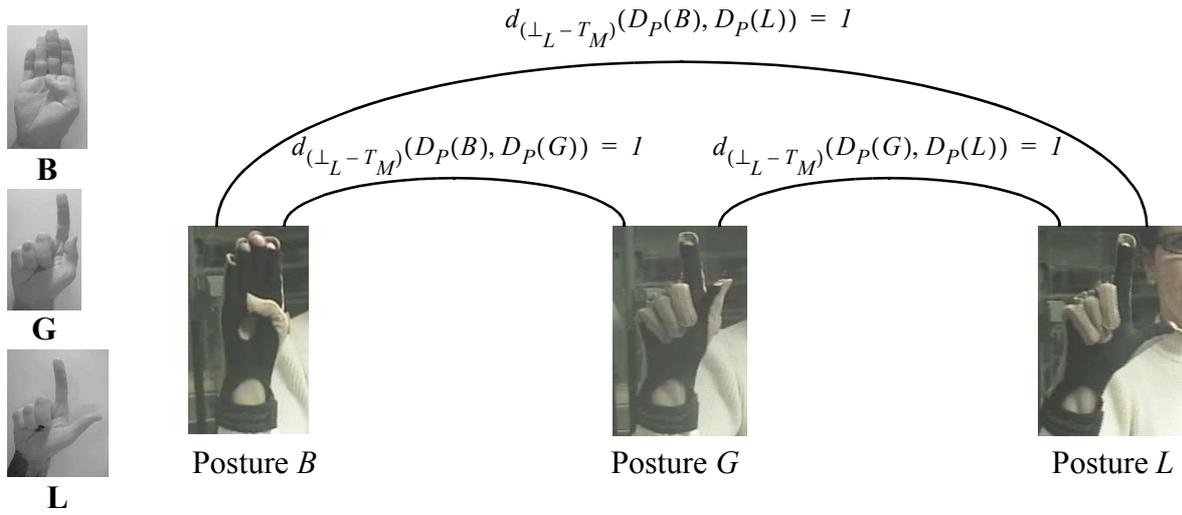


Fig. 96 La distance $d_{(\perp_L - T_M)} \circ (D_P \times D_P)$ est ici saturée : elle ne différencie pas plus G et L que B et G ou B et L.

b) Utilisation d'une symbolisation restreinte

On peut également utiliser la distance $d_{(\perp_L - T_M)}$ sur le produit cartésien d'un ensemble restreint de symbolisations. Il est, par exemple, possible de comparer deux postures uniquement selon la configuration de leur pouce (orientation et flexion). On considérera alors la symbolisation produit $\langle P, L_{Pouce}, R_{Pouce} \rangle$ (127). La distance $d_{(\perp_L - T_M)} \circ (D_{Pouce} \times D_{Pouce})$ donnera sur les postures de la figure 96 :

$$\begin{aligned}
 d_{(\perp_L - T_M)}(D_{Pouce}(B), D_{Pouce}(G)) &= 1, \\
 d_{(\perp_L - T_M)}(D_{Pouce}(B), D_{Pouce}(L)) &= 1, \\
 d_{(\perp_L - T_M)}(D_{Pouce}(G), D_{Pouce}(L)) &= 1.
 \end{aligned} \tag{130}$$

Ainsi, qu'elle s'applique à la description D_P prenant en compte l'ensemble des doigts ou à la description D_{Pouce} ne prenant en compte que la configuration du pouce, la distance $d_{(\perp_L - T_M)}$ entre les postures de la figure 96 est toujours égale à 1. On retrouve ici le défaut de cette distance qui est de ne pouvoir comparer quantitativement que des entités assez proches les unes des autres pour que leurs descriptions ne soient pas disjointes.

3.2.2. Distance du transport

La distance du transport peut également être utilisée pour mesurer les distances entre postures. Pour calculer cette distance, il faut qu'une métrique soit préalablement définie sur l'univers lexical. Nous avons vu qu'il est toujours possible d'utiliser la distance discrète, notée δ_I , définie comme suit sur un ensemble S de symboles :

$$\forall (s_1, s_2) \in S^2, \quad \delta_I(s_1, s_2) = \begin{cases} 1 & \text{si } s_1 \neq s_2, \\ 0 & \text{si } s_1 = s_2. \end{cases} \quad (131)$$

Mais alors, la distance du transport est égale à la distance $d_{(\perp_L - T_M)}$ et l'on se retrouve dans le même cas que la section précédente. Nous préférons utiliser une distance qui ait un lien avec la sémantique des symboles, c'est-à-dire avec leur signification. Nous commençons donc par présenter une méthode permettant de définir une telle distance sur l'univers symbolique d'une ϕ -symbolisation.

a) Construction d'une distance sur l'univers symbolique d'une ϕ -symbolisation

Pour définir la distance sur les symboles d'un ϕ -symbolisation $\langle E, S, R \rangle$, nous construisons une relation sur S , appelée relation d'adjacence et notée \triangleleft , puis nous définissons la distance entre deux symboles comme étant le plus court chemin entre ces symboles dans le graphe associé à cette relation d'adjacence. Rappelons que les ϕ -symbolisations et plus particulièrement les *Id*-symbolisations que nous utilisons dans ce manuscrit sont construites en utilisant trois processus détaillés au chapitre 3 :

- Construction d'une symbolisation sur un intervalle de \mathbb{R} par définition des significations comme des intervalles flous (L-R) ;
- Regroupement de symboles ;
- Produit cartésien.

Pour chacun de ces processus, nous allons donner une méthode permettant d'obtenir une relation d'adjacence sur l'univers symbolique de la symbolisation construite.

(i) Construction sur un intervalle réel

Soit $\langle E, S, R \rangle$ une ϕ -symbolisation dont l'univers de discours est un intervalle de \mathbb{R} , les significations des symboles de S sont, par le choix de construction que nous avons fait au chapitre 3, des intervalles flous (L-R) de E formant une ϕ -partition. La façon dont sont construits ces intervalles flous permet de définir une relation

d'adjacence entre les symboles : deux symboles sont adjacents si leurs significations ont une intersection non nulle.

$$\forall (s_1, s_2) \in S^2, \quad s_1 \# s_2 \Leftrightarrow M(s_1) \cap M(s_2) \neq \emptyset. \quad (132)$$

Sur la figure 9, on a représenté la signification des symboles de L_{IP} dans la symbolisation $\langle X_{IP_1}, L_{IP}, R_{IP_1} \rangle$ ainsi que le graphe d'adjacence sur les symboles, issu de ces significations.

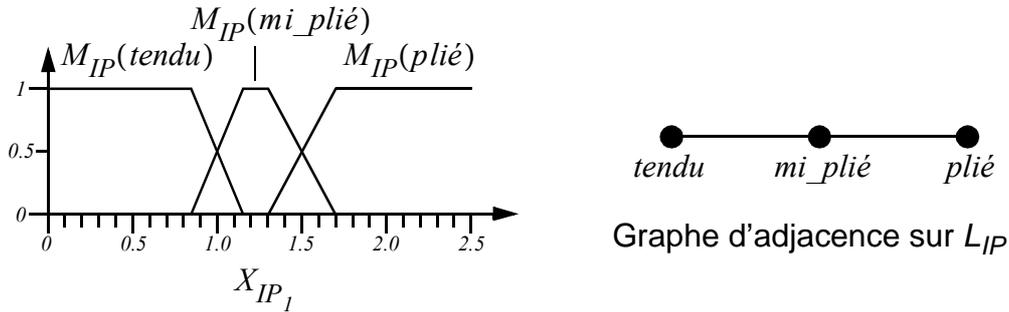


Fig. 97 Exemple d'Id-symbolisation sur un intervalle de \mathbb{R} . Les significations des symboles sont des intervalles flous (L-R) formant une partition stricte. Ces intervalles flous permettent de définir un graphe d'adjacence sur les symboles.

(ii) Produit cartésien

Soit $\langle E_1, S_1, R_1 \rangle$ et $\langle E_2, S_2, R_2 \rangle$ deux ϕ -symbolisations. Nous supposons qu'une relation d'adjacence est définie sur S_1 et sur S_2 . On peut alors définir une relation d'adjacence sur $S_1 \times S_2$ comme suit :

$$\forall ((s_1, s_2), (t_1, t_2)) \in (S_1 \times S_2)^2, \quad (s_1, s_2) \# (t_1, t_2) \Leftrightarrow \begin{cases} (s_1 = t_1) \text{ et } (s_2 \# t_2) \\ \text{ou} \\ (s_2 = t_2) \text{ et } (s_1 \# t_1). \end{cases} \quad (133)$$

Cette construction est illustrée sur l'exemple du produit cartésien de L_{IP} et L_{MCP} à la figure suivante.

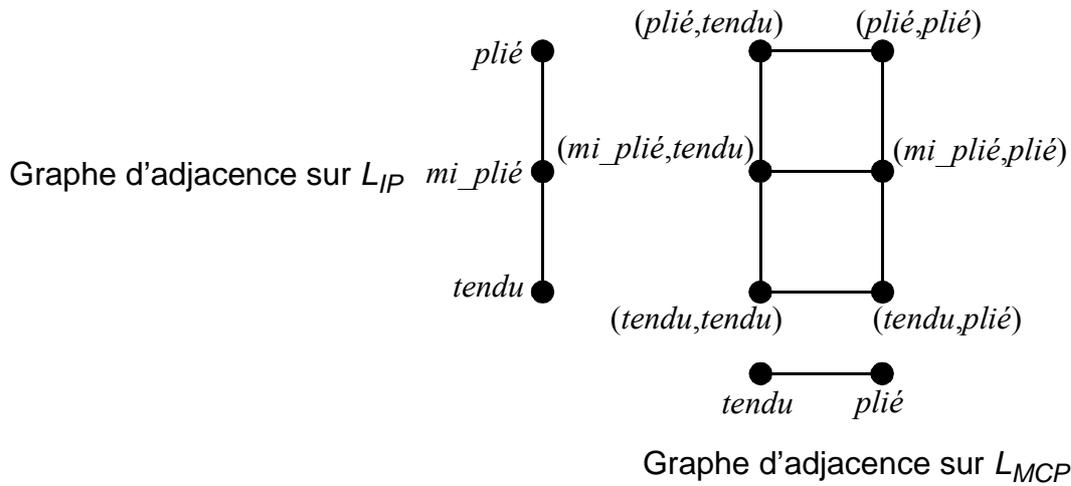


Fig. 98 Construction du graphe d'adjacence sur $L_{IP} \times L_{MCP}$ à partir de ceux existant sur L_{IP} et L_{MCP}

La distance symbolique $d_{S_1 \times S_2}$ issue de la relation $\#$ sur $S_1 \times S_2$ se calcule très simplement en fonction des distances d_{S_1} et d_{S_2} respectivement issues des relations d'adjacence sur S_1 et S_2 :

$$d_{S_1 \times S_2} = d_{S_1} + d_{S_2}. \quad (134)$$

Cette construction peut être étendue au produit cartésien d'un nombre quelconque de symbolisations. La distance symbolique obtenue sur le produit cartésien est alors égale à la somme des distances symboliques définies sur les ensembles le constituant. Ainsi, la distance sur L_P se calculera en sommant les distances définies sur les ensembles lexicaux $L_{orientPouce}$, $L_{flexionPouce}$, $L_{flexionDoigt}$ et $L_{écart}$.

(iii) Regroupement de symboles d'une symbolisation

Considérons une ϕ -symbolisation $\langle E_1, S_1, R_1 \rangle$ et un graphe d'adjacence sur S_1 . Soit $\langle E_2, S_2, R_2 \rangle$ une symbolisation obtenue par regroupement des termes de $\langle E_1, S_1, R_1 \rangle$. Un tel regroupement se fait en choisissant une partition δ de S_1 , puis en définissant une bijection f de δ dans S_2 (voir Fig. 99). La composition de ces deux opérations forme une fonction surjective g , de S_1 dans S_2 .

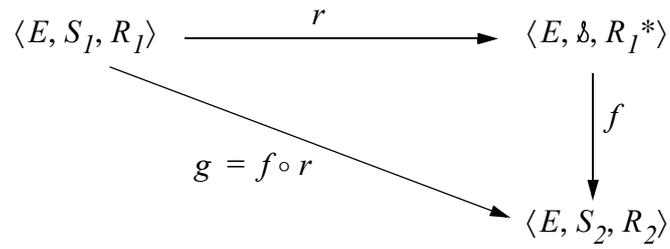


Fig. 99 Principe de la construction d'une symbolisation par regroupement de symboles.

On peut tout d'abord construire une relation d'adjacence sur δ en définissant deux éléments comme adjacents s'ils contiennent des éléments adjacents dans S_1 :

$$\forall (s_1^*, s_2^*) \in \delta^2, \quad s_1^* \mathcal{A} s_2^* \Leftrightarrow \exists (s_1, s_2) \in s_1^* \times s_2^*, s_1 \mathcal{A} s_2. \quad (135)$$

La relation d'adjacence sur S_2 est alors l'image de cette relation sur δ par la bijection f :

$$\forall (t_1, t_2) \in S_2^2, \quad t_1 \mathcal{A} t_2 \Leftrightarrow (f^{-1}(t_1) \mathcal{A} f^{-1}(t_2)). \quad (136)$$

Il est aussi possible de déduire la relation d'adjacence sur S_2 directement à partir de celle sur S_1 par le biais de la fonction g :

$$\forall (t_1, t_2) \in S_2^2, \quad t_1 \mathcal{A} t_2 \Leftrightarrow \exists (s_1, s_2) \in g^{-1}(\{t_1\}) \times g^{-1}(\{t_2\}), s_1 \mathcal{A} s_2, \quad (137)$$

$g^{-1}(\{t\})$ désignant l'ensemble des antécédents de t par g .

La figure 100 représente la construction du graphe d'adjacence de $L_{flexionDoigt}$ à partir de celui de $L_{IP} \times L_{MCP}$ par regroupement de termes.

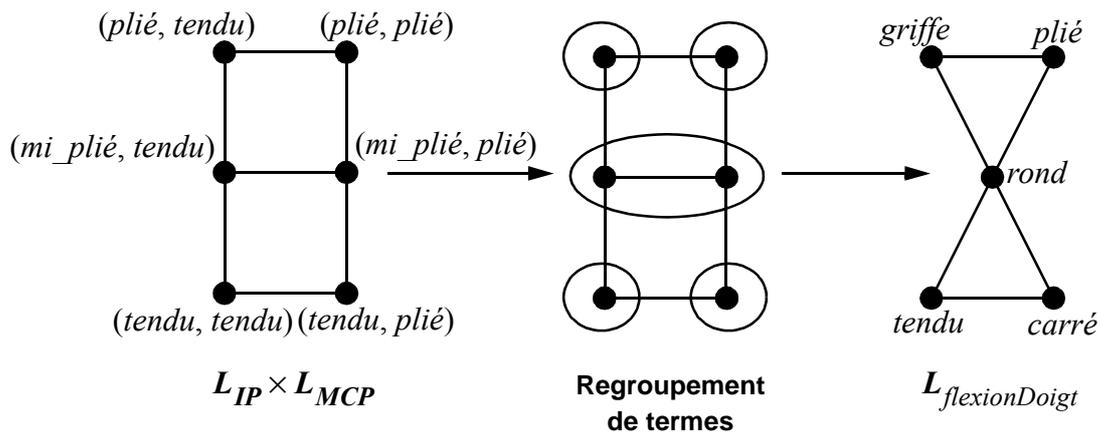


Fig. 100 Obtention d'un graphe d'adjacence par regroupement de termes.

(iv) Graphes d'adjacence obtenus

Les graphes d'adjacence de $L_{orientPouce}$, $L_{flexionPouce}$, $L_{flexionDoigt}$ et $L_{écart}$ obtenus avec cette méthode sont donnés à la figure 101.

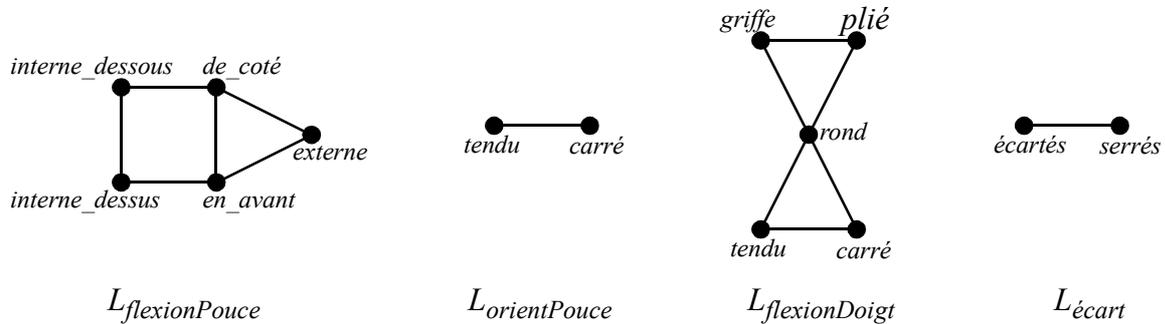


Fig. 101 Graphes d'adjacence obtenus pour les quatre ensembles lexicaux utilisés.

A partir de ces graphes, on peut retrouver simplement la distance entre deux termes d'un ensemble lexical. On aura par exemple :

$$\begin{aligned}
 d_{L_{flexionPouce}}(interne_dessus, externe) &= 2, \\
 d_{L_{flexionPouce}}(de_coté, externe) &= 1.
 \end{aligned}
 \tag{138}$$

Ces quatre distances symboliques sont ensuite utilisées pour la définition de la distance sur L_P .

b) Distance du transport avec la symbolisation $\langle P, L_P, R_P \rangle$

On peut maintenant calculer la distance du transport entre deux descriptions de postures dans la symbolisation $\langle P, L_P, R_P \rangle$. La distance induite entre les postures sera notée $d_{tp \circ (D_P \times D_P)}$ et vaudra :

$$\forall (p_1, p_2) \in P^2, \quad d_{tp \circ (D_P \times D_P)}(p_1, p_2) = d_{tp}(D_P(p_1), D_P(p_2)). \quad (139)$$

Cette distance est appliquée aux postures B , G et L sur la figure 102. La distance entre B et G ou L est, cette fois, bien plus grande que la distance entre G et L . La distance du transport intègre le fait que seule la configuration du pouce varie entre la posture G et la posture L . D'autre part, on remarque que la distance entre B et L est plus grande que celle entre B et G , ce qui est dû au fait que la distance entre les orientations *interne_dessus* et *externe* du pouce est plus grande que la distance entre les orientations *de_coté* et *externe* (Equation (138)).

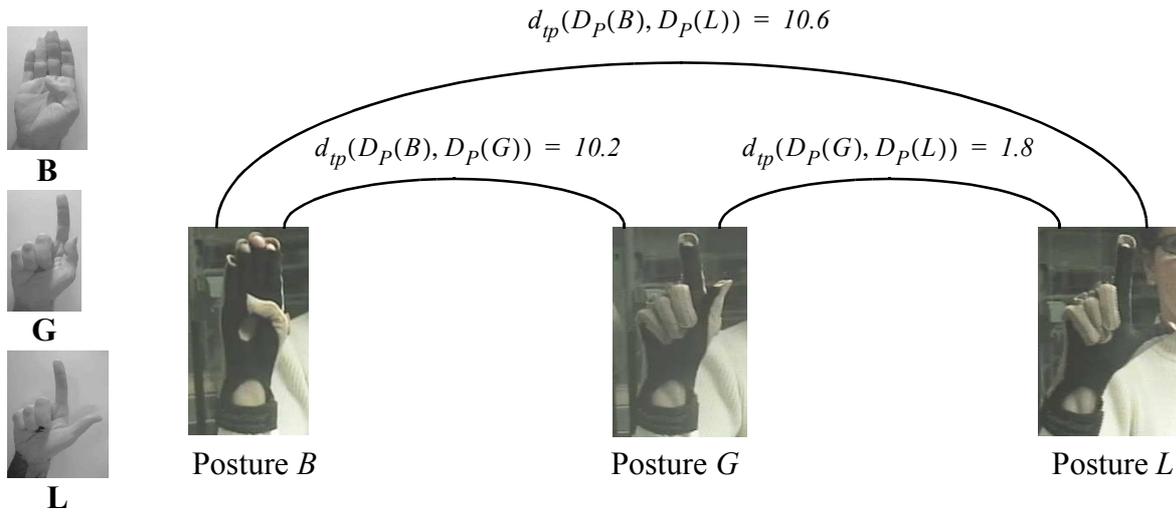


Fig. 102 La distance $d_{tp \circ (D_P \times D_P)}$ entre les postures B , G et L

c) Distance du transport avec une symbolisation restreinte

On peut également utiliser la distance du transport sur une description ne prenant en compte qu'une partie des doigts de la main. Si l'on ne considère, par exemple, que la configuration du pouce, on peut calculer la distance $d_{tp \circ (D_{Pouce} \times D_{Pouce})}$, D_{Pouce} étant la description associée à la symbolisation produit (127). On obtient alors :

$$\begin{aligned}
d_{tp}(D_{Pouce}(B), D_{Pouce}(G)) &= 1.8, \\
d_{tp}(D_{Pouce}(B), D_{Pouce}(L)) &= 2.8, \\
d_{tp}(D_{Pouce}(G), D_{Pouce}(L)) &= 1.
\end{aligned} \tag{140}$$

La distance entre G et L est alors égale à la distance entre les configurations *de_coté* et *externe*, le pouce étant *tendu* dans ces deux postures. Dans la distance entre B et G ou L , on retrouve la distance entre les termes *interne_dessous* et *de_coté* ou *externe* à laquelle s'ajoute une composante correspondant au fait que le pouce est *plié* dans B et *tendu* dans les deux autres postures.

3.2.3. Conclusion

La distance entre deux postures peut donc être calculée de différentes façons. Soit en utilisant la distance $d_{(\perp_L - T_M)}$ sur les descriptions, soit avec la distance du transport. On peut, dans le calcul de chacune de ces distances, considérer l'ensemble des doigts ou seulement une partie de ceux-ci. La distance $d_{(\perp_L - T_M)}$ est peu coûteuse en temps de calcul, mais elle ne permet pas de comparer des postures éloignées les unes des autres. La distance du transport, une fois qu'une distance a été définie sur les symboles, permet de comparer quantitativement deux postures, même éloignées. Son principal défaut est qu'elle n'a pas de formule analytique et que son calcul nécessite la résolution d'un problème de programmation linéaire appelé problème du transport [de Werra 03]. Le temps de calcul devient fortement pénalisant lorsque l'on utilise les descriptions dans la symbolisation $\langle P, L_P, R_P \rangle$ qui sont des sous-ensembles flous de 50 000 éléments.

Nous allons maintenant voir comment comparer deux gestes à partir de ces distances entre postures.

3.3. Distances entre gestes

3.3.1. Normalisation des gestes

Un geste G , une fois échantillonné, est une séquence de postures : $G = (p_1, p_2, \dots, p_{n_G})$. La taille d'un geste G , n_G dépend du signe qui est effectué mais aussi de la vitesse d'exécution à laquelle il est effectué. Pour pouvoir comparer les différentes séquences entre elles, il faut les ramener à une taille normalisée définie a priori, qui sera notée N_0 . La normalisation consiste à rééchantillonner les gestes afin de passer d'un échantillonnage à intervalles de temps égaux à un échantillonnage à déplacements égaux. Pour cela, on calcule la distance cumulée d'une posture i à la posture p_1 du geste G , pour i allant de 1 à n_G :

$$d_c(G, 1) = 0,$$

$$d_c(G, i) = \sum_{k=2}^i d_P(p_k, p_{k-1}), \text{ pour } i \geq 2, \quad (141)$$

où d_P est une distance sur les postures. On en déduit la longueur normalisée d'un segment du geste :

$$l_{nor}(G) = \frac{d_c(G, n_G)}{N_0 - 1}. \quad (142)$$

On peut alors sélectionner dans le geste G les postures p_{i_k} , $k = 1, \dots, N_0$, i_k étant l'indice compris entre 1 et n_G tel que $d_c(G, i_k)$ soit la distance cumulée la plus proche de $(k-1) \times l_{nor}(G)$. On a toujours $p_{i_1} = p_1$ et $p_{i_{N_0}} = p_{n_G}$. Le geste normalisé G_n sera la séquence de postures $(p_{i_1}, p_{i_2}, \dots, p_{i_{N_0}})$. La taille de normalisation N_0 est un paramètre à choisir avec attention. S'il est choisi trop petit la structure du signe ne sera pas conservée dans le geste normalisé, par un effet analogue à celui du théorème d'échantillonnage de Shannon. Si N_0 est choisi trop grand, l'effet de la normalisation sera fortement amoindri.

Sur la figure suivante, nous reprenons les gestes de la figure 95 et leur appliquons cette normalisation avec $N_0 = 5$. Les différentes étapes de la normalisation sont détaillées ci-dessous.

- Etape (0) : Calcul des distances cumulées des postures des gestes. Les postures sont représentées comme des points d'un espace de dimension 1. La longueur d'un segment d'un geste est donc la différence de hauteur entre les deux postures à l'extrémité de ce segment. Les distances cumulées sont portées en ordonnée. Un pointillé relie chaque posture avec sa distance cumulée. Les distances cumulées normalisées sont représentées par un trait fin à gauche de l'axe des ordonnées.
- Etapes (1) - (5) : Sélection de la posture p_{i_k} , $k = 1, \dots, 5$, telle que $d_c(G, i_k)$ soit la distance cumulée la plus proche de $(k-1) \times l_{nor}(G)$. Cette distance cumulée normalisée est mise en trait gras. La posture sélectionnée est celle dont la distance cumulée à p_1 est la plus proche de ce trait gras. Elle est également mise en trait gras et est reportée dans le geste normalisé, représenté à gauche du geste original. La même posture peut être sélectionnée plusieurs fois de suite. C'est le cas par

exemple de la posture p_5 du geste prototype G_S . Elle est sélectionnée comme deuxième et troisième posture du geste normalisé car sa distance cumulée est à la fois la plus proche de $(2 - 1) \times l_{nor}(G_S)$ (étape 2) et de $(3 - 1) \times l_{nor}(G_S)$ (étape 3). On a donc $i_2 = i_3 = 5$.

- Etape (6) : Les gestes normalisés sont comparés au geste prototype, également normalisé. Le geste G_1 , dont la séquence de postures est plus courte que la séquence du prototype, peut maintenant être comparé au prototype. Une fois normalisé, il présente très peu de différences avec le geste prototype. La normalisation du geste G_2 donne un geste également proche du geste prototype normalisé. Par contre, la normalisation du geste G_3 donne un geste très différent du geste prototype.

Dans cette procédure de normalisation, les couples de postures dont on calcule la distance sont toujours des postures consécutives d'un geste. La fréquence d'échantillonnage (100 Hz) étant élevée par rapport à la vitesse d'exécution des gestes, deux postures consécutives sont généralement assez proches l'une de l'autre et leurs descriptions sont rarement disjointes. La distance $d_{(\perp_L - T_M)}$ ne sera alors pas saturée et la normalisation avec cette distance s'effectuera dans de bonnes conditions.

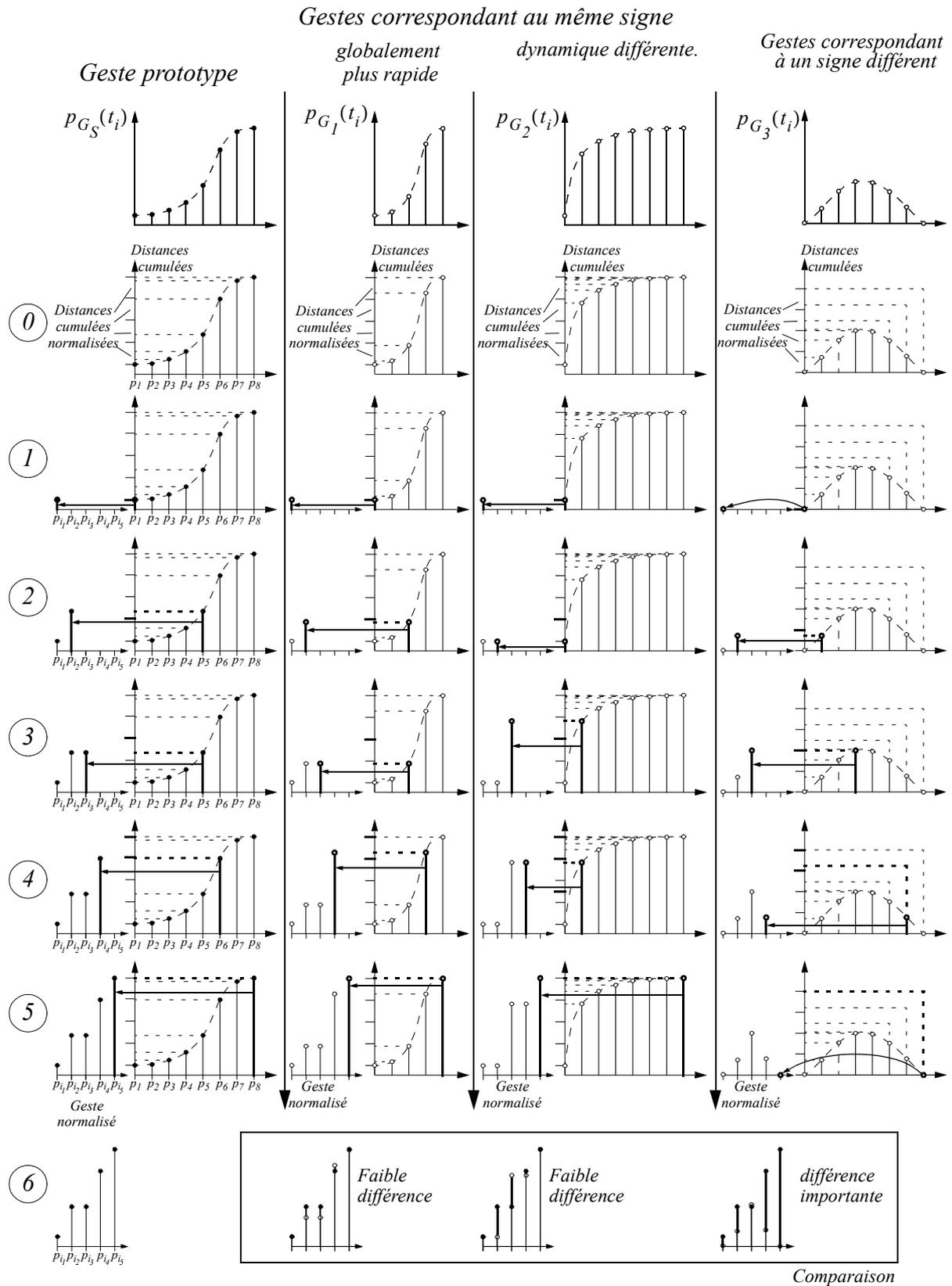


Fig. 103 Normalisation des gestes illustrée sur quatre exemples.

3.3.2. Distance entre gestes

Si N_0 est la taille de normalisation choisie, tout geste normalisé peut alors être vu comme un N_0 -uplet de postures ou comme un élément de $P^{N_0} = \overbrace{P \times \dots \times P}^{N_0 \text{ fois}}$. Etant donné qu'une distance est choisie sur P , ce dernier est un espace métrique. P^{N_0} est alors un produit cartésien d'espaces métriques. Nous rappelons ici un résultat sur les produits cartésiens de ces espaces [Berge 66] :

Proposition 21 : Soit $I = \{1, \dots, N\}$ et $\{(E_i, d_i), i \in I\}$ une famille de N espaces métriques. On note $E = E_1 \times \dots \times E_N$ le produit cartésien des ensembles E_i . Soit $k \in \mathbb{N}^* \cup \{\infty\}$ et d^k l'opérateur binaire sur E défini comme suit :

$$\text{Si } k \in \mathbb{N}^*, \forall (e, f) \in E^2, \quad d^k(e, f) = \left(\sum_{i \in I} d_i(e_i, f_i)^k \right)^{1/k}, \quad (143)$$

$$\text{et } \forall (e, f) \in E^2, \quad d^\infty(e, f) = \max_{i \in I} d_i(e_i, f_i), \quad (144)$$

où $e_1 = (e_1, \dots, e_N)$ et $f = (f_1, \dots, f_N)$. Alors (E, d^k) est un espace métrique.

Cette proposition fournit donc toute une famille de distances utilisables sur les gestes normalisés. On a d'autre part :

$$\forall (e, f) \in E^2, \quad d^\infty(e, f) \leq \dots \leq d^2(e, f) \leq d^1(e, f). \quad (145)$$

Nous n'utiliserons que les deux distances extrêmes correspondant à $k = 1$ et $k = \infty$. Ainsi, pour tout couple de gestes normalisés $G_{n1} = (p_{1,1}, \dots, p_{1,N_0})$ et $G_{n2} = (p_{2,1}, \dots, p_{2,N_0})$, en notant d_P la distance choisie sur les postures, on considèrera les deux distances suivantes :

$$d_G^1(G_{n1}, G_{n2}) = \sum_{i \in \{1, \dots, N_0\}} d_P(p_{1,i}, p_{2,i}). \quad (146)$$

$$d_G^\infty(G_{n1}, G_{n2}) = \max_{i \in \{1, \dots, N_0\}} d_P(p_{1,i}, p_{2,i}). \quad (147)$$

Ces distances vont maintenant être utilisées pour la reconnaissance de signes dynamiques.

3.4. Reconnaissance

Les signes que nous étudions sont représentés à la figure 104. Sur cette figure, le symbole $\times 2$ signifie que le mouvement correspondant est effectué deux fois. Ainsi, dans le signe *Clic*, le pouce ne se replie qu'une seule fois tandis que dans le signe *Clic2*, il se replie deux fois. La reconnaissance est testée sur deux ensembles de gestes, *GestesN* et *GestesR*, correspondant respectivement à une réalisation en vitesse normale des huit signes et à une réalisation plus rapide de ces mêmes signes.

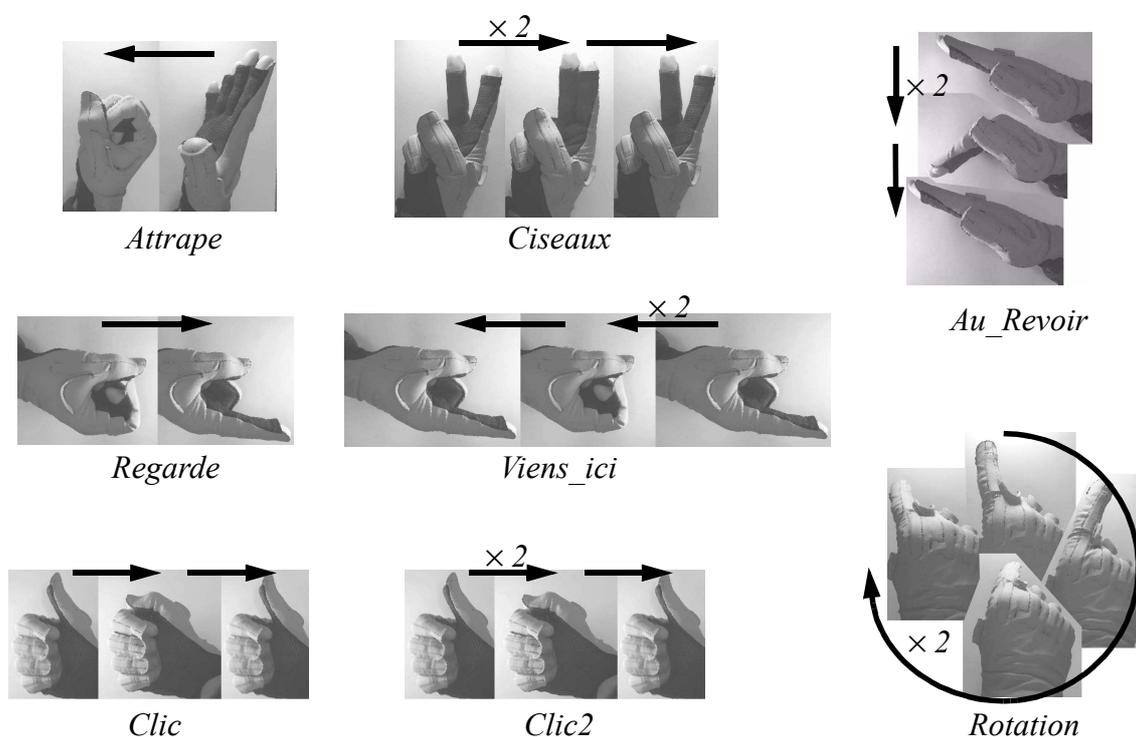


Fig. 104 Huit signes dynamiques.

Deux approches sont utilisées pour la reconnaissance. La première ne fait qu'associer un prototype à chaque signe. La seconde décrit chaque signe par quatre composantes : les postures de début et de fin de signe, une composante statique et une composante dynamique.

3.4.1. Première approche pour la reconnaissance

Dans cette première approche, les postures sont comparées en utilisant l'ensemble des doigts de la main, c'est-à-dire en utilisant les descriptions dans la symbolisation $\langle P, L_P, R_P \rangle$. Pour la classification des gestes, nous utilisons la règle du plus proche voisin (1-PPV). Un geste

prototype est associé à chaque signe. L'enregistrement de ces gestes correspond à la phase d'apprentissage. Dans la phase de classification, les gestes à reconnaître sont comparés à tous les prototypes. Ils sont, pour cela, normalisés avec une taille de normalisation N_0 égale à 10. Puis leur distance à chaque prototype est calculée. On leur attribue alors le signe dont le prototype est le plus proche.

Dans le tableau de la figure suivante, nous donnons les valeurs de la distance d_G^1 entre les gestes de la série *GestesR* et les gestes prototypes, pour les deux choix de distance entre postures. Pour chaque geste de *GestesR*, la plus faible des distances aux prototypes est indiquée en caractère gras. Elle correspond au signe reconnu. On trouvera en annexe des tableaux analogues pour la distance d_G^∞ ainsi que pour la série de geste *GestesN*.

Tous les signes sont correctement reconnus. De plus, les distances obtenues entre les différents gestes correspondent à des ressemblances existant entre les signes correspondants. Un geste correspondant au signe *Clic* se retrouve proche d'un geste correspondant au signe *Clic2*. La série *GestesN* donne des résultats équivalents. Seule la distance d_G^∞ lorsqu'elle est associée à $d_{(\perp_L - T_M) \circ (D_P \times D_P)}$ comme distance entre postures donne de mauvais résultats (voir tableau en annexe). Lorsque l'on calcule cette distance entre deux gestes, il suffit qu'un couple de postures soit un peu trop éloigné et la distance finale vaut 1. Le phénomène de saturation est alors prépondérant et nuit à la reconnaissance des signes.

Mais cette méthode de reconnaissance n'est pas applicable en pratique. Le fait d'utiliser les descriptions dans la symbolisation $\langle P, L_P, R_P \rangle$ pour comparer les postures implique de manipuler des sous-ensembles flous d'un ensemble lexical de 50 000 éléments. Les calculs de distance entre postures sont alors très longs, particulièrement lorsque l'on utilise la distance du transport. Or, dans les signes que nous étudions, il n'y a généralement qu'une partie des doigts qui est en mouvement. On peut donc améliorer le système de reconnaissance en appliquant la procédure de normalisation uniquement sur les doigts qui sont en mouvement. D'autre part, il est possible de caractériser un signe dynamique par certaines composantes qui ne sont pas soumises à la variabilité temporelle. C'est ce que l'on fait dans la deuxième méthode de reconnaissance.

		Gestes comparés avec la distance d_G^I Postures comparées avec la distance $d_{(\perp_L - T_M) \circ (D_P \times D_P)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesR	Au Revoir	3,0	10,0	9,9	10,0	10,0	10,0	10,0	10,0
	Viens ici	10,0	6,0	9,7	10,0	10,0	7,3	9,6	10,0
	Attrape	9,9	9,9	6,2	10,0	10,0	10,0	10,0	10,0
	Clic2	10,0	10,0	9,9	2,3	4,2	10,0	10,0	10,0
	Clic	10,0	10,0	10,0	4,6	2,6	10,0	10,0	10,0
	Regarde	10,0	8,5	10,0	10,0	10,0	5,7	9,5	10,0
	Rotation	10,0	9,7	10,0	10,0	10,0	9,2	5,0	10,0
	Ciseaux	10,0	10,0	10,0	10,0	10,0	10,0	10,0	6,1

		Gestes comparés avec la distance d_G^I Postures comparées avec la distance $d_{r \circ (D_P \times D_P)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesR	Au Revoir	4,9	108,6	68,5	92,3	95,7	105,1	99,1	80,9
	Viens ici	114,2	11,8	66,9	49,2	47,0	20,9	26,9	47,1
	Attrape	71,3	62,9	16,3	62,2	59,4	64,1	61,3	62,2
	Clic2	93,6	50,9	65,3	4,9	8,4	40,0	54,4	77,9
	Clic	95,2	48,2	60,9	10,4	4,8	35,8	49,8	72,9
	Regarde	112,8	23,9	72,2	46,1	44,5	13,1	31,2	55,1
	Rotation	102,7	30,0	65,1	58,6	55,1	33,8	12,2	38,0
	Ciseaux	90,7	47,5	67,7	78,7	76,0	54,8	39,6	13,7

Fig. 105 Résultat de la comparaison des gestes de la série GestesR avec la distance d_G^I .

3.4.2. Deuxième approche pour la reconnaissance

a) Modèle de signe dynamique

Dans cette deuxième approche, nous utilisons un modèle des signes dynamiques se rapprochant du modèle Stokoe [Stokoe 80]. Ce modèle de Stokoe s'applique aux gestes de la main dans son ensemble, prenant en compte sa configuration mais aussi sa position, son orientation et son mouvement. Il décompose les signes dynamiques en une posture initiale et une posture finale.

Ces postures particulières, correspondant au début et à la fin d'un signe dynamique S , sont des composantes invariantes de ce signe. Elles peuvent être modélisées comme des signes statiques. Ces signes seront notés D_S et F_S . Un geste G , ne pourra alors être reconnu comme signe S que si ses postures de départ et d'arrivée correspondent à D_S et F_S . Le modèle de Stokoe ne considère que le mouvement de la main entre ces deux postures et ne s'intéresse pas à la dynamique de la configuration de la main. Dans notre cas, nous considérons uniquement la configuration de la main et ne pouvons pas ignorer ce qui se passe entre la posture initiale et la posture finale d'un signe dynamique. Par contre, pour la plupart des signes dynamiques que nous étudions, une partie des doigts reste immobile. Ces doigts sont dans une configuration

particulière que l'on peut également modéliser comme un signe statique et qui caractérise aussi le signe dynamique. Ce signe sera appelé la composante statique du signe S et noté St_S . Les trois signes D_S , F_S et St_S formeront ce que l'on appellera les caractéristiques statiques du signe S . Enfin, un signe dynamique a une composante dynamique. Celle-ci sera représentée par un geste prototype. Mais seuls les doigts en mouvement dans ce signe seront pris en compte dans la comparaison des gestes à reconnaître avec ce prototype. On utilisera donc les descriptions des postures dans une symbolisation restreinte que l'on note $\langle P, L_S, R_S \rangle$.

Prenons l'exemple du signe *Regarde* (voir Fig. 104). Les signes statiques de début et de fin $D_{Regarde}$ et $F_{Regarde}$, ainsi que la composante statique $St_{Regarde}$ sont donnés à la figure suivante. Dans ce signe dynamique, seul l'index est en mouvement. Sa configuration n'intervient donc pas dans la composante statique $St_{Regarde}$ et la symbolisation restreinte utilisée pour la comparaison des gestes avec un prototype de *Regarde* est :

$$\langle P, L_{Regarde}, R_{Regarde} \rangle = \langle P, L_{flexionDoigt}, R_{flexionIndex} \rangle. \quad (148)$$

	$L_{orientPouce}$	$L_{flexionPouce}$	$L_{flexionIndex}$	$L_{flexionMajeur}$	$L_{flexionAnnulaire}$	$L_{flexionAuriculaire}$	$I_{ecartIndexMajeur}$	$I_{ecartMajeurAnnulaire}$	$I_{ecartAnnulaireAuriculaire}$
$D_{Regarde}$	interne_dessus ou en_avant	quelconque	plié ou griffe	plié	plié	plié	quelconque	quelconque	quelconque
$F_{Regarde}$	interne_dessus ou en_avant	quelconque	tendu ou carré	plié	plié	plié	quelconque	quelconque	quelconque
$St_{Regarde}$	interne_dessus ou en_avant	quelconque	quelconque	plié	plié	plié	quelconque	quelconque	quelconque

Fig. 106 Signes statiques de début et de fin et composante statique du signe « Regarde ».

Ainsi tout signe statique S sera défini par quatre composantes : ses signes statiques de début et de fin, D_S et F_S , sa composante statique St_S et sa symbolisation $\langle P, L_S, R_S \rangle$. On trouvera en annexe les définitions de ces quatre composantes, pour les huit signes dynamiques considérés.

b) Reconnaissance

La phase de reconnaissance se fait en deux étapes. Pour qu'un geste G soit reconnu comme un signe S , il faut tout d'abord qu'il vérifie les caractéristiques statiques de ce signe. Sa première et sa dernière posture doivent correspondre aux signes statiques de départ et de fin de

S et l'ensemble de ses postures à la composante statique du signe. Ces conditions s'écriront plus formellement comme suit :

Soit $G = (p_1, p_2, \dots, p_{n_G})$ un geste à reconnaître et S un signe dynamique. On dira que

G vérifie les caractéristiques statiques de S s'il vérifie :

$$\mu_{M^*(D_S)}(p_1) \geq \alpha, \quad (149)$$

$$\mu_{M^*(F_S)}(p_{n_G}) \geq \alpha, \quad (150)$$

$$i = \min_{1, \dots, n_G} \mu_{M^*(St_S)}(p_i) \geq \alpha, \quad (151)$$

où α est un seuil que l'on choisit égal à 0.1. Ce seuil correspond au degré de reconnaissance au delà duquel on considère qu'un signe statique est reconnu. Il est donc le même pour ces trois conditions puisqu'il n'y a pas de raison particulière d'être plus restrictif pour le signe de début de geste, de fin de geste ou pour la composante statique. La condition (151) signifie que chaque posture p_i du geste G doit être reconnue comme étant la composante statique. Le taux de reconnaissance de St_S pour chaque posture de G doit être supérieur à α , d'où l'utilisation de l'opérateur *min*.

Si un geste G vérifie les caractéristiques statiques d'un signe S , alors on compare sa dynamique avec celle du geste prototype G_S de S , en utilisant, pour le calcul des distances entre les postures de ces deux gestes, les descriptions dans la symbolisation $\langle P, L_S, R_S \rangle$. Cette comparaison est schématisée à la figure 107, où D est la description associée à la symbolisation $\langle P, L_S, R_S \rangle$.

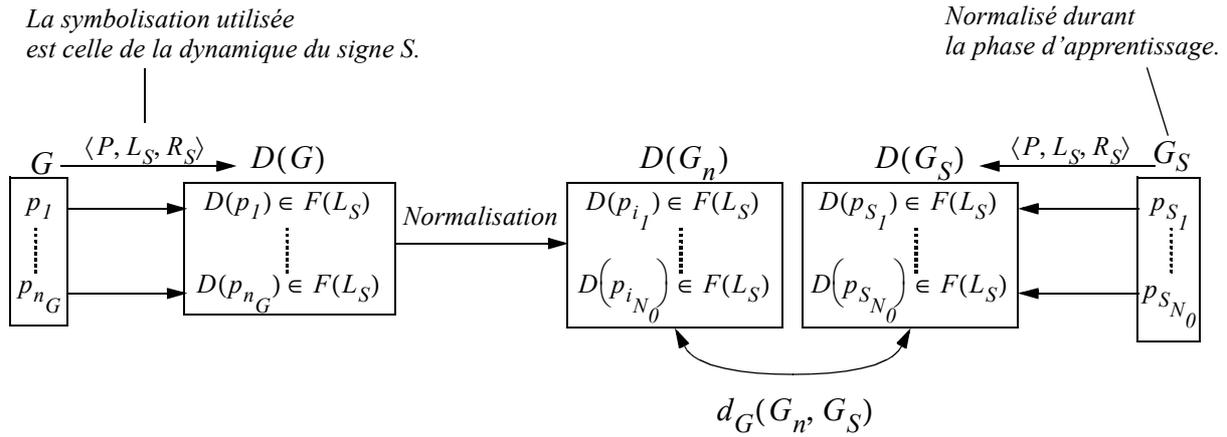


Fig. 107 Principe de la comparaison de la dynamique d'un geste avec celle d'un signe S .

c) Résultats

La figure 108 donne, pour chaque geste de la série $GestesR$, son adéquation - ou non - avec les caractéristiques statiques des différents signes. Ce tableau montre que ces caractéristiques statiques sont suffisantes pour reconnaître la plupart des signes. Les signes *Clic* et *Clic2* ne peuvent pas être différenciés puisqu'ils ont les mêmes caractéristiques statiques. Ce résultat est très encourageant car la vérification de l'adéquation d'un geste avec les caractéristiques statiques d'un signe dynamique nécessite très peu de calculs et permet d'envisager de faire de la reconnaissance de signes dynamiques en temps réel. On trouvera les résultats concernant la série de gestes $GestesN$ en annexe.

		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesR	Au Revoir	X	-	-	-	-	-	-	-
	Viens ici	-	X	-	-	-	-	X	-
	Attrape	-	-	X	-	-	-	-	-
	Clic2	-	-	-	X	X	-	-	-
	Clic	-	-	-	X	X	-	-	-
	Regarde	-	-	-	-	-	X	-	-
	Rotation	-	-	-	-	-	-	X	-
	Ciseaux	-	-	-	-	-	-	-	X

Fig. 108 Adéquation des gestes de $GestesR$ avec les trois caractéristiques statiques des différents signes.

Normalement, on ne calcule la distance entre un geste et le prototype d'un signe que si ce geste vérifie les caractéristiques statiques du signe en question. Cependant, on donne ici l'ensemble des distances des gestes de la série $GestesR$ avec les prototypes afin d'illustrer le

calcul de ces distances. Ces résultats sont donnés à la figure 109 pour la distance d_G^I . On trouvera l'ensemble des résultats en annexe.

		Gestes comparés avec la distance d_G^I Postures comparées avec la distance $d_{(\perp_L - T_M) \circ (D \times D)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesR	Au Revoir	2,9	10,0	8,8	10,0	10,0	10,0	10,0	10,0
	Viens ici	7,7	2,9	6,4	8,8	8,7	6,1	8,0	7,3
	Attrape	8,7	9,4	4,5	9,9	10,0	9,8	10,0	9,8
	Clic2	5,7	10,0	8,9	1,2	3,6	9,9	9,5	9,6
	Clic	5,2	9,9	8,8	3,5	1,8	9,9	9,4	9,5
	Regarde	7,7	5,6	7,4	8,3	8,1	2,4	8,0	7,3
	Rotation	7,4	9,9	9,3	10,0	10,0	9,0	3,8	7,7
	Ciseaux	6,1	3,8	5,2	6,1	6,1	4,8	6,0	0,8

		Gestes comparés avec la distance d_G^I Postures comparées avec la distance $d_{tr \circ (D \times D)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesR	Au Revoir	4,8	69,7	41,8	78,9	79,7	70,0	62,2	41,0
	Viens ici	9,4	1,8	7,7	12,7	12,7	7,4	9,7	8,7
	Attrape	34,8	27,4	6,5	28,8	28,8	31,2	34,8	39,7
	Clic2	7,8	17,7	14,7	2,1	4,5	16,5	17,3	18,3
	Clic	6,7	17,8	14,9	4,4	2,6	16,2	16,6	17,7
	Regarde	10,8	7,8	12,6	11,0	10,8	1,8	11,1	10,4
	Rotation	9,3	16,4	17,8	24,8	24,8	15,4	5,3	11,1
	Ciseaux	6,1	3,8	5,2	6,1	6,1	4,8	6,0	0,8

Fig. 109 Résultat de la comparaison des gestes de la série GestesR avec la distance d_G^I . D est la description associée à la symbolisation $\langle P, L_S, R_S \rangle$.

Ces résultats indiquent que tous les signes auraient pu être reconnus en ne comparant que leur composante dynamique. Mais si des caractéristiques statiques ont été définies pour les signes, ce qui est le cas ici, il n'y a pas d'intérêt à se priver de l'information qu'elles fournissent. Par contre, il aurait été possible de ne pas définir ces caractéristiques statiques et de reconnaître efficacement les signes, avec un nombre réduit de calculs en ne précisant que les doigts en mouvement dans chaque signe. On retrouve également des distances plus faibles entre les gestes correspondant à des signes se ressemblant.

4. Conclusion

La description lexicale de la configuration de la main construite au chapitre 4 a pu être utilisée pour reconnaître des signes statiques ainsi que des signes dynamiques. Nous rappelons ci-dessous la structure générale d'un système de reconnaissance de signes.

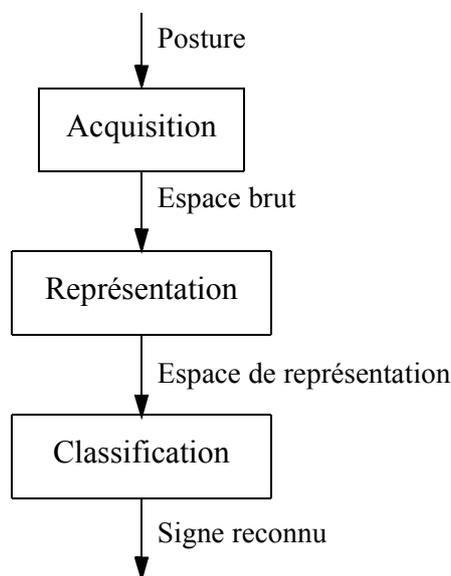


Fig. 110 Les trois étapes d'un système de reconnaissance de signes.

L'étape de représentation correspond, dans notre cas, à la transformation des données numériques issues du Cyberglove en neuf descriptions lexicales. L'espace de représentation est alors l'ensemble de ces descriptions lexicales. Plus formellement, il s'agit de l'ensemble $F_{Id}(L_P)$. L'une des originalités de notre approche pour la reconnaissance de signes est alors que l'espace de représentation est totalement indépendant du système de mesure utilisé. Les ensembles lexicaux $L_{flexionPouce}$, $L_{orientPouce}$, $L_{flexionDoigt}$, $L_{écart}$ ont été choisis en fonction de l'aspect visuel de la main et non des données fournies par le Cyberglove. Au contraire, dans un système numérique « classique » de reconnaissance de signes, l'espace de représentation est généralement fortement dépendant du système de mesure utilisé. Lorsque le système de mesure est un gant numérique, l'espace de représentation est très souvent l'espace des données brutes ou d'une projection linéaire de cet espace. Dans le cas des systèmes de vision, il s'agit souvent de caractéristiques de l'image de la main, telles que son centre de gravité, son histogramme, l'aire de la surface de l'image correspondant à la main, etc.

L'étape de classification est donc ici elle-même indépendante du système de mesure utilisé. Dans le cas de la reconnaissance de signes statiques, elle correspond uniquement à la définition des signes à reconnaître comme des sous-ensembles de L_P , c'est-à-dire comme des classes de configurations particulières de la main. Elle ne nécessite aucun apprentissage dans le sens classique du terme : il n'est pas nécessaire d'enregistrer au préalable des signes statiques

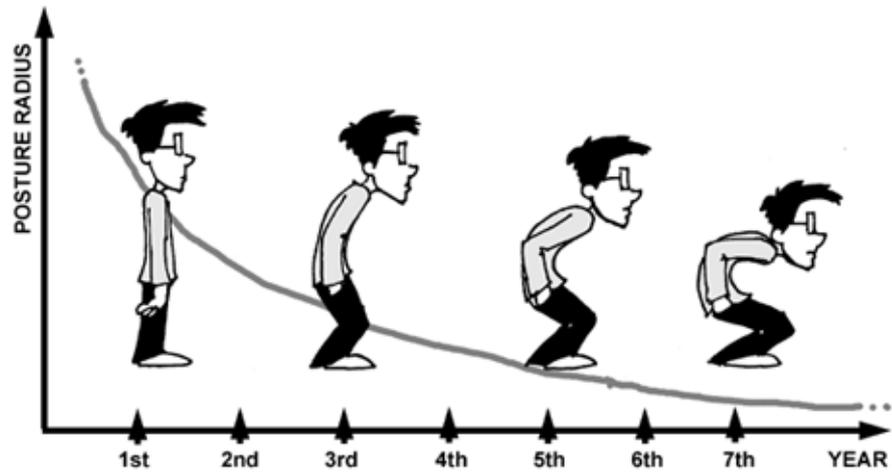
pour pouvoir les reconnaître. Cependant, le fait de devoir définir un signe S comme un sous-ensemble de L_P peut sembler fastidieux et il est parfois plus simple de donner un exemple de ce signe. On peut cependant imaginer extraire les définitions des signes à partir d'exemples fournis par un utilisateur. Cette possibilité pourra être étudiée dans des travaux ultérieurs. D'autre part, nous avons rencontré une limitation du formalisme des symbolisations pour la définition du signe M . Une autre perspective est donc d'étendre ce formalisme afin qu'il soit possible de définir correctement ce signe M .

Deux approches ont été envisagées pour la reconnaissance de signes dynamiques. L'une considérant les signes dynamiques uniquement comme des séquences de postures sans structure particulière, l'autre modélisant ces derniers comme l'association d'un signe statique de départ, d'un signe statique d'arrivée, d'une composante statique et d'une composante dynamique. La deuxième approche est la plus prometteuse. La comparaison des parties dynamiques d'un geste avec celles des différents signes à reconnaître n'est effectuée que lorsque les composantes statiques ne suffisent pas à la reconnaissance. Dans ce cas, nous avons choisi d'utiliser une normalisation temporelle pour nous affranchir de la variabilité temporelle des parties dynamiques. D'autres approches sont envisageables pour de futurs travaux. Il serait notamment intéressant d'étudier l'utilisation de modèles de Markov cachés qui ont donné de très bons résultats dans les systèmes purement numériques. Une extension de ces modèles de Markov, utilisable avec des données discrètes floues [Tsuboka 95], pourrait être directement utilisée.

THE
GRAD STUDENT
RADIUS
OF
POSTURE
CURVATURE

$$R = \frac{1}{N}$$

N = # YEARS IN PHD PROGRAM



CHAPITRE 6

*Contrôle d'un robot par le geste
de la main*

1. Introduction.

Dans ce chapitre, nous étudions les possibilités de commande d'un robot mobile par le geste de la main à partir du formalisme défini dans les chapitres précédents. La commande d'un robot par le geste fait généralement intervenir un ensemble d'intermédiaires : pour envoyer une instruction, il faut faire un geste ergotique qui actionne un intermédiaire envoyant à son tour un ordre au robot. L'intérêt d'utiliser le geste de la main dans sa fonction sémiotique comme instrument de contrôle est de supprimer tout intermédiaire, rendant la communication plus simple, plus fiable.

Contrairement à un système de télémanipulation, où le robot reproduit le geste de l'opérateur, un système de contrôle de robot par le geste comprend une étape d'interprétation du geste. L'approche généralement utilisée consiste à définir un ensemble de signes correspondant à un ensemble d'actions réalisables par le robot. Ces signes peuvent être statiques [Karlsson 98.b][Wachs 02] ou dynamiques [Lee 96][Waldherr 98][Iba 99][Ehrenmann 01], mesurés par un gant numérique [Lee 96][Karlsson 98.b][Iba 99][Wachs 02] ou par une système de vision [Waldherr 98][Ehrenmann 01]. Les systèmes de la littérature ne prévoient le plus souvent que la génération d'actions binaires : si un signe est reconnu, l'action correspondante est déclenchée, s'il ne l'est pas, rien ne se passe. Or il serait intéressant de pouvoir générer des commandes graduelles, permettant de contrôler l'amplitude de l'action correspondante.

Dans [Karlsson 98.b], un signe statique permet effectivement de passer dans un mode graduel, où la vitesse d'un robot dépend alors de la valeur d'un capteur particulier. Mais ce système est très limité puisqu'il ne permet de ne générer qu'une seule action graduelle. De plus, le fait de devoir faire un signe pour rentrer dans un certain mode, pour enfin pouvoir effectuer une action en faisant un autre signe, rend le système difficile à utiliser et peu réactif.

Dans la section suivante, nous présentons un système de contrôle d'un robot mobile basé sur l'utilisation de signes statiques, puis nous verrons comment générer des commandes graduelles pour le contrôle d'un robot pouvant effectuer des actions proportionnelles. La génération de commandes graduelles fera notamment intervenir la notion de défuzzification symbolique pour laquelle nous proposerons une nouvelle approche basée sur l'utilisation de la distance du transport.

2. Contrôle d'un robot mobile par des séquences de signes statiques

2.1. Les signes de commande

Dans cette section, nous présentons un système de contrôle d'un robot mobile par des signes statiques. Le robot commandé est le robot de LEGO Mindstorm (voir Fig. 111).



Fig. 111 Le robot de LEGO Mindstorm, commandé avec dix signes statiques.

Nous avons défini dix actions possibles pour ce robot. Chaque action A est associée à un signe statique S_A . Cinq actions lui permettent de se déplacer (*Avance*, *Recul*, *Gauche*, *Droite*, *Stop*), deux actions correspondent à des rotations rapides du robot sur lui-même, deux autres à l'émission de sons par le robot et enfin une action correspond à une mise en marche/arrêt du robot. Les dix signes statiques choisis pour ces actions sont représentés à la figure 112.

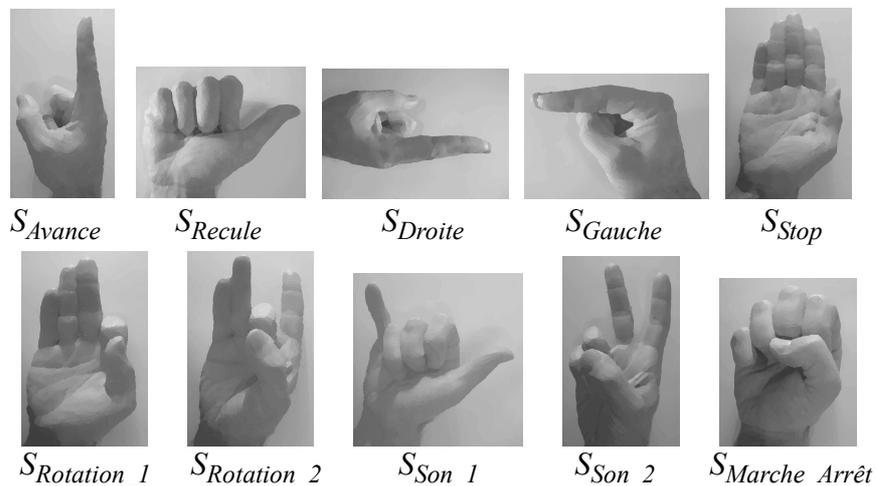


Fig. 112 Les dix signes statiques correspondant à des commandes du robot mobile.

Une configuration de la main très simple a été choisie pour le signe S_{Stop} . La commande *Stop*, doit en effet pouvoir, en cas d'urgence, être envoyée très rapidement, sans réfléchir et sans difficulté. L'action *Marche_arrêt* est nécessaire si l'on veut pouvoir utiliser la main droite pour autre chose que la commande du robot sans risquer d'envoyer des commandes involontaires.

Les signes S_{Gauche} et S_{Droite} sont très semblables. Ce choix permet de passer rapidement de l'un à l'autre et d'effectuer des mouvements de godilles.

Les ordres sont ponctuels : ils commandent le déclenchement d'une action qui s'effectue tant qu'un autre ordre ne déclenche pas une autre action. Ce principe de fonctionnement est illustré à la figure suivante donnant le schéma d'une trajectoire à faire suivre au robot et indiquant à quel moment doivent être effectués les différents signes. L'enregistrement de la trajectoire effectivement réalisée par le robot est également représenté.

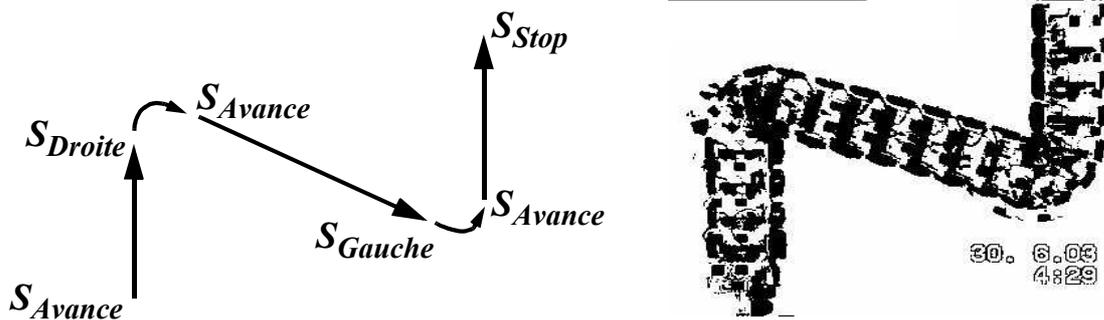


Fig. 113 Trajectoire à faire suivre au robot et trajectoire suivie.

Conformément à la formalisation des signes statiques introduite dans le chapitre précédent, chaque signe est défini comme une classe de configurations de la main, c'est-à-dire comme un sous-ensemble de l'univers lexical L_P .

2.2. Segmentation des gestes et reconnaissance

Contrairement à la reconnaissance de signes statiques présentée dans le chapitre précédent, où les postures étaient enregistrées, puis présentées une à une au système de reconnaissance pour leur identification, la reconnaissance des signes pour le contrôle du robot se fait en situation réelle. Le geste de l'opérateur est mesuré et analysé en continu et le système doit être capable de détecter un signe et de déclencher l'action correspondante en temps réel.

La chaîne de traitement des données, conduisant d'une posture p_0 , qui est la configuration de la main à un instant t_0 , à une action éventuelle A , est schématisée à la figure suivante. La posture p_0 est mesurée par le Cyberglove et transformée en un ensemble de dix-huit valeurs numériques. Quinze de ces valeurs sont converties par les neuf symbolisations introduites au chapitre 4, qui fournissent neuf descriptions floues, c'est-à-dire neuf sous-ensembles flous symboliques. Ces neuf descriptions sont ensuite utilisées pour le calcul des dix taux de reconnaissance des signes statiques de commande. On rappelle la formule donnant le taux de reconnaissance du signe S_{Avance} en fonction des descriptions d'une posture p :

$$rec_{S_{Avance}}(p) = \mu_{D_{orientPouce}(p)}(interne_dessus) \cdot \mu_{D_{flexionIndex}(p)}(tendu) \cdot \mu_{D_{flexionMajeur}(p)}(plié) \cdot \mu_{D_{flexionAnnulaire}(p)}(plié) \cdot \mu_{D_{flexionAuriculaire}(p)}(plié). \quad (152)$$

Chacun de ces taux de reconnaissance est alors comparé à un seuil de reconnaissance, qui est ici fixé à 0.1 . Si le taux de reconnaissance d'un signe est supérieur à ce seuil, le signe est déclaré comme reconnu.

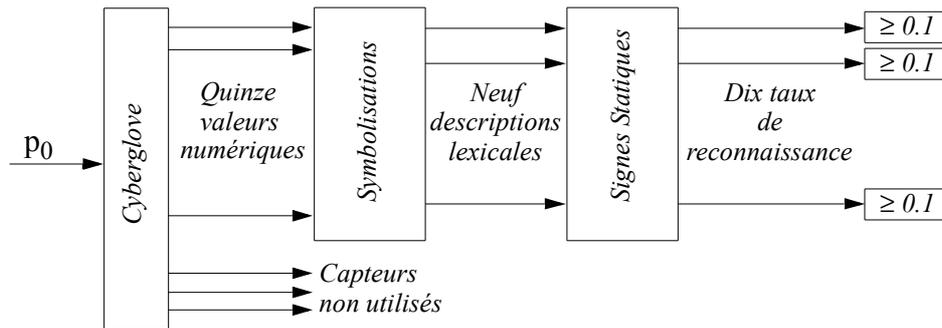


Fig. 114 Chaîne de traitement des données, d'une posture p_0 aux dix taux de reconnaissance correspondants.

Si un seul signe est reconnu, l'action correspondante est déclenchée. Si deux signes ou plus sont reconnus, l'action déclenchée est celle du signe ayant le plus fort taux de reconnaissance. Si aucun signe n'est reconnu, le robot continue l'action qu'il était en train d'effectuer.

On reprend l'exemple de trajectoire donnée à la figure 113. La commande correspondante ne fait appel qu'à cinq actions : *Avance*, *Droite*, *Gauche*, *Stop*, plus la commande *Marche_arrêt*. Sur la figure 115, nous avons représenté les cinq courbes correspondant aux taux de reconnaissance de ces cinq signes durant la période de commande. Cette figure donne également la superposition des cinq courbes, permettant de comparer leur niveaux. Sur chaque courbe, le seuil de reconnaissance est indiqué par une ligne horizontale. Les huit instants t_1, t_2, \dots, t_8 correspondant aux huit actions commandées au robot sont également représentés.

On peut remarquer qu'au début de l'enregistrement, le signe S_{Stop} a son taux de reconnaissance qui passe au-dessus du seuil de reconnaissance. Cependant, aucune action n'est déclenchée car le signe $S_{Marche_Arrêt}$ n'a pas encore été réalisé. Entre les instants t_3 et t_4 , les signes S_{Gauche} et S_{Droite} sont simultanément reconnus, ce qui arrive souvent puisque ces deux signes sont très proches l'un de l'autre. Ici, c'est le signe S_{Droite} qui a le taux de reconnaissance le plus élevé, c'est donc la commande *Droite* qui est transmise au robot.

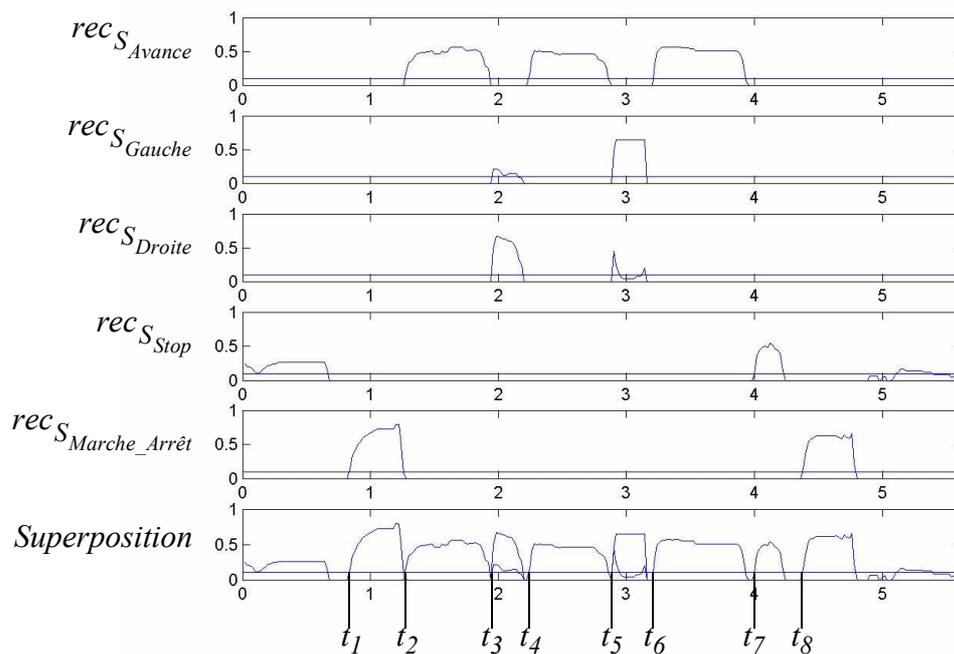


Fig. 115 Taux de reconnaissance de cinq signes de commandes durant le contrôle du robot.

Il est intéressant de remarquer que les instants où sont envoyées les commandes ne correspondent pas aux maxima des taux de reconnaissance, ils les anticipent. Cette anticipation permet de compenser les temps de traitement de la commande par le robot et d'avoir un système très réactif.

Au final, ce système de commande de robot est opérationnel. Il permet de contrôler facilement le robot et répond en temps réel aux ordres envoyés. Son principal défaut est que seules des actions binaires sont générées. Il faut donc fixer à l'avance la vitesse du robot et on ne peut pas trouver un compromis entre des manoeuvres précises nécessitant une vitesse faible et de longues trajectoires nécessitant une vitesse élevée. Le besoin d'utiliser des commandes graduelles apparaît rapidement. La génération de telles commandes est traitée dans la section suivante.

3. Génération de commandes graduelles

3.1. Principe

Le système de reconnaissance de signes statiques décrit dans les chapitres précédents manipule des données graduelles tout au long du processus de reconnaissance pour prendre, au final, une décision binaire : un signe est reconnu ou ne l'est pas. On perd donc, dans la dernière étape de la reconnaissance, l'étape de seuillage, une partie de l'information contenue dans la gradualité des descriptions floues. Il serait intéressant de pouvoir utiliser cette information afin de générer des commandes graduelles.

Nous considérons donc maintenant un robot pouvant effectuer des actions graduelles. Chaque action sera associée à une commande prenant ses valeurs dans l'intervalle $[0, 1]$, 0 correspondant à l'action nulle et 1 à l'action maximale.

3.1.1. Commande proportionnelle au taux de reconnaissance

Une solution simple qui peut être envisagée est d'associer chaque action A à un signe S_A , comme dans le cas des actions binaires, à la différence que l'amplitude c_A de la commande pour une posture p_0 serait maintenant égale au taux de reconnaissance du signe S_A :

$$c_A(p_0) = \mu_{M^*(S_A)}(p_0). \quad (153)$$

Cette solution très directe présente en réalité plusieurs inconvénients. Premièrement, comme dans le cas binaire, il faudrait appliquer un seuillage permettant d'éviter que des taux de reconnaissance trop faibles génèrent des actions. La commande proportionnelle deviendrait alors, avec un seuil de 0.1 :

$$c_A = \frac{\max(\text{rec}_{S_A} - 0.1, 0)}{0.9}. \quad (154)$$

De plus, si l'on observe les taux de reconnaissance représentés à la figure 115, on s'aperçoit que ceux-ci dépassent difficilement la valeur 0.5 . Cela s'explique par le fait que le taux de reconnaissance d'un signe est le produit de plusieurs valeurs comprises entre 0 et 1 (voir Equation (152)). Pour que l'amplitude maximale de l'action soit atteinte, il faudrait encore réduire la zone de la commande proportionnelle et donc la gradualité de cette commande, à des taux de reconnaissance compris entre 0.1 et 0.5 . Enfin, et c'est là le point le plus important, le taux de reconnaissance d'un signe dépendrait de la configuration de l'ensemble des doigts. Pour générer une commande graduelle, il faudrait soit réussir à modifier la configuration d'un doigt en gardant les autres doigts immobiles, soit coordonner l'ensemble de ses doigts pour obtenir la commande voulue. Une telle action serait en fait très difficilement contrôlable et cette solution

n'est donc pas envisagée. On lui préfère la commande graduelle contrôlée par la configuration d'un seul doigt.

3.1.2. Commande graduelle contrôlée par la configuration d'un doigt

Chaque action A est associée à un signe statique $S_A \subseteq L_P$ appelé signe d'activation de l'action A et à l'une des neuf symbolisations des configurations des doigts. Cette symbolisation notée $\langle P, L_A, R_A \rangle$, correspond au doigt contrôlant la commande de l'action A . Celle-ci ne peut être déclenchée que si le signe S_A est reconnu. La commande c_A dépend alors de la description D_A , de la posture p_0 en cours, dans la symbolisation $\langle P, L_A, R_A \rangle$. Finalement, le schéma général de la génération de ce type de commande graduelle est donné à la figure 116.

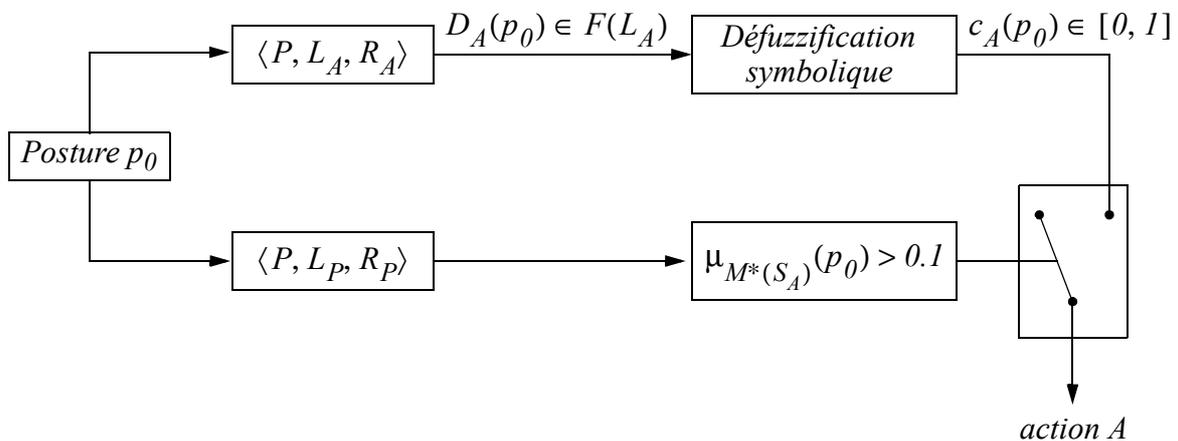


Fig. 116 Principe de l'action graduelle contrôlée par la configuration d'un seul doigt.

Prenons l'exemple de l'action *Avance*. Le signe d'activation S_{Avance} est le signe utilisé pour les commandes binaires dont on a enlevé la contrainte sur la configuration de l'index en spécifiant ce doigt comme devant être *quelconque* (voir Fig. 117). La symbolisation utilisée pour générer la commande c_{Avance} sera $\langle P, L_{Avance}, R_{Avance} \rangle = \langle P, L_{flexionDoigt}, R_{Index} \rangle$. Cette commande correspondra à une vitesse d'avance plus ou moins rapide du robot. On contrôlera donc la vitesse du robot par la seule flexion de l'index.

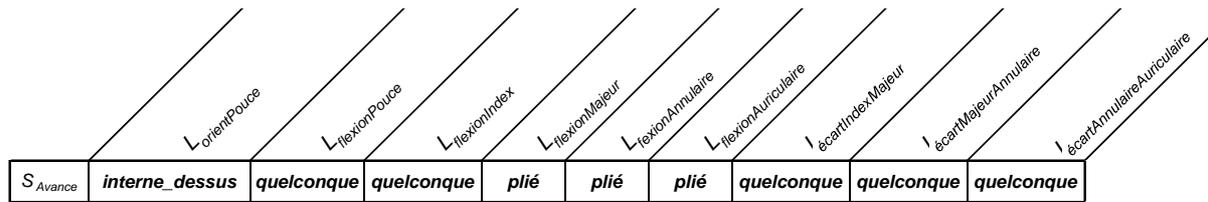


Fig. 117 Signe d'activation de l'action Avance.

Il reste à définir comment passer de la description $D_A(p_0)$, qui est un sous-ensemble flou de termes, à la commande $c_A \in [0, 1]$. Une telle opération n'est autre qu'une défuzzification symbolique, concept mis en évidence dans la typologie des systèmes flous [Foulloy 95]. Dans la section suivante, nous nous intéressons donc au problème de la défuzzification symbolique. Nous rappelons cette typologie ainsi que des approches pouvant être utilisées pour cette défuzzification. Nous modifions l'une de ces approches et introduisons une nouvelle méthode de défuzzification symbolique basée sur la distance du transport.

3.2. Défuzzification symbolique

3.2.1. Typologie des systèmes flous

Le fonctionnement interne des systèmes flous se décompose en trois étapes que sont la fuzzification, l'inférence et la défuzzification. La fuzzification transforme une entrée numérique en sous-ensembles flous d'entrée. A partir de ceux-ci, l'inférence détermine les sous-ensembles flous de sortie. Ces derniers sont ensuite convertis en sorties numériques par la défuzzification. Les différentes implantations possibles d'un système flou sont mises en évidence par la typologie des systèmes flous [Foulloy 95] présentée à la figure suivante.

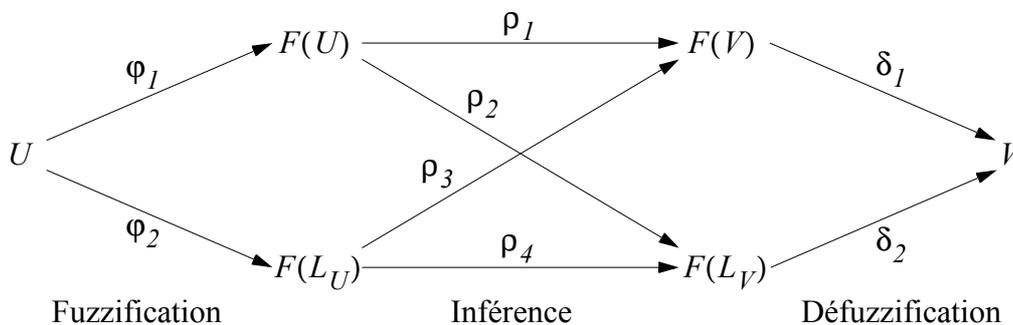


Fig. 118 Typologie des systèmes flous.

Cette classification différencie deux types de fuzzification, quatre types d'inférence et deux types de défuzzification. Il y a alors quatre implantations possibles pour un système flou, correspondant aux quatre types d'inférence :

$$\begin{aligned}
 SF1 : \quad v &= (\delta_1 \circ \rho_1 \circ \varphi_1)(u), \\
 SF2 : \quad v &= (\delta_2 \circ \rho_2 \circ \varphi_1)(u), \\
 SF3 : \quad v &= (\delta_1 \circ \rho_3 \circ \varphi_2)(u), \\
 SF4 : \quad v &= (\delta_2 \circ \rho_4 \circ \varphi_2)(u).
 \end{aligned} \tag{155}$$

Nous nous intéressons ici à la défuzzification δ_2 , appelée défuzzification symbolique.

3.2.2. Défuzzification symbolique

Ce type de défuzzification a été mis en évidence par la définition de la typologie et a été peu étudié dans la littérature. Il existe deux approches principales pour cette défuzzification.

a) Passage par une défuzzification numérique

Cette méthode consiste à se ramener à une défuzzification δ_1 , appelée défuzzification numérique, par une transformation δ_3 , permettant de passer de $F(L_V)$ à $F(V)$ (voir Fig. 119). Une telle méthode n'est donc pas à proprement parler une défuzzification symbolique. Elle nécessite, de plus, de faire deux choix, celui d'une transformation δ_3 et d'une méthode de défuzzification numérique, qui relèvent généralement de l'arbitraire.

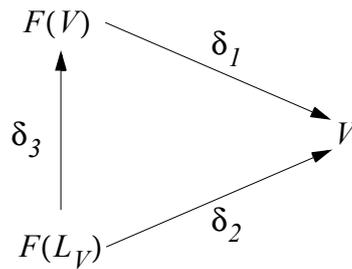


Fig. 119 Passage par une défuzzification numérique (δ_1).

Nous préférons ici utiliser une défuzzification purement symbolique [Galichet 01], comme la méthode des hauteurs.

b) Méthode des hauteurs

A chaque terme l de L_V est associée une valeur v_l de V . L'image d'un sous-ensemble flou F de L_V par δ_2 est alors :

$$\delta_2(F) = H(F) = \frac{\sum_{l \in L_V} v_l \cdot \mu_F(l)}{\sum_{l \in L_V} \mu_F(l)}. \quad (156)$$

Les sous-ensembles flous que nous défuzzifions sont des descriptions associées à des *Id*-symbolisations. Ce sont alors des éléments de $F_{Id}(L_V)$, c'est-à-dire qu'ils vérifient :

$$\sum_{l \in L_V} \mu_F(l) = 1. \quad (157)$$

La défuzzification d'une description F par la méthode des hauteurs s'écrit alors :

$$H(F) = \sum_{l \in L_V} v_l \cdot \mu_F(l). \quad (158)$$

Cette méthode des hauteurs permet de spécifier, pour chaque terme de L_V , la valeur de défuzzification correspondante. Mais il peut arriver qu'on ne veuille spécifier ces valeurs que sur une partie seulement des termes, soit parce que l'ensemble L_V est trop grand ou encore parce qu'on n'a pas d'idée précise sur les valeurs à associer à certains termes. Pour pouvoir prendre en compte cette situation, on peut imaginer de modifier la formule des hauteurs de la façon suivante :

$$H(F) = \frac{\sum_{l \in L'_V} v_l \cdot \mu_F(l)}{\sum_{l \in L'_V} \mu_F(l)}, \quad (159)$$

où L'_V est l'ensemble des termes dont on a spécifié les valeurs v_l . Mais ceci reviendrait à choisir $v_l = 0$ pour les termes n'appartenant pas à L'_V . Il serait plus intéressant que les valeurs $H(\{l\})$, pour ces termes $l \notin L'_V$, dépendent des valeurs associées aux termes de L'_V .

Pour cela, nous introduisons une nouvelle méthode de défuzzification symbolique, inspirée de la méthode des hauteurs, utilisant la distance du transport.

c) Défuzzification symbolique basée sur la distance du transport

La valeur défuzzifiée d'un sous-ensemble flou F de L_V selon la formule des hauteurs est une combinaison linéaire des valeurs v_l associées aux termes de L_V . On peut généraliser cette approche en considérant les défuzzifications symboliques effectuant une combinaison linéaire des valeurs v_l :

$$\delta_2(F) = \frac{\sum_{l \in L_V} v_l \cdot \gamma_l(F)}{\sum_{l \in L_V} \gamma_l(F)}. \quad (160)$$

Deux conditions sont toutefois imposées aux coefficients $\gamma_l(F)$ dans cette combinaison :

- Si F est lui-même un singleton $\{l_0\}$, le coefficient $\gamma_l(F)$ doit être nul pour tous les termes $l \neq l_0$. Cette propriété assure que la valeur défuzzifiée d'un singleton l_0 soit bien la valeur v_{l_0} spécifiée pour ce singleton.
- Si F est plus proche, selon une certaine métrique, d'un terme l_1 que d'un terme l_2 , le coefficient associé à l_1 sera plus important que celui associé à l_2 .

Ces deux conditions peuvent s'écrire plus formellement :

$$\forall l_0 \in L_V, \quad l \neq l_0 \Rightarrow \gamma_l(\{l_0\}) = 0, \quad (161)$$

$$\forall (l_1, l_2) \in L_V^2, \quad d(F, \{l_1\}) \leq d(F, \{l_2\}) \Rightarrow \gamma_{l_1}(F) \geq \gamma_{l_2}(F), \quad (162)$$

où d étant une distance définie sur $F(L_V)$. On dira qu'une telle défuzzification est basée sur la distance d .

Dans la méthode des hauteurs, on a $\gamma_l(F) = \mu_F(l)$. Si on considère uniquement des éléments de $F_{Id}(L_V)$, cette méthode vérifie les conditions (161) et (162), avec $d_{(\perp_L - T_M)}$ comme métrique sur $F_{Id}(L_V)$. C'est alors une défuzzification symbolique basée sur la distance $d_{(\perp_L - T_M)}$.

Supposons maintenant qu'une distance soit définie sur L_V . Notons d_{tp} la distance de transport associée sur $F_{Id}(L_V)$. On peut définir une défuzzification symbolique basée sur cette distance en choisissant des coefficients qui vérifient (161) et (162), d_{tp} étant utilisée pour cette deuxième condition. Dans la suite de ce chapitre, nous utiliserons les coefficients définis comme suit :

$$\gamma_{l_0}(F) = \prod_{l \in L'_V \setminus \{l_0\}} d_{tp}(F, \{l\}), \quad (163)$$

où L'_V est l'ensemble des termes de L_V pour lesquels une valeur v_l est spécifiée. On peut vérifier facilement que cette distance vérifie les conditions (161) et (162). Cette deuxième condition étant très proche de la propriété caractérisant les relations de ressemblance [De Cock 03] ou de proximité [Janis 03]. Chacun de ces coefficients peut être interprété comme un degré de ressemblance de F avec l_0 .

La défuzzification symbolique basée sur ces coefficients, qui sera notée H_{tp} , est finalement la suivante :

$$\delta_2(F) = H_{tp}(F) = \frac{\sum_{l \in L'_V} v_l \cdot \left(\prod_{l' \in L'_V \setminus \{l\}} d_{tp}(F, \{l'\}) \right)}{\sum_{l \in L'_V} \left(\prod_{l' \in L'_V \setminus \{l\}} d_{tp}(F, \{l'\}) \right)}. \quad (164)$$

Si $l_0 \in L_V \setminus L'_V$, ce qui signifie qu'aucune valeur n'est spécifiée pour l_0 , on a :

$$H_{tp}(\{l_0\}) = \frac{\sum_{l \in L'_V} v_l \cdot \frac{1}{d_{tp}(\{l_0\}, \{l\})}}{\sum_{l \in L'_V} \frac{1}{d_{tp}(\{l_0\}, \{l\})}}, \quad (165)$$

qui est bien définie car $d_{tp}(\{l_0\}, \{l\})$ ne peut pas être nul. La valeur défuzzifiée d'un terme l_0 n'appartenant pas à L'_V dépend donc cette fois des valeurs associées aux termes de L'_V . Cette dépendance est inversement proportionnelle à la distance de l_0 aux différents termes de L'_V .

Notons que le calcul de la distance du transport à un singleton a une formule analytique. Cette dernière peut donc, dans ce cas particulier, être calculée très rapidement :

$$\forall F \in F_{Id}(L_V), \forall l_0 \in L_V, \quad d_{tp}(F, \{l_0\}) = \sum_{l \in L_V \setminus \{l_0\}} \mu_F(l) \cdot d_{L_V}(l, l_0), \quad (166)$$

où d_{L_V} est la distance choisie sur L_V .

Illustrons le calcul de cette défuzzification symbolique sur un exemple. On considère l'ensemble lexical $L_{flexionDoigt}$. Nous utiliserons comme distance sur cet ensemble, la distance définie au chapitre 5, qui est le plus court chemin dans le graphe d'adjacence. Ce graphe d'adjacence est rappelé à la figure suivante.

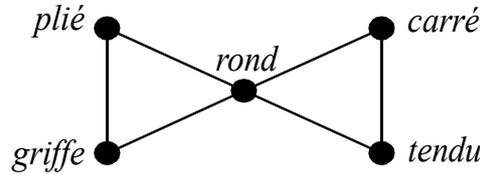


Fig. 120 Graphe d'adjacence de l'ensemble lexical $L_{flexionDoigt}$

On peut facilement retrouver les distances entre les termes à partir de ce graphe. On aura par exemple :

$$\begin{aligned} d_{L_{flexionDoigt}}(tendu, pli\acute{e}) &= 2, \\ d_{L_{flexionDoigt}}(tendu, carr\acute{e}) &= 1. \end{aligned} \quad (167)$$

Trois valeurs de l'intervalle $[0, 1]$ sont associées aux trois termes *plié*, *carré* et *tendu* :

$$v_{tendu} = 1, \quad v_{carr\acute{e}} = 0.5, \quad v_{pli\acute{e}} = 0. \quad (168)$$

On calcule alors la valeur défuzzifiée du sous-ensemble flou suivant :

$$F = 0.25/tendu + 0.25/carr\acute{e} + 0.5/rond. \quad (169)$$

Le détail du calcul est donné ci-dessous :

On peut également calculer la valeur défuzzifiée du singleton $\{rond\}$:

$$\begin{aligned}
 H_{tp}(F) &= \frac{1 \cdot (d_{tp}(F, \{\text{carré}\}) \cdot d_{tp}(F, \{\text{plié}\})) + 0.5 \cdot (d_{tp}(F, \{\text{tendu}\}) \cdot d_{tp}(F, \{\text{plié}\})) + 0}{d_{tp}(F, \{\text{carré}\}) \cdot d_{tp}(F, \{\text{plié}\}) + d_{tp}(F, \{\text{tendu}\}) \cdot d_{tp}(F, \{\text{plié}\}) + d_{tp}(F, \{\text{tendu}\}) \cdot d_{tp}(F, \{\text{carré}\})} \\
 &= \frac{1 \cdot (0.75 \cdot 1.5) + 0.5 \cdot (0.75 \cdot 1.5)}{0.75 \cdot 1.5 + 0.75 \cdot 1.5 + 0.75 \cdot 0.75} = 0.6.
 \end{aligned}$$

$$H_{tp}(\{\text{rond}\}) = \frac{1 \cdot (1 \cdot 1) + 0.5 \cdot (1 \cdot 1)}{1 \cdot 1 + 1 \cdot 1 + 1 \cdot 1} = 0.5. \quad (170)$$

La défuzzification symbolique H_{tp} n'est pas une extension directe de la méthode des hauteurs. Pour que ce soit le cas, il faudrait qu'en choisissant la distance discrète comme métrique sur L_V et en choisissant L'_V égal à L_V , on retrouve la formule des hauteurs. Rappelons en effet que la distance du transport associée à la distance discrète n'est autre que $d_{(\perp_L - T_M)}$. Mais si on utilise $d_{(\perp_L - T_M)}$ dans (164), on obtient :

$$H_{tp}(F) = \frac{\sum_{l \in L'_V} v_l \cdot \left(\prod_{l' \in L'_V \setminus \{l\}} (1 - \mu_F(l')) \right)}{\sum_{l \in L'_V} \left(\prod_{l' \in L'_V \setminus \{l\}} (1 - \mu_F(l')) \right)}, \quad (171)$$

et (171) n'est pas égale à (156). Prenons par exemple $V = [0, 1]$ et $L_V = \{A, B, C\}$, avec $v_A = 0$, $v_B = 0.5$ et $v_C = 1$. On considère le sous-ensemble flou $F = 0.5/A + 0.5/B$. Ce sous-ensemble flou appartient bien à $F_{Id}(L_V)$. Sa valeur défuzzifiée avec la méthode des hauteurs est :

$$H(F) = 0.5 \cdot 0 + 0.5 \cdot 0.5 + 0 \cdot 1 = 0.25. \quad (172)$$

Le même sous-ensemble flou défuzzifié avec l'opérateur H_{tp} associé à la distance discrète donnera :

$$H_{tp}(F) = \frac{\frac{1}{0.5} \cdot 0 + \frac{1}{0.5} \cdot 0.5 + \frac{1}{1} \cdot 1}{\frac{1}{0.5} + \frac{1}{0.5} + \frac{1}{1}} = 0.4. \quad (173)$$

Nous allons maintenant étudier l'application des ces différents opérateurs de défuzzification symbolique à la génération de commandes graduelles.

3.3. Génération d'actions graduelles

3.3.1. Utilisation de la méthode des hauteurs

Soit une action A et $\langle P, L_A, R_A \rangle$ la symbolisation correspondant au doigt contrôlant la commande de cette action. Pour tout terme l de L_A , on définit une commande $c_l \in [0, 1]$ de l'action A correspondante. Soit alors une posture p_0 . La commande correspondant à cette posture sera finalement égale à :

$$c_A(p_0) = \sum_{l \in L_A} c_l \cdot \mu_{D_A(p_0)}(l). \quad (174)$$

Illustrons ce mécanisme sur l'exemple du signe *Avance*. Nous avons choisi $\langle P, L_{Avance}, R_{Avance} \rangle = \langle P, L_{flexionDoigt}, R_{flexionIndex} \rangle$. Il faut donc attribuer des amplitudes d'action à chacun des termes de $L_{flexionDoigt}$. Dans un premier temps, nous choisissons la distribution suivante :

$$\begin{aligned} c_{tendu} &= 1, \\ c_{carré} &= c_{rond} = c_{griffe} = c_{plié} = 0. \end{aligned} \quad (175)$$

Sur la figure 121, deux séries de postures sont représentées. La série 121.a montre l'effet de l'ouverture de l'index sur l'action *Avance*, la figure 121.b montre l'effet de l'ouverture du pouce sur cette même action. On donne, pour chaque posture, la valeur de la commande c_{Avance} ainsi que le taux de reconnaissance du signe d'activation S_{Avance} .

On peut remarquer que c'est effectivement la flexion de l'index seule qui a une influence sur la commande de l'action *Avance*. L'affaiblissement de cette commande de la posture p_4 à la posture p_5 , est due au fait que l'index change de configuration entre les deux postures, il est en effet difficile de bouger le pouce tout en gardant l'index totalement immobile. Lorsque le pouce est dans une configuration trop différente de *interne_dessus*, le signe S_{Avance} n'est plus reconnu et l'action n'est pas déclenchée.

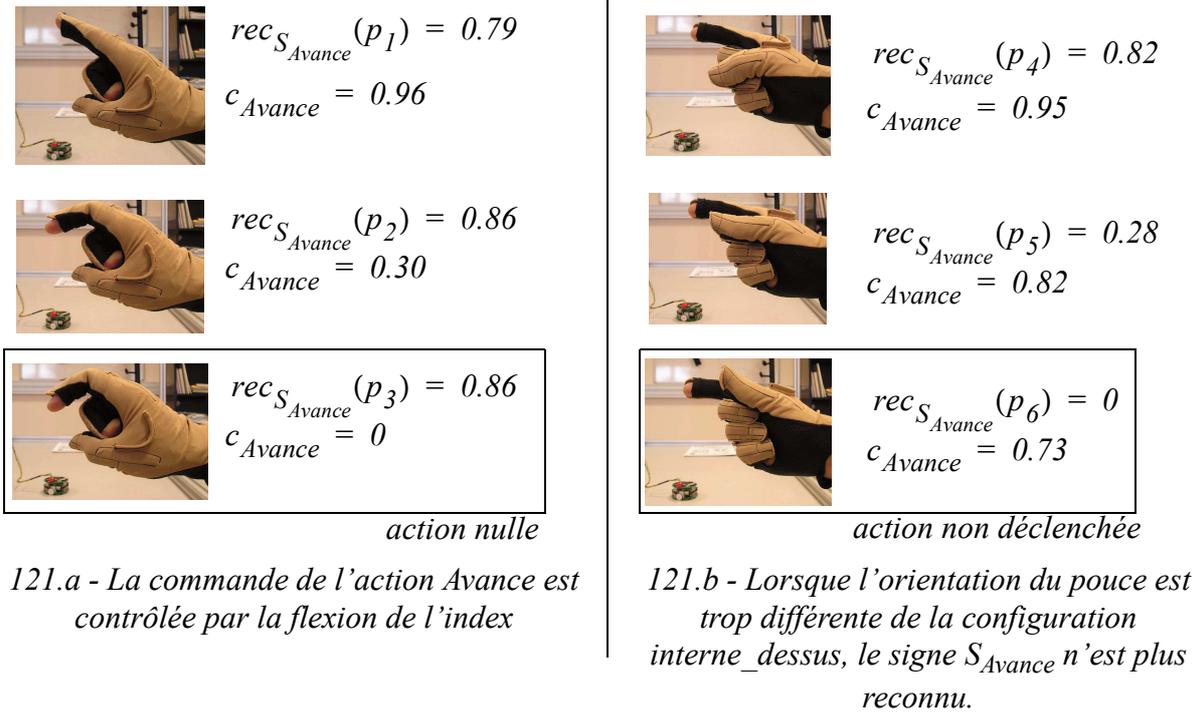


Fig. 121 Commande de l'action *Avance* pour six postures.

La distribution des commandes des différents termes que nous avons choisis (175) concentre le contrôle de l'action *Avance* sur la fin de l'extension de l'index ce qui ne permet pas d'être précis et devient rapidement fatigant pour l'utilisateur. Sur la figure 121.a, on peut effectivement remarquer que les postures p_2 et p_3 , pourtant très proches l'une de l'autre, génèrent des amplitudes très différentes. On peut également illustrer ce point en donnant, pour chaque couple (x_{IP_2}, x_{MCP_2}) de valeurs des capteurs *IP* et *MCP* de l'index, la commande $c_{Avance}(x_{IP_2}, x_{MCP_2})$ correspondante (voir Fig. 122). Avec la distribution (175), on peut montrer qu'on a :

$$c_{Avance}(x_{IP_2}, x_{MCP_2}) = \mu_{M_{flexionIndex}(tendu)}(x_{IP_2}, x_{MCP_2}). \quad (176)$$

Cette commande est nulle sur une grande partie du domaine de flexion de l'index. Elle est égale à 1 sur le noyau de la signification du terme *tendu*. La zone de commande graduelle est donc réduite et présente des pentes importantes.

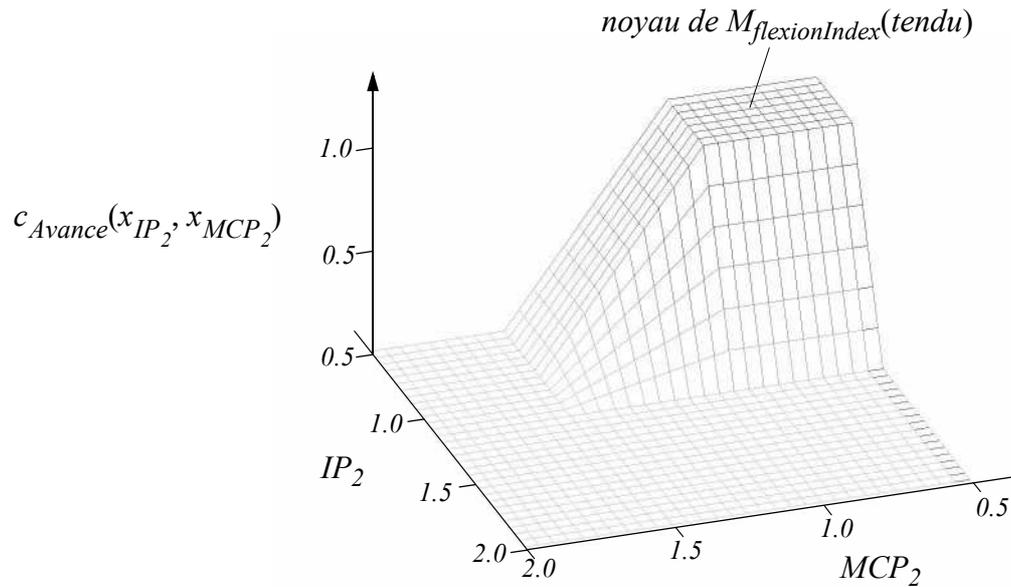


Fig. 122 Valeurs de $c_{Avance}(x_{IP_2}, x_{MCP_2})$ obtenues avec la méthode des hauteurs et la distribution (175), pour x_{IP_2} et x_{MCP_2} appartenant à $[0.5, 2.0]$.

Pour avoir une commande qui soit plus graduelle et plus confortable, il faut que son contrôle soit réparti sur tout le domaine de flexion de l'index et pas seulement sur la partie *tendu* de ce domaine. On peut pour cela choisir la distribution suivante pour les commandes associées au termes de $L_{flexionTendu}$:

$$\begin{aligned}
 c_{tendu} &= 1, \\
 c_{carré} &= c_{rond} = c_{griffe} = 0.5, \\
 c_{plié} &= 0.
 \end{aligned}
 \tag{177}$$

Alors, si on reprend les postures de la figure 121.a, on a les commandes suivantes.

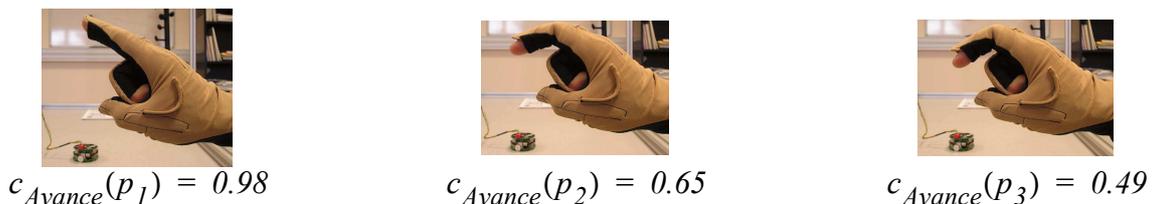


Fig. 123 Amplitude de la commande c_{Avance} pour les trois postures de la figure 121.a avec la nouvelle distribution de commandes (177).

L'amplitude de l'action *Avance* décroît maintenant beaucoup plus doucement et sera plus facile et plus confortable à manipuler.

Si on trace la commande $c_{Avance}(x_{IP_2}, x_{MCP_2})$ obtenue pour toutes les valeurs possibles des capteurs IP_2 et MCP_2 , on voit clairement que la commande est répartie sur tout le domaine de flexion de l'index. Les « plateaux » de la commande correspondent aux noyaux des significations des différents termes de $L_{flexionIndex}$.

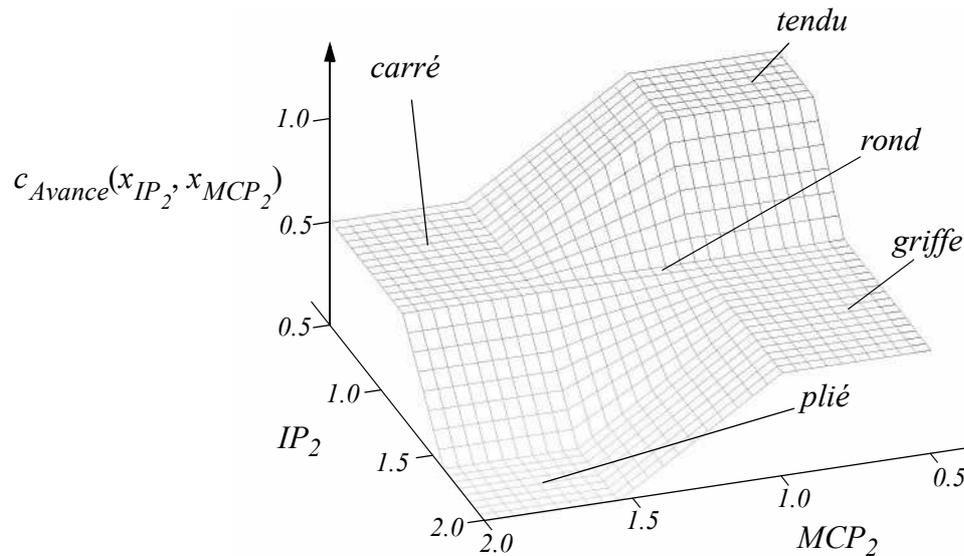


Fig. 124 Valeurs de $c_{Avance}(x_{IP_2}, x_{MCP_2})$ obtenues avec la méthode des hauteurs et la distribution (177), pour x_{IP_2} et x_{MCP_2} appartenant à $[0.5, 2.0]$.

3.3.2. Utilisation de la distance du transport

Nous pouvons également définir l'action graduelle en utilisant la défuzzification symbolique H_{tp} basée sur la distance du transport. Nous utiliserons comme distance sur $L_{flexionDoigt}$, la distance basée sur le graphe d'adjacence rappelé dans la section 3.2.2.c (Fig. 120).

La défuzzification H_{tp} permet de ne spécifier que les commandes correspondant aux configurations *tendu* et *plié*. On choisit les commandes suivantes :

$$\begin{aligned} c_{tendu} &= 1, \\ c_{plié} &= 0. \end{aligned} \tag{178}$$

Nous reprenons maintenant les trois postures de la figure 121.a. Les amplitudes de commande pour ces trois postures sont les suivantes :

$$\begin{aligned} c_{Avance}(p_1) &= 0.98, \\ c_{Avance}(p_2) &= 0.65, \\ c_{Avance}(p_3) &= 0.40. \end{aligned} \quad (179)$$

Ces valeurs confirment bien le fait qu'en ne spécifiant que les valeurs associées aux termes *tendu* et *plié*, la commande se répartie sur l'ensemble du domaine de flexion de l'index, cette répartition se fait selon la métrique choisie sur $L_{flexionIndex}$. On trace une nouvelle fois la valeur de la commande pour les différentes valeurs possibles des capteurs de flexion IP_2 et MCP_2 .

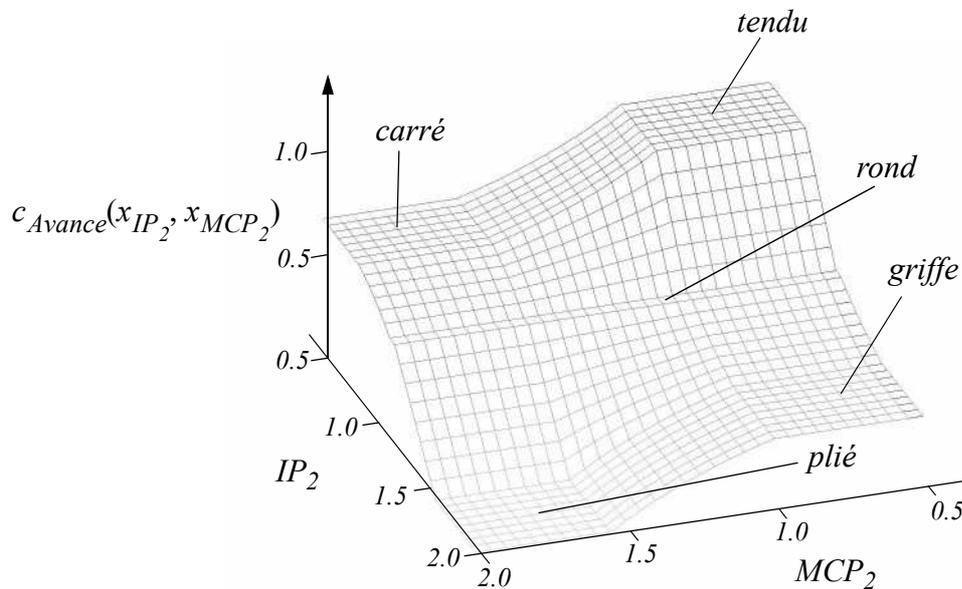


Fig. 125 Valeurs de $c_{Avance}(x_{IP_2}, x_{MCP_2})$ obtenues avec la défuzzification H_{tp} basée sur la distance du transport et la distribution (178), pour x_{IP_2} et x_{MCP_2} appartenant à $[0.5, 2.0]$.

On obtient donc une commande répartie sur tout le domaine de flexion de l'index. Les commandes correspondant aux termes autres que *tendu* et *carré* sont des combinaisons linéaires de c_{tendu} et $c_{carré}$. On peut remarquer que les pentes de la commandes ne sont pas linéaires, mais ces non-linéarités restent faibles et ne sont pas gênantes pour la commande du robot.

3.4. Contrôle d'un robot par des actions graduelles

3.4.1. Principe

Nous appliquons cette méthode de génération de commandes graduelles au contrôle du robot Khepera de K-Team [www.k-team.com] (voir Fig. 126). Ce robot possède deux roues motrices, chacune ayant un moteur indépendant. La vitesse de rotation de chaque roue peut être ajustée de manière continue. Ce robot est donc capable d'effectuer des actions graduelles. L'équilibre du robot est assuré par deux plots glissant sur le sol, disposés perpendiculairement aux roues.

Le contrôle se fera par deux commandes graduelles, appelées *Moteur_Gauche* et *Moteur_Droit* et cinq commandes binaires : *Recule*, *Stop*, *Gauche*, *Droite* et *Marche_Arrêt*. L'action *Moteur_Gauche* commande la rotation du moteur gauche et l'action *Moteur_Droit* commande celle du moteur droit. Les cinq actions binaires ont la même signification que dans la première partie de ce chapitre. Les deux actions graduelles ont le même signe d'activation : $S_{Moteur_Gauche} = S_{Moteur_Droit} = S_{Avance}$, S_{Avance} étant toujours le signe défini à la figure 117. Elles seront donc toujours activées en même temps. Elles seront de plus contrôlées par le même doigt, l'index, qui pourra, en contrôlant simultanément la rotation des deux moteurs, faire avancer et tourner le robot.

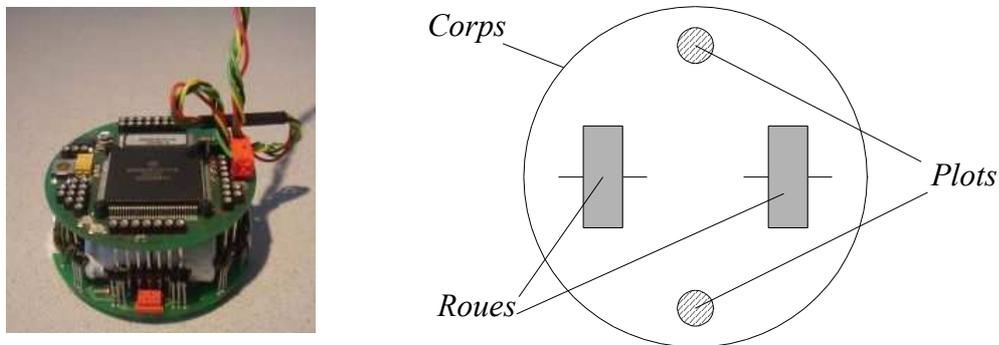


Fig. 126 Le Robot Khepera, photo et schéma de la vue de dessous.

Les signes associés aux action binaires *Recule*, *Stop*, *Droite* et *Gauche* sont les mêmes que ceux utilisés dans la première partie de ce chapitre. Par contre le signe $S_{Marche_Arrêt}$ doit être modifié car celui utilisé dans la première partie est maintenant inclus dans le signe S_{Avance} . Nous utiliserons alors le signe suivant :

 $S_{Marche_Arrêt}$ Fig. 127 Le nouveau signe $S_{Marche_Arrêt}$

Les six signes statiques que l'on cherche à reconnaître sont donc S_{Avance} , S_{Recul} , S_{Stop} , S_{Gauche} , S_{Droite} et $S_{Marche_Arrêt}$. Le système de reconnaissance et de sélection de l'action devant être réalisée est le même que dans le cas binaire. Un signe est reconnu si son taux de reconnaissance est supérieur à un seuil, toujours fixé à 0.1 . Parmi les signes reconnus, on sélectionne ensuite celui ayant le plus fort taux de reconnaissance. Si c'est le signe S_{Avance} qui est sélectionné, les deux actions $Moteur_Gauche$ et $Moteur_Droit$ sont activées simultanément et contrôlées par l'index.

3.4.2. Génération de l'action graduelle

La défuzzification symbolique H_{tp} est utilisée. La distance définie sur les symboles est celle issue du graphe d'adjacence (voir Fig. 120). Il nous reste donc à définir les commandes associées aux termes de $L_{flexionDoigt}$. Nous noterons c_{Gauche} et c_{Droit} les commandes des deux moteurs. Plusieurs distributions de commandes ont été réalisées et testées. Elles conduisent à des comportements différents du robot et le choix de l'une par rapport à l'autre dépendra de la tâche devant être réalisée par le robot. La distribution de commandes que nous utilisons ici est la suivante :

$$\begin{aligned}
 c_{Gauche_{tendu}} &= 1, & c_{Droit_{tendu}} &= 0.5, \\
 c_{Gauche_{carré}} &= 0.5, & c_{Droit_{carré}} &= 1, \\
 c_{Gauche_{plié}} &= 0, & c_{Droit_{plié}} &= 0.
 \end{aligned} \tag{180}$$

Sur la figure suivante, on donne la valeurs des commandes c_{Gauche} et c_{Droit} pour différentes postures.

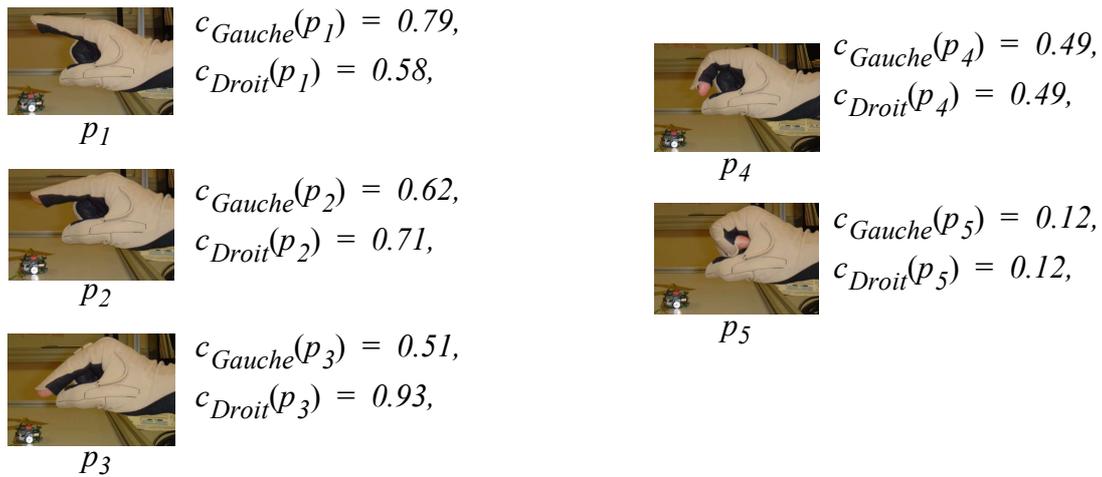


Fig. 128 Commandes du moteur gauche et du moteur droit pour cinq postures.

Il est possible de donner, à partir des vitesses d'avance des deux roues, la vitesse d'avance du robot ainsi que sa vitesse de rotation. Si on note v_{Gauche} et v_{Droite} les vitesses d'avance de chaque roue, on a :

$$\begin{aligned}
 v_{Avance} &= \frac{v_{Gauche} + v_{Droite}}{2}, \\
 v_{Rotation} &= \frac{v_{Gauche} - v_{Droite}}{GD},
 \end{aligned} \tag{181}$$

où v_{Avance} est la vitesse d'avance du robot, $v_{Rotation}$ sa vitesse de rotation donnée dans le sens horaire, le robot étant vu de dessus et GD est la distance entre les deux roues. Les vitesses d'avance des deux roues étant directement proportionnelles aux commandes appliquées aux moteurs, on peut alors donner, en valeurs normalisées, la vitesse d'avance et la vitesse de rotation du robot connaissant les commandes des deux moteurs. Nous donnons alors les vitesses d'avance et de rotation normalisées pour les cinq postures de la figure 128.

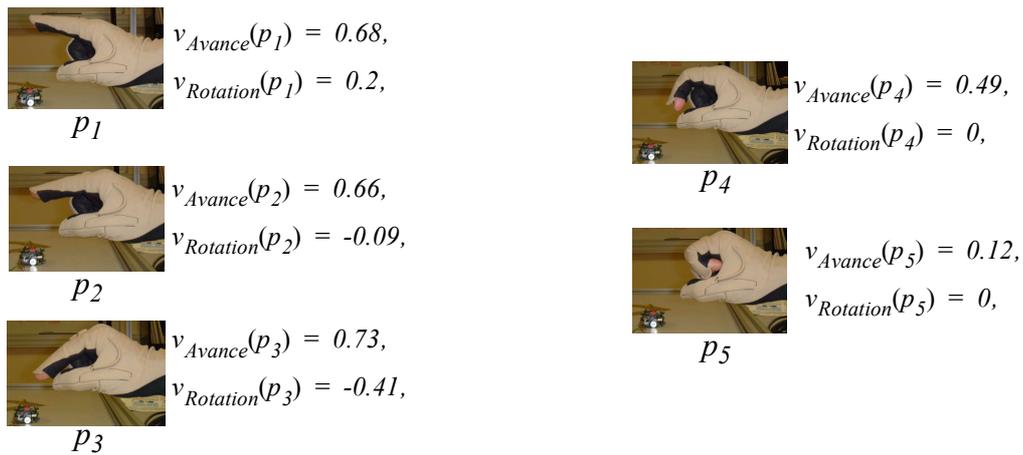


Fig. 129 Amplitude de l'avance et de la rotation pour les cinq postures de la figure 128.

Lorsque l'index sera dans la configuration *tendu*, le robot décrira donc une courbe vers la droite et lorsqu'il sera dans la configuration *carré*, le robot décrira une courbe vers la gauche. Lorsque l'index sera dans la configuration *rond*, le robot avancera droit devant lui, avec une vitesse d'avance réduite.

On peut également donner les vitesses d'avance et de rotation pour les différentes valeurs de flexion de l'index, c'est-à-dire pour les différentes valeurs des capteurs de flexion IP_2 et MCP_2 . C'est ce que l'on fait à la figure suivante.

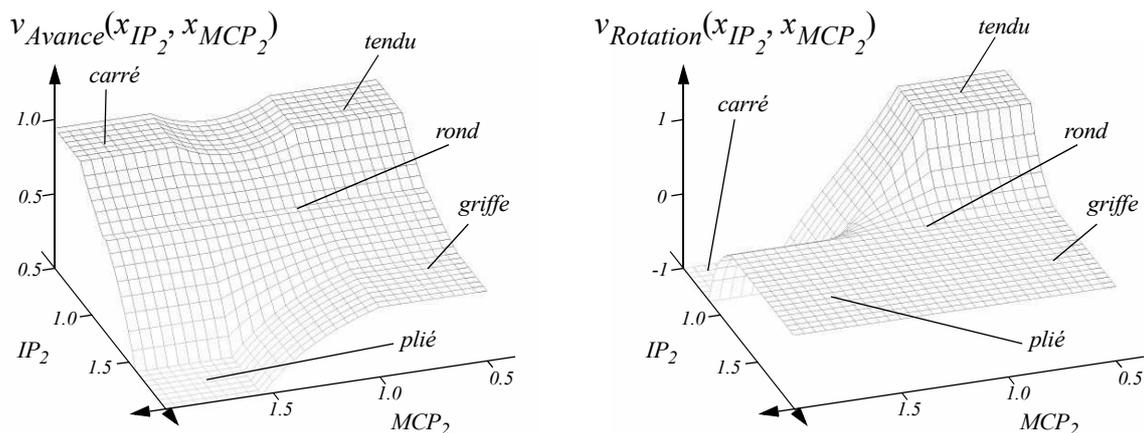


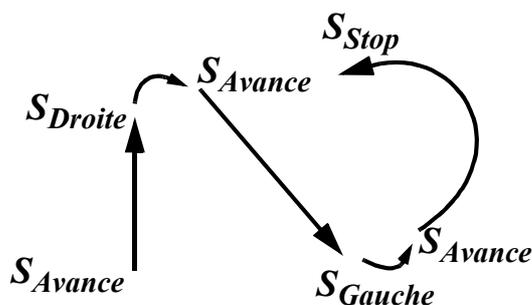
Fig. 130 Avance et rotation du robot pour les différentes valeurs de capteurs de flexion IP_2 et MCP_2 .

Ces courbes confirment le fait que le robot tourne à droite lorsque l'index est *tendu* et à gauche lorsque l'index est *carré*. En faisant varier l'index entre ces deux configurations, on peut

ajuster la rotation du robot. Lorsque l'index est dans la configuration *rond* ou *griffe*, le robot avance tout droit et enfin, lorsqu'il est *plié*, le robot est immobile. Dès que le signe S_{Avance} n'est plus reconnu, le robot continue d'avancer à allure constante jusqu'à ce qu'un autre ordre lui soit envoyé.

3.4.3. Exemple de trajectoire

Cette façon de commander le robot est illustrée à la figure suivante où une trajectoire à faire suivre au robot est représentée. Cette trajectoire comporte une courbe qu'il est impossible de suivre avec des actions binaires. Mais l'action graduelle que nous avons défini permet effectivement de suivre une telle courbe. Les différents signes statiques à effectuer pour cette trajectoire sont également représentés sur ce schéma. La figure 131.b montre que le robot effectue la trajectoire voulue.



131.a - Trajectoire à faire suivre au robot et signes à effectuer pour réaliser cette tâche.



131.b - Trajectoire suivie par le robot.

Fig. 131 Contrôle du robot Khepera par des actions binaires et graduelles.

Les taux de reconnaissance des signes S_{Avance} , S_{Recul} , S_{Stop} , S_{Gauche} et S_{Droite} durant la période de commande pour le suivi de cette trajectoire sont donnés à la figure 132. Sur cette figure sont également représentées les vitesses d'avance et de rotation normalisées, qui sont commandées par l'index lorsque le signe S_{Avance} est reconnu. Comme dans le cas des actions binaires, le robot continue d'effectuer une action donnée tant qu'un autre signe n'est pas reconnu. Six signes doivent être effectués pour réaliser la trajectoire. Les six instants correspondant à la reconnaissance de ces six signes et au déclenchement de l'action correspondante, sont représentés à la figure 132. Entre l'instant t_1 et l'instant t_2 , ainsi qu'entre les instants t_3 et t_4 , le robot doit suivre une trajectoire rectiligne. La vitesse de rotation est alors très faible et oscille autour de zéro pour récupérer des erreurs de trajectoire. Entre les instants t_2 et t_3 , le robot doit effectuer une rotation à droite sur lui-même. Le signe S_{Droite} est alors utilisé. On ne peut en effet effectuer une rotation sans vitesse d'avance avec l'action graduelle *Avance*.

On retrouve cette situation entre les instants t_4 et t_5 . Enfin, entre les instants t_5 et t_6 , le robot effectue la courbe finale. L'action graduelle *Avance* est utilisée. Cette fois-ci la vitesse de rotation est bien négative ce qui se traduit par une rotation à gauche du robot.

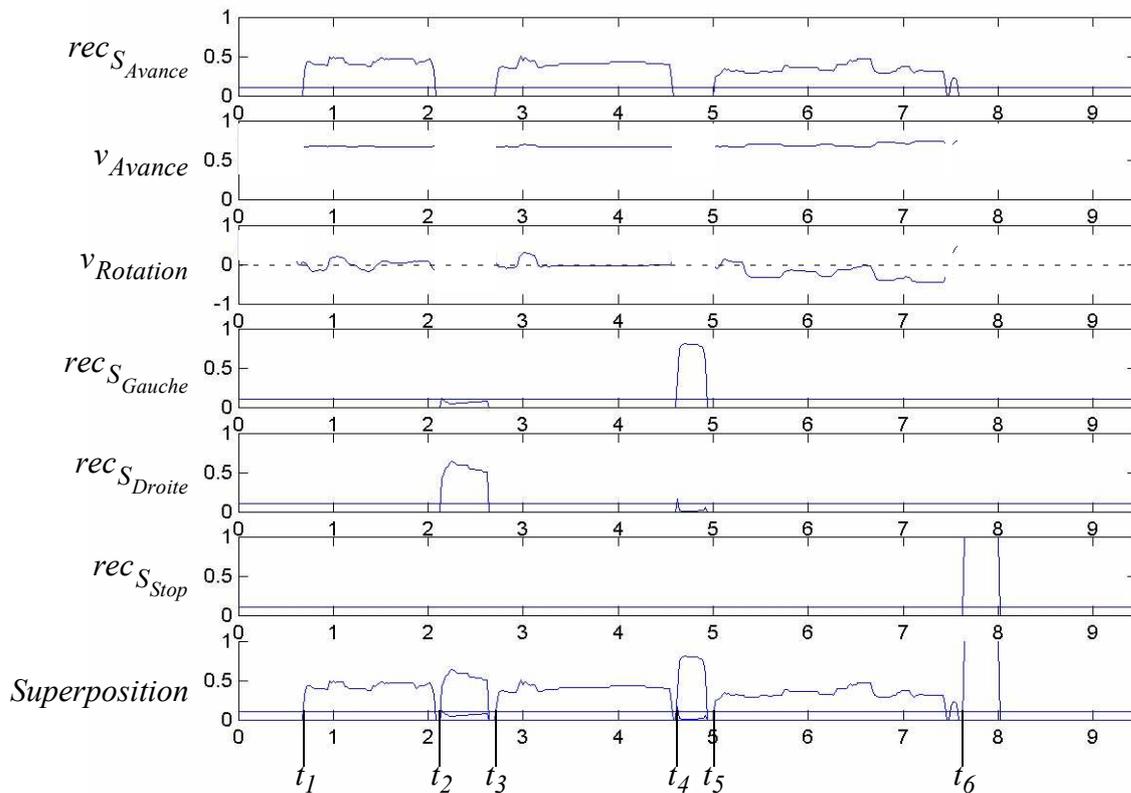


Fig. 132 Taux de reconnaissance des quatre signes statiques et amplitudes des actions graduelles durant le contrôle du robot.

Ce type d'action graduelle permet donc de commander le robot correctement. Il est en revanche plus difficile de maintenir le robot dans une trajectoire rectiligne que ce n'était le cas avec des commandes binaires. Dans de futurs travaux, il faudra en outre s'interroger sur le fait que les cinq termes choisis pour la description de l'index, l'ont été afin de pouvoir représenter correctement l'aspect visuel de la main et ne sont pas forcément les mieux adaptés à la génération de commandes pour un robot. D'autre part, bien d'autres stratégies de contrôle du robot pourraient être envisagées. Il serait, entre autre, intéressant de pouvoir faire varier les amplitudes maximales des commandes c_{Gauche} et c_{Droit} pendant le contrôle du robot, par exemple en utilisant le pouce pour le contrôle d'une troisième commande graduelle.

4. Conclusion

Le formalisme des symbolisations floues et son utilisation pour la description de la configuration de la main nous a permis de reconnaître des signes statiques en temps réel. Cette capacité a pu être utilisée pour la commande d'un robot par un ensemble de dix signes statiques. Mais la génération de commandes uniquement binaires est très limitée et la gradualité des représentations lexicales utilisées a permis la génération de commandes graduelles. Celle-ci a nécessité la mise en oeuvre d'une défuzzification symbolique. La méthode des hauteurs, généralement utilisée pour ce problème, a été étendue à une nouvelle méthode de défuzzification symbolique basée sur la distance du transport. Le problème que pose généralement cette distance est celui de son calcul nécessitant la résolution d'un problème de programmation linéaire. Mais dans ce cas précis, le calcul de la distance du transport fait toujours intervenir un singleton et on dispose dans ce cas d'une formule analytique. Cette nouvelle défuzzification a pu être utilisée avec succès pour la commande graduelle et il serait maintenant intéressant d'envisager son application dans d'autres systèmes utilisant un contrôleur flou.



JORGE CHAM © 2004

www.phdcomics.com

Conclusion

Dans ce travail, nous avons construit une description symbolique de la configuration de la main et défini des méthodes de traitement de cette information symbolique qui nous ont permis de reconnaître des signes statiques et dynamiques ainsi que de contrôler un robot mobile avec le geste de la main. Les contributions apportées dans ce travail sont de plusieurs ordres.

Une première contribution d'ordre bibliographique est la revue des systèmes intrusifs de mesure du geste manuel dans le chapitre 1. Les principales caractéristiques de ce geste ont été rappelées dans ce même chapitre. Nous nous sommes concentrés sur les systèmes de localisation de la main dans l'espace et de mesure de sa configuration. Nous avons également mentionné les principes de mesure du geste par les systèmes de vision, sans toutefois présenter les différents systèmes existant. Cette revue nous a permis de souligner la diversité des technologies utilisées pour cette mesure, plus particulièrement de celles mises en oeuvre dans les gants numériques. Elle a également permis de situer le Cyberglove que nous avons utilisé par rapport aux autres technologies existantes.

L'autre apport d'ordre bibliographique se trouve dans le chapitre 2 où nous avons référencé et détaillé les principales méthodes envisagées dans la littérature pour la reconnaissance de signes statiques ainsi que de signes dynamiques. Nous nous sommes particulièrement attachés à la description des méthodes de classification. Nous avons, là encore, pu souligner la diversité de ces méthodes que nous avons tenté de répertorier selon les approches utilisées. Nous avons fourni les performances des différents systèmes de

reconnaissance, en soulignant toutefois la difficulté qui se pose lorsque l'on veut comparer ces performances. En effet, la variété des gestes étudiés et des systèmes de mesure utilisés contribue au manque de données de référence telles qu'on en trouve en général pour d'autres problèmes de classification.

Des contributions d'ordre théorique ont été apportées principalement au chapitre 3, dans lequel nous avons tout d'abord fait une présentation détaillée du formalisme des symbolisations et plus particulièrement des ϕ -symbolisations que nous avons introduites. Ce formalisme rassemble dans un cadre cohérent différents travaux portant sur la conversion numérique-symbolique. Trois processus complémentaires de construction de ces symbolisations ont été proposés et les bases de règles ont été introduites comme une représentation particulière d'un de ces trois processus. Dans ce chapitre 3, on trouve également une étude détaillée d'une famille particulière de relations, les relations $(\perp - T^*)$. Après un bref rappel sur les relations d'équivalences floues et leur rapport à certaines métriques généralisées, nous avons présenté plusieurs résultats concernant ces relations particulières. Nous avons enfin introduit un nouvel opérateur de distance entre descriptions floues issues d'une *Id*-symbolisation. Cet opérateur, appelé distance du transport, permet de généraliser aux descriptions floues une métrique définie sur les symboles. Elle est une extension directe de la distance associée à la relation d'égalité floue $(\perp_L - T_M)$ puisque ces deux distances sont égales lorsque la métrique définie sur les symboles est la métrique discrète.

On trouve une autre contribution d'ordre théorique dans le chapitre 6, concernant le problème de la défuzzification symbolique qui est l'une des deux défuzzifications mises en évidence par la typologie des contrôleurs flous. La méthode des hauteurs est généralement utilisée pour cette défuzzification, mais nous avons vu que cette méthode présente plusieurs limitations et avons proposé une nouvelle méthode de défuzzification, basée sur l'utilisation de la distance du transport.

Enfin, nous avons apporté des contributions d'ordre plus technique, notamment dans le chapitre 4 où nous avons proposé une description lexicale de la configuration de la main et construit neuf symbolisations, utilisant quatre ensembles lexicaux. Cette description s'est révélée pertinente dans le sens où elle a permis de reconnaître efficacement des signes statiques, des signes dynamiques et qu'elle a également pu être utilisée pour la génération d'actions graduelles. Dans la reconnaissance des signes dynamiques, nous avons, en outre, proposé une nouvelle méthode de normalisation temporelle basée uniquement sur le calcul de distances entre postures ainsi qu'une méthode permettant de construire une métrique sur un ensemble de symboles d'une *Id*-symbolisation construite selon les trois processus introduits au chapitre 3. Enfin, nous avons mis en oeuvre un système de commande d'un robot mobile par le geste de la main, utilisant des commandes graduelles.

Ce travail ouvre de nombreuses perspectives, également d'ordres différents. Concernant les perspectives d'ordre théorique, nous avons déjà mentionné, en conclusion du chapitre 3, l'intérêt qu'il y aurait à pouvoir identifier un symbole s avec le singleton $\{s\}$, ces deux objets

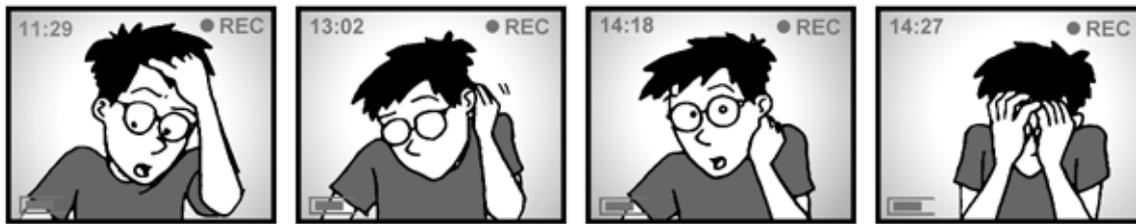
ayant la même signification dans une symbolisation. Une solution serait à terme de construire le formalisme des symbolisations sur la base de la méréologie plutôt que la théorie des ensembles. A plus court terme, certaines conjectures ont été émises concernant les relations $(\perp - T^*)$ autres que $(\perp_L - T_M)$. Il reste à confirmer ou infirmer ces conjectures, le premier cas permettrait alors de disposer d'une relation d'égalité floue pour les ϕ -symbolisations en général et pas seulement pour les *Id*-symbolisations. Ce travail de généralisation aux ϕ -symbolisations doit également être effectué pour la distance du transport qui n'est aujourd'hui applicable qu'aux descriptions floues issues d'*Id*-symbolisations.

Une autre perspective théorique, à court terme, serait de travailler à la définition d'une défuzzification symbolique basée sur la distance du transport qui soit une extension directe de la méthode des hauteurs. Une telle défuzzification symbolique devrait se réduire à la méthode des hauteurs lorsque la métrique discrète est définie sur les symboles, ce qui n'est aujourd'hui pas le cas.

Nous avons également rencontré dans le chapitre 5 une limitation concernant la définition du signe *M*. Il faudra donc étendre le formalisme des symbolisations pour pouvoir lever cette limitation et permettre la définition de signes statiques faisant intervenir des contraintes de similarité.

D'autres perspectives, d'ordre technique sont bien sûr envisageables. Les neuf symbolisations ont été construites manuellement, il serait intéressant de développer des algorithmes d'apprentissage automatique des significations permettant d'adapter rapidement le système à un nouvel utilisateur, tout comme il serait intéressant d'envisager d'autres symbolisations pour la description de la main, par exemple inspirées des systèmes de notation des signes HamNoSys et SignWriting. Concernant la reconnaissance de signes dynamiques, des modèles de Markov capables de traiter des données discrètes floues pourraient être appliqués pour la comparaison dynamique des signes. Enfin, bien d'autres types d'actions graduelles pourraient être envisagés et il serait intéressant d'étudier l'apport que peut représenter la nouvelle méthode de défuzzification symbolique pour d'autres systèmes à base de contrôleurs flous.

Enfin, ce système de description de la configuration de la main pourra être intégré dans un système plus global, faisant intervenir d'autres types de mesures, tels qu'un système de localisation de la main ou une caméra permettant d'analyser la posture générale de l'opérateur. L'approche symbolique utilisée ici permettra de simplifier la fusion de données hétérogènes provenant des différents capteurs et de nous acheminer vers la construction d'un système d'interaction homme-machine multi-modal.



"S-SO YOU'RE SAYING
A PH.D. W-WILL NOT
BRING ME FAME AND
P-PRESTIGE..? WO-
MEN DON'T GO NUTS
OVER PROFESSORS..?"

"I'M AFRAID NOT..."

"BUT AT LEAST I'LL
STILL BE RICH, RIGHT?"

"ACTUALLY, WITH TODAY'S
STARTING SALARIES FOR
PH.D'S, YOU'LL NEVER
CATCH UP TO SOMEONE
WHO STARTED WORKING
AFTER THEIR BACHELOR'S"

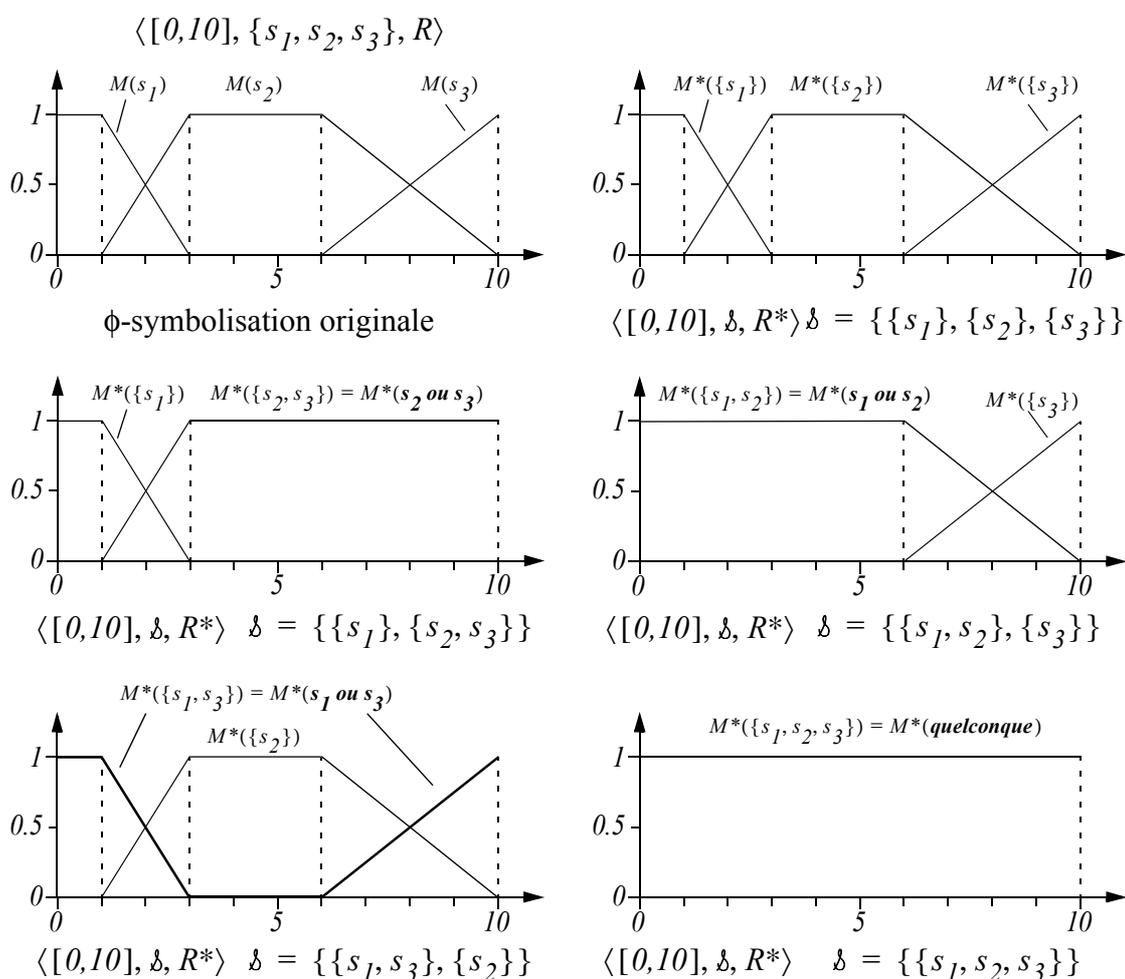
JORGE CHAM ©THE STANFORD DAILY

Annexes

1. Regroupement des symboles d'une ϕ -symbolisation à 3 symboles

n	$B(n)$	n	$B(n)$	n	$B(n)$
1	1	6	203	11	678570
2	2	7	877	12	4213597
3	5	8	4140	13	27644437
4	15	9	21147	14	190899322
5	52	10	115975	15	1382958545

Tableau 2 : Nombres de Bell.

Fig. 133 Regroupements de symboles d'une ϕ -symbolisation à trois symboles, avec $\phi = Id$.

2. Démonstration de la proposition 16 du chapitre 3

Proposition 16 : Soit une bijection ϕ croissante concave de $[0, 1]$ dans lui-même. Soit S un ensemble fini de symboles et $F_\phi(S)$ la famille de sous-ensembles flous de S définie comme suit :

$$\forall A \in F(S), \quad A \in F_\phi(S) \Leftrightarrow \sum_{s \in S} \phi(\mu_A(s)) = 1. \quad (182)$$

Alors la relation $(\perp_L^\phi - T_M)$ est une T_L -égalité sur $F_\phi(S)$.

Preuve : La relation $(\perp_L^\phi - T_M)$ est symétrique comme toute relation $(\perp - T^*)$. Elle s'écrit comme suit :

$$\forall (A, B) \in F(S)^2, \quad \mu_{(\perp_L^\phi - T_M)}(A, B) = \perp_L^\phi T_M(\mu_A(s), \mu_B(s)). \quad (183)$$

Rappelons la définition de la t-conorme \perp_L^ϕ :

$$\forall (u, v) \in [0, 1]^2, \quad \perp_L^\phi(u, v) = \phi^{-1}(\perp_L(\phi(u), \phi(v))). \quad (184)$$

Alors $\perp_L^\phi T_M(\mu_A(s), \mu_B(s)) = \phi^{-1}(\perp_L(\phi(T_M(\mu_A(s), \mu_B(s))))$ et ϕ étant croissante, $\phi(T_M(\mu_A(s), \mu_B(s))) = T_M(\phi(\mu_A(s)), \phi(\mu_B(s)))$. Finalement, la relation $(\perp_L^\phi - T_M)$ peut s'écrire comme suit :

$$\forall (A, B) \in F(S)^2, \quad \mu_{(\perp_L^\phi - T_M)}(A, B) = \phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(A), \phi(B))), \quad (185)$$

où $\phi(A)$ est le sous-ensemble flou dont le degré d'appartenance pour tout symbole s vaut $\phi(\mu_A(s))$. Si A est un élément de $F_\phi(S)$ alors $\phi(A)$ est un élément de $F_{Id}(S)$, partie de $F(S)$ sur laquelle $(\perp_L - T_M)$ est une T_L -égalité. Alors, avec $\phi(I) = I$, on a :

$$\begin{aligned}
\forall (A, B) \in F_\phi(S)^2, \quad \mu_{(\perp_L^\phi - T_M)}(A, B) = I &\Leftrightarrow \mu_{(\perp_L - T_M)}(\phi(A), \phi(B)) = I, \\
&\Leftrightarrow \phi(A) = \phi(B), \\
&\Leftrightarrow A = B.
\end{aligned} \tag{186}$$

Ainsi, pour que $(\perp_L^\phi - T_M)$ soit une T_L -égalité, il ne reste plus qu'à montrer qu'elle est T_L -transitive sur $F_\phi(S)$, c'est-à-dire qu'elle vérifie :

$$\forall (A, B, C) \in F_\phi(S)^3, \quad T_L\left(\mu_{(\perp_L^\phi - T_M)}(A, B), \mu_{(\perp_L^\phi - T_M)}(B, C)\right) \leq \mu_{(\perp_L^\phi - T_M)}(A, C). \tag{187}$$

Or, pour tout triplet A, B, C de $F_\phi(S)^3$, on a les équivalences suivantes :

$$\begin{aligned}
T_L\left(\mu_{(\perp_L^\phi - T_M)}(A, B), \mu_{(\perp_L^\phi - T_M)}(B, C)\right) &\leq \mu_{(\perp_L^\phi - T_M)}(A, C) \\
\Leftrightarrow T_L\left(\phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(A), \phi(B))), \phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(B), \phi(C)))\right) &\leq \phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(A), \phi(C))) \\
\Leftrightarrow \max\left(\phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(A), \phi(B))) + \phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(B), \phi(C))) - I, \theta\right) &\leq \phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(A), \phi(C))).
\end{aligned}$$

Le cas où la partie gauche de cette dernière inégalité est nulle rend cette dernière vraie. Il reste à prouver l'inégalité suivante dans le cas contraire :

$$\phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(A), \phi(B))) + \phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(B), \phi(C))) \leq \phi^{-1}(\mu_{(\perp_L - T_M)}(\phi(A), \phi(C))) + I. \tag{188}$$

$$AB = \mu_{(\perp_L - T_M)}(\phi(A), \phi(B)),$$

$$BC = \mu_{(\perp_L - T_M)}(\phi(B), \phi(C)),$$

$$AC = \mu_{(\perp_L - T_M)}(\phi(A), \phi(C)).$$

Posons :

Alors l'inégalité (188) restant à prouver s'écrit :

$$\phi^{-1}(AB) + \phi^{-1}(BC) \leq \phi^{-1}(AC) + I \tag{189}$$

Dans (189), seul le cas où $AC < AB$, $AC < BC$ et $BC < I$ n'est pas trivial. Supposons alors qu'on ait :

$$AB \leq BC, \quad (190)$$

ce qui ne change rien à la généralité du problème. Dans ce cas, on a finalement $AC < AB \leq BC < I$ et, ϕ étant concave, son inverse est convexe et donne :

$$\frac{\phi^{-1}(AB) - \phi^{-1}(AC)}{AB - AC} \leq \frac{\phi^{-1}(I) - \phi^{-1}(BC)}{I - BC}. \quad (191)$$

Enfin, la T_L -transitivité de $(\perp_L - T_M)$ sur $F_{Id}(S)$ donne $AB + BC \leq AC + I$ soit $AB - AC \leq I - BC$. Alors, si on reprend (191), on obtient :

$$\frac{\phi^{-1}(AB) - \phi^{-1}(AC)}{I - BC} \leq \frac{\phi^{-1}(AB) - \phi^{-1}(AC)}{AB - AC} \leq \frac{\phi^{-1}(I) - \phi^{-1}(BC)}{I - BC},$$

ce qui, en simplifiant par $I - BC$, prouve (189) et donc la T_L -transitivité de $(\perp_L^\phi - T_M)$ sur $F_\phi(S)$. ■

3. Reconnaissance de signes dynamiques

		Gestes comparés avec la distance d_G^∞ Postures comparées avec la distance $d_{(\perp_L - T_M)^\circ (D_P \times D_P)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesR	Au Revoir	0,7	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	Viens ici	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	Attrape	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	Clic2	1,0	1,0	1,0	0,4	0,8	1,0	1,0	1,0
	Clic	1,0	1,0	1,0	0,9	0,5	1,0	1,0	1,0
	Regarde	1,0	1,0	1,0	1,0	1,0	0,7	1,0	1,0
	Rotation	1,0	1,0	1,0	1,0	1,0	1,0	0,7	1,0
	Ciseaux	1,0	1,0	1,0	1,0	1,0	1,0	1,0	0,7

		Gestes comparés avec la distance d_G^∞ Postures comparées avec la distance $d_{tr^\circ (D_P \times D_P)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesR	Au Revoir	3,0	10,0	9,9	10,0	10,0	10,0	10,0	10,0
	Viens ici	10,0	6,0	9,7	10,0	10,0	7,3	9,6	10,0
	Attrape	9,9	9,9	6,2	10,0	10,0	10,0	10,0	10,0
	Clic2	10,0	10,0	9,9	2,3	4,2	10,0	10,0	10,0
	Clic	10,0	10,0	10,0	4,6	2,6	10,0	10,0	10,0
	Regarde	10,0	8,5	10,0	10,0	10,0	5,7	9,5	10,0
	Rotation	10,0	9,7	10,0	10,0	10,0	9,2	5,0	10,0
	Ciseaux	10,0	10,0	10,0	10,0	10,0	10,0	10,0	6,1

Fig. 134 Résultat de la comparaison des gestes de la série GestesR avec la distance d_G^∞ .

		Gestes comparés avec la distance d_G^1 Postures comparées avec la distance $d_{(\perp_L - T_M) \circ (D_P \times D_P)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesN	Au Revoir	2,1	10,0	10,0	10,0	10,0	10,0	10,0	10,0
	Viens ici	10,0	4,2	9,8	10,0	10,0	6,6	9,7	10,0
	Attrape	9,9	9,9	4,9	10,0	9,9	10,0	10,0	10,0
	Clic2	10,0	10,0	9,9	3,1	4,8	10,0	10,0	10,0
	Clic	10,0	10,0	9,9	5,2	2,6	10,0	10,0	10,0
	Regarde	10,0	8,1	10,0	10,0	10,0	3,5	9,4	10,0
	Rotation	10,0	9,8	10,0	10,0	10,0	9,5	4,5	10,0
	Ciseaux	10,0	10,0	10,0	10,0	10,0	10,0	10,0	5,2

		Gestes comparés avec la distance d_G^1 Postures comparées avec la distance $d_{tr \circ (D_P \times D_P)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesN	Au Revoir	2,4	112,8	70,0	94,6	95,0	114,6	103,9	84,3
	Viens ici	114,5	7,5	65,7	47,7	48,1	14,8	24,7	46,3
	Attrape	72,0	64,7	10,4	63,5	59,0	68,9	63,8	64,1
	Clic2	93,6	51,6	66,7	4,1	9,6	43,3	54,3	78,6
	Clic	95,2	48,3	61,8	9,3	5,2	40,6	49,9	73,9
	Regarde	113,3	21,4	72,6	43,6	45,2	7,9	28,2	53,3
	Rotation	103,4	30,5	63,6	59,3	55,4	36,5	11,3	39,3
	Ciseaux	91,8	46,8	66,4	80,0	77,2	53,3	41,6	9,7

Fig. 135 Résultat de la comparaison des gestes de la série GestesN avec la distance d_G^1 .

		Gestes comparés avec la distance d_G^∞ Postures comparées avec la distance $d_{(\perp_L - T_M) \circ (D_P \times D_P)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesN	Au Revoir	0,5	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	Viens ici	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	Attrape	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	Clic2	1,0	1,0	1,0	0,6	0,9	1,0	1,0	1,0
	Clic	1,0	1,0	1,0	0,9	0,5	1,0	1,0	1,0
	Regarde	1,0	1,0	1,0	1,0	1,0	0,7	1,0	1,0
	Rotation	1,0	1,0	1,0	1,0	1,0	1,0	0,5	1,0
	Ciseaux	1,0	1,0	1,0	1,0	1,0	1,0	1,0	0,6

		Gestes comparés avec la distance d_G^∞ Postures comparées avec la distance $d_{tr \circ (D_P \times D_P)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesN	Au Revoir	0,5	12,4	11,4	10,1	10,2	11,9	11,0	8,8
	Viens ici	12,2	1,9	10,2	6,0	5,8	3,7	3,6	5,5
	Attrape	11,7	10,3	3,3	10,2	9,2	11,2	9,5	7,6
	Clic2	10,0	6,0	10,5	0,6	1,7	5,9	5,8	8,9
	Clic	10,1	5,6	9,8	1,6	0,9	5,6	5,5	8,5
	Regarde	12,0	4,3	10,9	5,8	5,8	1,7	4,0	6,9
	Rotation	11,3	3,5	8,7	6,7	6,0	4,7	1,5	4,4
	Ciseaux	9,6	6,0	7,4	8,8	8,5	6,8	4,9	1,4

Fig. 136 Résultat de la comparaison des gestes de la série GestesN avec la distance d_G^∞ .

4. Composantes des signes dynamiques

La figure suivante donne les signes statiques de début et de fin ainsi que les composantes statiques des huit signes dynamiques.

		L _{orientPouce}	L _{flexionPouce}	L _{flexionIndex}	L _{flexionMajeur}	L _{flexionAnnulaire}	L _{flexionAuriculaire}	I _{ecartIndexMajeur}	I _{ecartMajeurAnnulaire}	I _{ecartAnnulaireAuriculaire}
D _{Attrape}	de_coté ou externe ou en_avant	tendu	tendu	tendu	tendu	tendu	quelconque	quelconque	quelconque	
F _{Attrape}	en_avant ou interne_dessus	quelconque	plié	plié	plié	plié	quelconque	quelconque	quelconque	
St _{Attrape}	quelconque	quelconque	quelconque	quelconque	quelconque	quelconque	quelconque	quelconque	quelconque	
D _{Ciseaux}	en_avant ou interne_dessus ou interne_dessous	quelconque	tendu ou carré	tendu ou carré	plié ou rond ou griffe	plié ou rond ou griffe	écartés	quelconque	quelconque	
F _{Ciseaux}	en_avant ou interne_dessus ou interne_dessous	quelconque	tendu ou carré	tendu ou carré	plié ou rond ou griffe	plié ou rond ou griffe	écartés	quelconque	quelconque	
St _{Ciseaux}	en_avant ou interne_dessus ou interne_dessous	quelconque	tendu ou carré	tendu ou carré	plié ou rond ou griffe	plié ou rond ou griffe	quelconque	quelconque	quelconque	
D _{Clic}	externe	tendu	plié	plié	plié	plié	quelconque	quelconque	quelconque	
F _{Clic}	externe	tendu	plié	plié	plié	plié	quelconque	quelconque	quelconque	
St _{Clic}	quelconque	quelconque	plié	plié	plié	plié	quelconque	quelconque	quelconque	
D _{Clic2}	externe	tendu	plié	plié	plié	plié	quelconque	quelconque	quelconque	
F _{Clic2}	externe	tendu	plié	plié	plié	plié	quelconque	quelconque	quelconque	
St _{Clic2}	quelconque	quelconque	plié	plié	plié	plié	quelconque	quelconque	quelconque	
D _{Regarde}	interne_dessus ou en_avant	quelconque	plié ou griffe	plié	plié	plié	quelconque	quelconque	quelconque	
F _{Regarde}	interne_dessus ou en_avant	quelconque	tendu ou carré	plié	plié	plié	quelconque	quelconque	quelconque	
St _{Regarde}	interne_dessus ou en_avant	quelconque	quelconque	plié	plié	plié	quelconque	quelconque	quelconque	
D _{Viens_ici}	en_avant ou interne_dessus	quelconque	tendu ou carré	plié	plié	plié	quelconque	quelconque	quelconque	
F _{Viens_ici}	en_avant ou interne_dessus	quelconque	tendu ou carré	plié	plié	plié	quelconque	quelconque	quelconque	
St _{Viens_ici}	en_avant ou interne_dessus	quelconque	quelconque	plié	plié	plié	quelconque	quelconque	quelconque	
D _{Au_Revoir}	externe ou de_coté ou en_avant	tendu	tendu	tendu	tendu	tendu	serrés	serrés	serrés	
F _{Au_Revoir}	externe ou de_coté ou en_avant	tendu	tendu	tendu	tendu	tendu	serrés	serrés	serrés	
St _{Au_Revoir}	externe ou de_coté ou en_avant	tendu	quelconque	quelconque	quelconque	quelconque	quelconque	quelconque	quelconque	
D _{Rotation}	en_avant ou interne_dessus ou interne_dessous	quelconque	tendu ou carré	plié ou rond ou griffe	plié ou rond ou griffe	plié ou rond ou griffe	serrés	quelconque	quelconque	
F _{Rotation}	en_avant ou interne_dessus ou interne_dessous	quelconque	tendu ou carré	plié ou rond ou griffe	plié ou rond ou griffe	plié ou rond ou griffe	serrés	quelconque	quelconque	
St _{Rotation}	en_avant ou interne_dessus ou interne_dessous	quelconque	tendu ou carré	plié ou rond ou griffe	plié ou rond ou griffe	plié ou rond ou griffe	quelconque	quelconque	quelconque	

Fig. 137 Trois composantes des huit signes dynamiques.

$$\begin{aligned}
\langle P, L_{Attrape}, R_{Attrape} \rangle &= \langle P, L_{flexionPouce}, R_{flexionPouce} \rangle \times \langle P, L_{orientPouce}, R_{orientPouce} \rangle \\
&\quad \times \langle P, L_{flexionDoigt}, R_{flexionIndex} \rangle \times \langle P, L_{flexionDoigt}, R_{flexionMajeur} \rangle \\
&\quad \times \langle P, L_{flexionDoigt}, R_{flexionAnnulaire} \rangle \times \langle P, L_{flexionDoigt}, R_{flexionAuriculaire} \rangle \\
\langle P, L_{ciseaux}, R_{ciseaux} \rangle &= \langle P, L_{écart}, R_{écartIndexMajeur} \rangle \\
\langle P, L_{Clic}, R_{Clic} \rangle &= \langle P, L_{flexionPouce}, R_{flexionPouce} \rangle \times \langle P, L_{orientPouce}, R_{orientPouce} \rangle \\
\langle P, L_{Clic2}, R_{Clic2} \rangle &= \langle P, L_{flexionPouce}, R_{flexionPouce} \rangle \times \langle P, L_{orientPouce}, R_{orientPouce} \rangle \\
\langle P, L_{Regarde}, R_{Regarde} \rangle &= \langle P, L_{flexionDoigt}, R_{flexionIndex} \rangle \\
\langle P, L_{Viens_Ici}, R_{Viens_Ici} \rangle &= \langle P, L_{flexionDoigt}, R_{flexionIndex} \rangle \\
\langle P, L_{Au_Revoir}, R_{Au_Revoir} \rangle &= \langle P, L_{flexionDoigt}, R_{flexionIndex} \rangle \times \langle P, L_{flexionDoigt}, R_{flexionMajeur} \rangle \\
&\quad \times \langle P, L_{flexionDoigt}, R_{flexionAnnulaire} \rangle \times \langle P, L_{flexionDoigt}, R_{flexionAuriculaire} \rangle \\
\langle P, L_{Rotation}, R_{Rotation} \rangle &= \langle P, L_{flexionDoigt}, R_{flexionIndex} \rangle \times \langle P, L_{écart}, R_{écartIndexMajeur} \rangle
\end{aligned}$$

Fig. 138 Symbolisations utilisées pour la comparaison d'un geste avec les composantes dynamiques des signes.

5. Reconnaissance de signes dynamiques - Méthode n° 2

		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesN	Au Revoir	X	-	-	-	-	-	-	-
	Viens ici	-	X	-	-	-	-	X	-
	Attrape	-	-	X	-	-	-	-	-
	Clic2	-	-	-	X	X	-	-	-
	Clic	-	-	-	X	X	-	-	-
	Regarde	-	-	-	-	-	X	-	-
	Rotation	-	-	-	-	-	-	X	-
	Ciseaux	-	-	-	-	-	-	-	X

Fig. 139 Adéquation des gestes de GestesN avec les trois caractéristiques statiques des signes.

		Gestes comparés avec la distance d_G^∞ Postures comparées avec la distance $d_{(\perp L - T_M) \circ (D_S \times D_S)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesR	Au Revoir	0,6	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	Viens ici	1,0	0,9	1,0	1,0	1,0	1,0	1,0	1,0
	Attrape	1,0	1,0	0,7	1,0	1,0	1,0	1,0	1,0
	Clic2	0,9	1,0	1,0	0,4	0,9	1,0	1,0	1,0
	Clic	0,9	1,0	1,0	0,9	0,5	1,0	1,0	1,0
	Regarde	1,0	1,0	1,0	1,0	1,0	0,6	1,0	1,0
	Rotation	1,0	1,0	1,0	1,0	1,0	1,0	0,8	1,0
	Ciseaux	1,0	1,0	1,0	1,0	1,0	1,0	1,0	0,4

		Gestes comparés avec la distance d_G^∞ Postures comparées avec la distance $d_{Tr \circ (D_S \times D_S)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesR	Au Revoir	1,1	7,9	8,0	8,0	8,0	7,9	6,6	4,9
	Viens ici	1,9	0,4	2,0	2,0	2,0	1,9	1,9	1,9
	Attrape	6,3	3,6	1,0	4,2	4,1	5,3	4,0	5,1
	Clic2	1,5	2,2	2,4	0,8	1,3	1,9	2,2	2,2
	Clic	1,3	2,3	2,4	1,3	0,7	1,9	2,2	2,1
	Regarde	2,0	2,0	2,0	2,0	2,0	0,4	2,0	2,0
	Rotation	1,5	2,0	2,6	3,0	3,0	2,3	1,2	1,7
	Ciseaux	1,0	1,0	1,0	1,0	1,0	1,0	1,0	0,4

Fig. 140 Résultat de la comparaison des gestes de la série GestesR avec la distance d_G^∞ .

		Gestes comparés avec la distance d_G^1 Postures comparées avec la distance $d_{(\perp_L - T_M) \circ (D_S \times D_S)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesN	Au Revoir	2,0	10,0	8,7	10,0	10,0	10,0	10,0	10,0
	Viens ici	7,6	2,5	6,3	8,8	8,7	6,0	8,7	7,3
	Attrape	8,7	9,4	3,8	9,9	9,9	9,9	10,0	9,9
	Clic2	5,8	10,0	8,3	2,0	3,2	10,0	9,6	9,7
	Clic	5,0	10,0	7,9	4,4	1,0	10,0	9,5	9,7
	Regarde	7,8	6,8	6,9	8,3	8,1	2,1	8,4	7,4
	Rotation	7,3	9,7	9,3	10,0	10,0	8,8	1,4	8,1
	Ciseaux	6,1	3,4	5,5	6,1	6,1	5,2	7,1	1,3

		Gestes comparés avec la distance d_G^1 Postures comparées avec la distance $d_{ir \circ (D_S \times D_S)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesN	Au Revoir	2,4	68,2	40,2	79,8	79,7	71,5	64,8	42,4
	Viens ici	9,2	1,2	7,8	12,7	12,6	8,5	10,3	8,9
	Attrape	35,2	29,5	6,3	29,5	29,4	35,2	36,3	41,0
	Clic2	7,9	18,0	13,1	2,6	5,3	20,5	18,2	18,9
	Clic	6,6	18,8	13,1	5,5	1,4	21,2	17,9	18,7
	Regarde	10,9	8,8	12,0	11,0	10,8	2,3	11,5	10,5
	Rotation	9,5	16,5	16,1	24,8	24,8	17,8	2,1	11,1
	Ciseaux	6,1	3,4	5,1	6,1	6,1	5,0	7,1	1,3

Fig. 141 Résultat de la comparaison des gestes de la série GestesN avec la distance d_G^1 .

		Gestes comparés avec la distance d_G^∞ Postures comparées avec la distance $d_{(\perp_L - T_M) \circ (D_S \times D_S)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesN	Au Revoir	0,5	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	Viens ici	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	Attrape	1,0	1,0	0,7	1,0	1,0	1,0	1,0	1,0
	Clic2	0,9	1,0	1,0	0,4	0,9	1,0	1,0	1,0
	Clic	0,8	1,0	1,0	0,9	0,2	1,0	1,0	1,0
	Regarde	1,0	1,0	1,0	1,0	1,0	0,6	1,0	1,0
	Rotation	1,0	1,0	1,0	1,0	1,0	1,0	0,3	1,0
	Ciseaux	1,0	0,9	1,0	1,0	1,0	1,0	1,0	0,5

		Gestes comparés avec la distance d_G^∞ Postures comparées avec la distance $d_{ir \circ (D_S \times D_S)}$							
		Gestes prototypes							
		Au Revoir	Viens ici	Attrape	Clic2	Clic	Regarde	Rotation	Ciseaux
GestesN	Au Revoir	0,5	7,9	7,8	8,0	8,0	8,0	6,8	4,9
	Viens ici	1,9	0,5	2,0	2,0	2,0	2,0	1,9	1,9
	Attrape	6,4	4,4	1,3	4,1	4,1	6,3	4,9	5,1
	Clic2	1,5	2,0	2,5	0,5	1,3	2,4	2,2	2,2
	Clic	1,2	2,5	2,5	1,3	0,2	2,4	2,1	2,2
	Regarde	2,0	2,0	2,0	2,0	2,0	0,6	2,0	2,0
	Rotation	1,6	2,2	2,8	3,0	3,0	2,6	0,5	1,5
	Ciseaux	1,0	0,9	1,0	1,0	1,0	1,0	1,0	0,5

Résultat de la comparaison des gestes de la série GestesN avec la distance d_G^∞ .

Bibliographie

- [Aukstakalnis 92] Aukstakalnis S., Blatner D., "Silicon mirage. The art and science of virtual reality", Eds. Roth S.F, PeachPit Press, Inc., Berkeley, CA, 1992, 318 pages.
- [Bauer 02] Bauer B., Kraiss K.F., "Video-based continuous sign language recognition using self organizing subunits", Dans: IEEE *International Conference on Pattern Recognition*, Quebec, Canada, Aout, 2002, pp.434-437.
- [Benoit 93] Benoit E., "Capteurs symboliques et capteurs flous: un nouveau pas vers l'intelligence", *Thèse de doctorat*, Université de Savoie, Janvier 1993, 164 pages.
- [Benoit 03] E. Benoit, T. Allevard, T. Ukegawa, H. Sawada, "Fuzzy Sensor for Gesture Recognition Based on Shape Recognition of Hand", *Proc. of Int. Symp. on Virtual Environments, Human-Computer Interfaces, and Measurement Systems (VECIMS)*, Lugnano, Suisse, 2003, pp. 63-67.
- [Berge 66] Berge C., "Espaces topologiques. Fonctions Multivoques", Dunod, Paris, 1966, 283 pages.
- [Bezdek 99] Bezdek J.C., Keller J., Krisnapuram R., Nikhil R.P., "Fuzzy models and algorithms for pattern recognition and image processing", Eds. Dubois D., Prade H., Kluwer Academic Publishers, 1999, 776 pages.
- [Bloch 99] Bloch I., "On fuzzy distances and their use in image processing under imprecision", *Pattern Recognition*, Vol. 32, 1999, pp. 1873-1895.

- [Boixader 00] Boixader D., Jacas J., Recasens J., "Fuzzy equivalence relations: advanced material", Dans: *Fundamentals of fuzzy sets*, Eds. Dubois D., Prade H., Kluwer Academic Publishers, Norwell, Massachusetts, 2000, pp. 261-290.
- [Bolt 80] Bolt R.A., "Put-That-There: voice and gesture at the graphic interface", *Computer Graphics*, Vol. 14, N° 3, 1980, pp. 262-270.
- [Bouvier 96] Bouvier A., Georges M., Le Lionnais F., "Dictionnaire des mathématiques", Presses Universitaires de France, Paris, France, 1996, 960 pages.
- [Bowman 02] Bowman D.A et al., "Novel use of Pinch Gloves for Virtual interactive techniques", *Virtual Reality*, Vol. 6, N° 3, 2002, pp. 122-129.
- [Braffort 96] Braffort A., "Reconnaissance et compréhension de gestes, application à la langue des signes", *Thèse de doctorat*, Université Paris-XI, Juin 1996, 227 pages.
- [Cadoz 94] Cadoz C., "Le geste, canal de la communication homme/machine - la communication instrumentale", *Techniques et sciences informatiques*, Vol. 13, N° 1, 1994, pp. 31-61.
- [Cadoz 00] Cadoz C., Wanderley M. M., "Gesture - Music", Dans: *Trends in Gestural Control of Music*, Eds. Wanderley M.M., Battier M., Ircam, 2000, pp. 71-94.
- [Chen 03] Chen F.-S., Fu C.-M., Huang C.-L., "Hand gesture recognition using a real-time tracking method and hidden Markov models", *Image and Vision Computing*, Vol. 21, 2003, pp. 745-758.
- [Corradini 01] Corradini A., Gross H.M., "Real-time gestures recognition by means of hybrid recognizers", Dans: *Proceedings of the Fourth International Gesture Workshop (GW'01)*, London, UK, 2001, pp.34-46.
- [De Baets 96] De Baets B., Mesiar R., "Fuzzy partitions and their entropy", Dans: *Proceedings of the 6th Int. Conf. on Information Processing and Management of Uncertainty (IPMU'96)*, Grenade, Espagne, 1996, pp.1419-1424.
- [De Baets 98] De Baets B., Mesiar R., "T-partitions", *Fuzzy Sets and Systems*, Vol. 97, 1998, pp. 211-223.
- [De Baets 02] B. De Baets, R. Mesiar, "Metrics and T-Equalities", *Journal of Mathematical Analysis and Applications*, Vol. 267, 2002, pp. 531-547.
- [De Cock 03] De Cock M., Kerre E., "On (un)suitable relations to model approximate equality", *Fuzzy Sets and Systems*, Vol. 133, N° 2, 2003, pp. 137-153.
- [DeFanti 77] DeFanti T.A., Sandin D.J., "Final report to the national endowment of the arts", *US NEA R60-34-163*, University of Illinois at Chicago Circle, Illinois, 1977.
- [De Nysse 379] De Nysse G., "Traité de la création de l'Homme", 379 Ap J.C.
- [Devroye 96] Devroye L., Györfi L., Lugosi G., "A Probabilistic Theory of Pattern Recognition", Springer, 1996, 636 pages.

- [de Werra 03] D. de Werra, T. M. Liebling, J.-F. Hêche, “Recherche opérationnelle pour ingénieurs 1”, Presses polytechniques et universitaires romandes, 2003.
- [Dipietro 03] Dipietro L., Sabatini A.M., Dario P., “Evaluation of an instrumented glove for hand-movement acquisition”, *Journal of Rehabilitation Research and Development*, Vol. 40, N° 2, 2003, pp. 179-190.
- [Dreyfus 04] Dreyfus G. et al., “Réseaux de neurones, méthodologies et applications, 2nd édition”, Editions Eyrolles, Paris, 2004, 417 pages.
- [Dubois 83] Dubois D., Prade H., “On distance between fuzzy points and their use for plausible reasoning”, Dans: *International Conference on Systems, Man and Cybernetics*, 1983, pp.300-303.
- [Dubois 87] Dubois D., Prade H., “Fuzzy numbers : an overview”, Dans: *Analysis of Fuzzy Information vol.1: Mathematics and Logic*, Eds. Bezdek J., CRC Press, 1987.
- [Dubois 97] Dubois D., Prade H., “The three semantics of fuzzy sets”, *Fuzzy Sets and Systems*, Vol. 90, 1997, pp. 141-150.
- [Dubois 00] Dubois D., Ostasiewicz W., Prade H., “Fuzzy sets : history and basic notions”, Dans: *Fundamentals of fuzzy sets*, Eds. Dubois D., Prade H., Kluwer Academic Publishers, Norwell, Massachusetts, 2000, pp. 21-124.
- [Duda 01] Duda, R.O. Hart E.H., Stork G.D., “Pattern Classification”, John Wiley 1 Sons, 2001, 654 pages.
- [Edmison 02] Edmison J. et al. “using piezoelectric materials for wearable electronic textiles”, Dans: *International Symposium on Wearable Computers*, Seattle, WA, USA, IEEE, Oct, 2002, pp.41-48.
- [Efron 41] Efron D., “Gesture and Environment”, King’s Crown, New York, 1941. Réédité : “Gesture, Race and Culture”, Mouton, The Hague, 1992.
- [Ehrenmann 01] Ehrenmann M., Lutticke T., Dillmann R., “Dynamic gestures as an input device for directing a mobile platform”, Dans: *IEEE International Conference on Robotics and Automation*, Séoul, Corée du Sud, 2001, pp.2596-2601.
- [Eisenstein 01] Eisenstein J. et Al., “Analysis of clustering techniques to detect hand signs”, Dans: *Proceedings of the International Symposium on Intelligent Multimedia, Video and Speech Processing*, Hong-Kong, Mai 2001.
- [Eiter 97] Eiter E., Mannila H., “Distance measures for point sets and their computations”, *Acta Informatica*, Vol. 34, N° 2, 1997, pp. 109-133.
- [Ekman 72] Ekman P., Friesen W.V., “Hand Movements”, *The Journal of Communication*, Vol. 22, Dec, 1972, pp. 353-374.
- [Fang 04] Fang G., Gao W., “Large vocabulary Sign Language recognition based on fuzzy decision trees”, *IEEE Transactions on Systems, Man and Cybernetics - Part A : Systems and Humans*, Vol. 34, N° 3, Mai, 2004, pp. 305-314.
- [Fels 93] Fels S.S., Hinton G.E., “Glove-Talk: A neural network interface between a data glove and a speech synthesizer”, *IEEE Transactions on Neural Networks*,

- Vol. 4, N° 1, 1993, pp. 2-8.
- [Fels 98]** Fels S.S., Hinton G.E., "Glove-TalkII: A neural network interface which maps gestures to parallel formant speech synthesiser controls", *IEEE Transactions on Neural Networks*, Vol. 9, N° 1, 1998, pp. 205-212.
- [Finkelstein 75]** Finkelstein L., "Representation by symbol systems as an extension of the concept of measurement", *Kybernetes*, Vol. 4, 1975, pp. 215-223.
- [Finkelstein 94]** Finkelstein L., "Measurement and instrumentation science - an analytical review", *Measurement*, Vol. 14, N°1, 1994, pp. 3-14.
- [Foulloy 95]** Foulloy L. Galichet S. "Typology of fuzzy controllers", Dans: *Theoretical Aspects of Fuzzy Control*, Eds. Nguyen H.T., Sugeno M., Tong, R., Yager, R.R., John Wiley & Sons Inc., 1995, pp. 65-90.
- [Foulloy 04.a]** Foulloy L., Benoit E., "Construction d'une classe de relations d'équivalence floues-I", Dans: *Rencontres francophones sur la Logique Floue et ses Applications (LFA'2004)*, Nantes, France, novembre 2004, pp. 9-16.
- [Foulloy 04.b]** Foulloy L., Benoit E., "Construction d'une classe de relations d'équivalence floues-II", Dans: *Rencontres francophones sur la Logique Floue et ses Applications (LFA'2004)*, Nantes, France, novembre 2004, pp. 19-24.
- [Foulloy 04.c]** Foulloy L., Benoit E., "Building a class of fuzzy equivalence relations", *Rapport Interne du Listic*, 2004.
- [Fuchs 03]** Fuchs Ph., Moreau G, "Traité de la réalité virtuelle - Vol1 : Fondements et interfaces comportementales", Les presses de l'école des mines, Paris, 2003, 515 pages.
- [Galichet 01]** Galichet S., "Contrôle flou : de l'interpolation numérique au codage de l'expertise", *Habilitation à diriger des recherches*, Université de Savoie, Dec 2001, 124 pages.
- [Gao 00]** Gao W., Ma J., Wu J., Wang C., "Sign language recognition based on HMM/ANN/DP", *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 14, N° 5, 2000, pp. 587-602.
- [Gentile 92]** Gentile C.T. et al., "Angular displacement sensors", *US Patent 5.086.785*, Gentile Entertainment Inc., Fev, 1992.
- [Gottwald 93]** Gottwald S., "Fuzzy Sets and Fuzzy Logic", Vieweg & Sohn, Wiesbaden, 1993, 216 pages.
- [Grant 98]** Grant H., Lai C.-K., "Simulation modeling with artificial reality technology (SMART): an integration of virtual reality and simulation modeling", Dans: *Winter Simulation Conference*, Washington DC, Dec 1998, pp.437-441.
- [Grimes 83]** Grimes G.J., "Digital data entry glove interface device", *US Patent 4.414.537*, Bell Telephone Laboratories, N.J., Nov, 1938.
- [Guillaume 04]** Guillaume S., Charnomordic B., "Generating an interpretable family of fuzzy partitions from data", *IEEE Transactions on Fuzzy Systems*, Vol. 12, N° 3,

- 2004, pp. 324-335.
- [Gupta 96]** Gupta V., Reddy N.P., “Surface electromyogram for the control of anthropomorphic teleoperator fingers”, Dans: *Medicine Meets Virtual Reality*, Amsterdam, Hollande, IO Press, 1996, pp.482-487.
- [Hernandez 02.a]** Hernandez-Hebollar J.L., Kyriakopoulos N., Lindeman R.Q., “The AcceleGlove a whole hand input for virtual reality”, Dans: 29th International Conference on Computer Graphics and Interactive Techniques, 2002.
- [Hernandez 02.b]** Hernandez-Hebollar J.L., Lindeman R.Q., Kyriakopoulos N., “A multi-class pattern recognition system for practical finger spelling translation”, Dans: Fourth IEEE International Conference on Multimodal Interfaces, 2002, pp.185-190.
- [Hernandez 04]** Hernandez-Hebollar J.L., Kyriakopoulos N., Lindeman R.Q., “A new instrumented approach for translating American Sign Language into sound and text”, Dans: IEEE *International Conference on Automatic Face and Gesture Recognition*, Séoul, Corée, Editeur, 2004, pp.547-552.
- [Hienz 99]** Hienz H., Bauer B., Kraiss K.F., “HMM-based continuous sign language recognition using stochastic grammars”, Dans: *Gesture Workshop GW'99*, Gif-Sur-Yvette, France, Mars, 1999, pp.185-196.
- [Holden 96]** Holden E.J., Owens R., Roy G.G., “Hand movement classification using an adaptive fuzzy system”, *International Journal of Expert System*, Vol. 9, N° 4, 1996, pp. 465-480.
- [Holzrichter 01]** Holzrichter J., Ng L.C., “Speech articulator and user gesture measurements using micropower, interferometric EM-sensors”, Dans: *IEEE Instrumentation and Measurement Technology Conference*, Budapest, Hongrie, Mai 2001, pp.1942-1946.
- [Howard 01]** Howard B., Howard S., “LightGlove: wrist-worn virtual typing and pointing”, Dans: 5th *IEEE International Symposium on Wearable Computers (ISWC'01)*, Zurich, Suisse, Octobre, 2001, pp.172-173.
- [Huang 95]** Huang T.S., Pavlovic V.I., “Hand gesture Modelling, Analysis and Synthesis”, Dans: *Proceedings of the International Workshop on Automatic Face and Gesture Recognition (FG'95)*, Zurich, 1995, pp. 73-79.
- [Huang 02]** Huang Y.S. et al., “Prototype optimization for nearest neighbor classification”, *Pattern Recognition*, Vol. 35, N° 6, 2002, pp. 1237-1245.
- [Iba 99]** Iba S., Weghe J.M.V., Paredis C.J.J., Khosla P.K., “An architecture for gesture-based control of mobile robots”, Dans: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Kyongju, Corée du Sud, Octobre, 1999, Vol. 2, pp.851-857.
- [Ishikawa 02]** Ishikawa M., Sasaki N., “Gesture recognition based on SOM using multiple sensors”, Dans: *Proceedings of the 9th International Conference on Neural Information Processing (ICONIP'02)*, 2002, pp.1300-1304.
- [Janis 03]** Janis V., “Resemblance is a nearness”, *Fuzzy Sets and Systems*, Vol. 133, N°

- 2, 2003, pp. 171-173.
- [Kadous 95]** Kadous W., "GRASP: recognition of Australian sign language using instrumented gloves", *Thèse de Master*, University of new South Wales, Oct. 1995.
- [Karlsson 98.a]** Karlsson B., "Fuzzy measures for sensor data fusion in industrial recycling", *Measurement Science & Technology*, Vol. 9, 1998, pp. 907-912.
- [Karlsson 98.b]** Karlsson N., Karlsson B., Wide P., "A glove equipped with finger flexion sensors as a command generator used in a fuzzy control system", *IEEE Transactions on Instrumentation and Measurement*, Vol. 47, N° 5, Octobre, 1998, pp. 1330-1334.
- [Kendon 88]** Kendon A. "How gestures can become like words". Dans *Crosscultural perspectives in nonverbal communication*, Ed. Poyatos F., Toronto, Hogrefe, 1988, pp. 131-141.
- [Kim 96]** Kim J.S., Jang W., Bien Z., "A dynamic gesture recognition for the Korean Sign Language", *IEEE transactions on Systems, Man and Cybernetics, partB*, Vol. 26, N° 2, 1996, pp. 354-359.
- [Kindratenko 01]** Kindratenko V., "A comparison of the accuracy of an electromagnetic and a hybrid ultrasound inertia position tracking system", *Presence*, Vol. 10, N° 6, 2001, pp. 657-663.
- [Klement 00]** Klement E.P., Mesiar R., Pap E., "Triangular norms", *Trends in Logic, Vol. 8*, Kluwer Academic Publishers, 2000.
- [Kohler 98]** Kohler M., Schröter S., "A survey of video-based gesture recognition - stereo and mono systems", *Rapport Technique N° 693/1998*, FB Intormatik LS VII, Universität Dortmund, 1998, 20 pages.
- [Kramer 87]** Kramer J., Leifer L., "The talking glove: An expressive and receptive 'verbal' communication aid for the deaf, deaf-blind and nonvocal", Dans: *Proceedings of the Annual Conference on Computer Tech/Spec Ed/Rehab*, Northridge, CA, 1987, pp.335-340.
- [Kramer 96]** Kramer J.F., "The Talking Glove : hand-gesture-to-speech using an instrumented glove and a tree-structured neural classifying vector Quantizer", *Thèse de doctorat*, Université de Stanford, 1996, 197 pages.
- [Kryzstoforki 04]** Kryzstoforki K. et al., "Recognition of palm finger movements on the basis of EMG signals with the application of wavelets", *TASK Quaterly*, Vol. 8, N° 2, 2004, pp. 269-280.
- [Lamar 00]** Lamar M.V., Bhuiyan N., Iwata A., "Hand gesture recognition using T-combNET; a neural network model", *IEICE Transactions on information and Systems*, Vol. E83-D, N° 11, 2000, pp. 1986-1995.
- [Lee 96]** Lee C., Xu Y., "Online, interactive learning of gestures for human/robot interfaces", Dans: *IEEE International Conference on Robotics and Automation*, 1996, Vol.4, pp.2982-2987.

- [Lee 97] Lee C.S., Park G.T., Kim J.S., “Real-time recognition of Korean Sign Language based on elementary components”, Dans: *IEEE International Conference on Fuzzy Systems*, Barcelone, Espagne, Juillet, 1997, pp.1463-1468.
- [Lengellé 02] Lengellé R. et al. “Décision et reconnaissance de formes en signal”, Hermès Science Publications, Paris, 2002, 319 pages.
- [Leroi-Gourhan 64] Leroi-Gourhan A., “Le geste et la parole - Technique et langage”, Science d’aujourd’hui, collection dirigée par A. George, Albin Michel, 1964, 323 pages.
- [Liang 95] Liang R.-H., Ouhyoung M., “A real-time continuous alphabetic sign language to speech conversion VR system”, *Eurographics*, Vol. 14, N° 3, 1995, pp. C67-C76.
- [Liang 98] Liang R.-H., Ouhyoung M., “A real-time continuous gesture recognition system for sign language”, Dans: *International Conference on Automatic Face and Gesture Recognition*, Nara, Japon, Avril, 1998, pp.558-565.
- [Liddell 90] Liddell S.K., “Structure for representing handshape and local movement at the phonemic level”, Dans: *Theoretical Issues in Sign Language Research, Vol.1*, The University of Chicago Press, Chicago&London, 1990.
- [Loeding 04] Loeding B.L., Darkar S., Parashar A., Karshmer A., “Progress in automated computer recognition of Sign Language”, Dans: *International Conference on Computer Helping People with special needs*, Springer-Verlag, 2004, pp.1079-1087.
- [Ma 00] Ma J., Gao W., Wu J., Wang C., “A continuous Chinese Sign Language recognition system”, Dans: *International Conference on Automatic Face and Gesture Recognition*, 2000, pp.428-432.
- [Machover 90] Machover, T., “Flora”, *Disque Compact*, Bridge Records, Bridge 9020, extrait disponible à l’URL : <http://www.bridgerecords.com/9020.htm>, 1990.
- [MacKenzie 95] MacKenzie I.S., “Input Devices and interaction techniques for advanced computing”, Dans: *Virtual environments and advanced interface design*, Eds. Barfield W., Furness T.A., Oxford University Press, Oxford, Royaume Uni, 1995, pp. 437-470.
- [Martin 00] Martin J., “Reconnaissance de gestes en vision par ordinateur”, *Thèse de doctorat*, INPG, Juillet 2000, 238 pages.
- [McNeil 92] McNeil D., “Hand and Mind : what gestures reveal about thought”, The University of Chicago Press, Chicago, 1992, 416 pages.
- [Meyer 92] Meyer K., Applewhite H., Biocca F., “A survey of position trackers”, *Presence*, Vol. 1, N° 2, 1992, pp. 173-200.
- [Micera 03] Micera S. et al., “Functional assessment of hand orthopedic disorders using a sensorised glove: preliminary results”, Dans: *Proc. of the 2003 IEEE International Conference on Robotics & Automation*, Taipei, Taiwan, Sept, 2003, pp.2212-2217.

- [**Mitchell 97**] Mitchell Tom M., "Machine Learning", McGraw-Hill, 1997, 414 pages.
- [**Montes 03**] Montes S., Couso I., Gil P., "Fuzzy δ - ε -partitions", *Information Sciences*, Vol. 152, 2003, pp. 267-285.
- [**Murakami 91**] Murakami K., Taguchi H., "Gesture recognition using recurrent neural networks", Dans: *CHI'91: Human Factor in Computing Systems*, New-Orelans, Louisianne, 1991, pp.237-242.
- [**Ng 02**] Ng C.W., Ranganath S., "Real-time gesture recognition system and application", *Image and Vision Computing*, Vol. 20, 2002, pp. 993-1007.
- [**Oda 98**] Oda S. et al., "Development of a glove-type interface for data manipulation of the virtual environment in the operating room", Dans: *20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Hong-Kong, China, Oct, 1998, pp.1258-1259.
- [**Ouhaddi 98**] Ouhaddi H., Horain P., "Conception et ajustement d'un modèle 3D articulé de la main", Dans: *Actes des 6èmes journées du Groupe de Travail Réalité Virtuelle*, Issy-les-Moulineaux, France, 12-13 mars, 1998, pp. 83-90.
- [**Perng 99**] J.K. Perng et al. "Acceleration Sensing Glove (ASG)", Dans: *International Symposium on Wearable Computers*, IEEE, 1999, pp.178-180.
- [**Picone 90**] Picone J., *IEEE Automatic Speech and Signal Processing Magazine*, Juillet, 1990, pp. 26-41.
- [**Prillwitz 89**] Prillwitz S. et al. "HamNoSys Version 2.0: Hamburg Notation System for Sign Languages: An Introductory Guide", Signum Press, Hamburd, 1989.
- [**Quek 94**] Quek K.H., "Toward a vision-based hand gesture interface", Dans: *Virtual Reality and Software Conference*, 1994, pp.17-31.
- [**Rabiner 89**] Rabiner L.R., "A tutorial on hidden Markov models and selected applications in speech recognition", *Proceedings of the IEEE*, Vol. 77, N° 2, 1989, pp. 257-286.
- [**Rachev 85**] Rachev S.T., "The Monge-Kantorovitch mass transference problem and its stochastic applications", *Theory of Probability and its Applications*, Vol. 29, N° 1, 1985, pp. 647-676.
- [**Rachev 98**] Rachev S.T., Rüschemdorf L., "Mass transportation problems", Springer-Verlag, New-York, 1998, 430 pages.
- [**Ramamoorthy 03**] Ramamoorthy A. et al., "Recognition of dynamic hand gestures", *Pattern Recognition*, Vol. 36, 2003, pp. 2069-2081.
- [**Rekimoto 01**] Rekimoto J., "GestureWrist and GesturePad: unobtrusive wearable interaction devices", Dans: *5th IEEE International Symposium on wearable computers*, 2001, pp.21-27.
- [**Rosenberg 99**] Rosenberg R., Slater M., "The Chording Glove: A glove-based text input device", *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 29, N° 2, mois, 1999, pp. 186-191.

- [Ruspini 69] Ruspini E., "A new approach to clustering", *Information and Control*, Vol. 15, 1969, pp. 22-32.
- [Saporta 90] Saporta G., "Probabilités, analyse de données et statistique", Editions Technip, Paris, 1990, 493 pages.
- [Sawada 98] Sawada H., Notsu T., Hashimoto S., "Japanese Sign language recognition based on gesture primitives using acceleration sensors and datagloves", *International Journal of Virtual Reality*, Vol. 3, N° 4, 1998, pp. 1-12.
- [Starner 95] Starner T., Pentland A., "Visual Recognition of American Sign Language using hidden Markov models", Dans: *International Workshop on Automatic Face and Gesture Recognition*, Zurich, 1995, pp.189-194.
- [Steininger 01] Steininger S., Lindemann B., Paetzold, T., "Labeling of gestures in SmartKom - the coding system", Dans: *GestureWorkshop GW*, 2001, pp.215-227.
- [Stokoe 80] Stokoe W.C., "Sign language structure", *Annual Review of Anthropology*, Vol. 9, 1980, pp. 365-390.
- [Sturman 92] Sturman D.J., "Whole-Hand Input", *Thèse de doctorat*, Institut de Technologie du Massachussets, 1992, 233 pages.
- [Sturman 94] Sturman D.J., Zeltzer D., "A survey of glove-based input", *IEEE Computer Graphics and Applications*, Vol. 14, N° 1, 1994, pp. 30-39.
- [Su 00] Su M.C., "A fuzzy rule-based approach to spatio-temporal hand gesture recognition", *IEEE Transactions on Systems, Man and Cybernetics - Part C : Applications and Reviews*, Vol. 30, N° 2, Mai, 2000, pp. 276-281.
- [Takahashi 91] Takahashi T., Kishino F., "Hand gesture coding based on experiments using a hand gesture interface device", *SIGCHI-Bulletin*, Vol. 23, N° 2, 1991, pp. 67-74.
- [Toney 98] Toney A. "A novel method for joint motion sensing on a wearable computer", Dans: *International Symposium on Wearable Computers*, IEEE, 1998, pp.158-159.
- [Tsuboka 95] Tsuboka E., Nakahashi J., "Mathematical considerations and improvement on the fuzzy vector quantization hidden Markov model", *Electronics and Communications in Japan Part. 3*, Vol. 78, N° 11, 1995, pp. 9-22.
- [Valverde 97] Valverde L., "On the structure of F-Indistinguishability operators", *Fuzzy Sets and Systems*, Vol. 17, 1985, pp. 313-328.
- [Vamplew 96] Vamplew P.W., "Recognition of sign language using neural networks", *Thèse de doctorat*, University of South Australia, 1996, 219 pages.
- [Velten 98] Velten Th. et al., "Micro sensors for the application in gesture recognition", Dans: *6th Micro System Technologies 98, International Conference on Micro Electro Opto Mechanical Systems and Components*, Postdam, Dec 1998, pp.127-132.
- [Velten 99] Velten T., Stefan D., Obermeier E., "Micro-coil with movable core for

- application in an inductive displacement sensor”, *Journal of Micromech. Microeng.*, Vol. 9, 1999, pp. 119-122.
- [Vogler 98]** Vogler C., Metaxas D., “ASL recognition based on a coupling between HMMs and 3D motion analysis”, Dans: *International Conference on Computer Vision*, Mumbai, Inde, Janvier, 1996, pp.363-369.
- [Vogler 99.a]** Vogler C., Metaxas D., “Toward scalability in ASL recognition: breaking down signs into phonemes”, Dans: *Proceedings of the 3rd International Gesture Workshop*, Gif-Sur-Yvette, France, 1999, pp.211-224.
- [Vogler 99.b]** Vogler C., Metaxas D., “Parallel hidden Markov models for American Sign Language recognition”, Dans: *International Conference on Computer Vision*, Kerkyra, Grèce, 1999, pp.116-122.
- [Vogler 03]** Vogler C., Metaxas D., “Handshapes and Movements: Multiple-Channel American Sign Language Recognition”, Dans: *Proceedings of the 5th International Gesture Workshop*, Genova, Italie, 2003, pp.247-258.
- [Wachs 02]** Wachs J., Kartoun U., Stern H., Edan Y. “Real-time hand gesture telerobotic system using fuzzy C-means clustering”, Dans: *World Automation Congress*, 2002, Vol. 13, pp.403-409.
- [Wagner 74]** Wagner A.R., Fischer J.M., “The string to string correction problem”, *Journal of the Association for Computing Machinery*, Vol. 21, N° 1, 1974, pp. 168-173.
- [Waldherr 98]** Waldherr S., Thrun S., Romera R., “A neural-network based approach for recognition of pose and motion gestures on a mobile robot”, Dans: *5th Brazilian Symposium on Neural Networks*, Belo Horizonte, Brésil, IEEE Computer Society, Décembre, 1998, pp.79-84.
- [Weissman 99]** Weissmann J., Salomon R., “Gesture recognition for virtual reality applications using datagloves and neural networks”, Dans: *3rd IEEE International Joint Conference on Neural Networks*, 1999, pp.2043-2046.
- [Wheeler 03]** Wheeler K.R. “Device control using gesture sensed form EMG”, Dans: *IEEE International Conference on Soft Computing in Industrial Applications*, Binghamton, New-York, Juin, 2003, pp.21-26.
- [Wilson 01]** Wilson A., Bobick A., “Hidden Markov models for modeling and recognizing gesture under variations”, *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 15, N° 1, 2001, pp. 126-160.
- [Wu 99]** Wu Y., Huang T.S, “Vision-based gesture recognition : a review”, Dans: *Proceedings of the 3rd International Gesture Workshop*, Gif-Sur-Yvette, France, 1999, pp.103-115.
- [Young 02]** Yougn S. et al. “The HTK Book (for HTK Version 3.2.1)”, 2002, 355 pages.
- [Zacksenhouse 01]** Zacksenhouse M., Marcovivi P., “Interactive recognition of simultaneous manipulative hand movements”, *Mechatronics*, Vol. 11, N° 4, 2001, pp. 389-407.

-
- [Zadeh 71.a] Zadeh L.A., “Quantitative Fuzzy Semantics”, *Information Sciences*, Vol. 3, 1971, pp. 159-176.
- [Zadeh 71.b] Zadeh L.A., “Similarity relations and fuzzy orderings”, *Information Sciences*, Vol. 3, 1971, pp. 177-200.
- [Zimmerman 85] Zimmerman T.G., “Optical flex sensor”, *US Patent 4.452.291*, VPL Research, Nov, 1938.
- [Zimmerman 87] Zimmerman T.G. et al, “A hand gesture interface device”, Dans: *Proc. Human Factors in Computing Systems and Graphics Interface*, ACM Press, NY, Avril, 1987, pp.189-192.
- [Zwick 87] Zwick R., Carlstein E., Budescu D.V., “Measures of similarity among fuzzy concepts: a comparative analysis”, *International Journal of Approximate reasoning*, Vol. 1, 1987, pp. 351-356.
- [www.5dt.com] Fifth Dimension Technologies.
- [www.ascension-tech.com] Ascension Technologies Corporation.
- [www.essentialreality.com] Essential Reality Inc.
- [www.fakespacelabs.com] FakeSpace labs
- [www.immersion.com] Immersion Corporation.
- [www.isense.com] InterSense, Inc.
- [www.k-team.com] K-Team Corporation.
- [www.lightglove.com] LightGlove.
- [www.measurand.com] Measurand Inc.
- [www.polhemus.com] Polhemus.
- [www.sarcos.com] Sarcos Inc.
- [www.signwriting.org] Sign Writing Site.

