

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE DES FRERES MENTOURI
FACULTÉ DES SCIENCES DE LA TECHNOLOGIE
DÉPARTEMENT D'ÉLECTRONIQUE

THESE

Présentée pour obtenir le diplôme de

DOCTORAT EN SCIENCES

Spécialité : ELECTRONIQUE

Option : ELECTRONIQUE

Par :

BOUCHEMAL Billel

THEME

ANALYSE DE L'INTERACTION HOMME - ROBOT DANS LES SYSTEMES SUPERVISES MULTIMODAUX

Devant le Jury :

Président	Hammoudi Zoheir	<i>Professeur, Université de Constantine</i>
Rapporteur	Zaatri Abdelouahab	<i>Professeur, Université de Constantine</i>
Examineurs	Belarbi Khaled	<i>Professeur, Ecole Nationale Polytechnique, Constantine</i>
	Chemachema Mohamed	<i>Maitre de Conférences A , Université de Constantine</i>
	Mahfoudi Chawki	<i>Maître de Conférence A, Université Larbi Ben Mhidi, Oum El Bouaghi</i>

Année : 2015

Dédicaces

A mes parents,

A mes grands-parents,

A mes frères et sœurs,

A tous mes amis,

Remerciements

Avant tout, mes remerciements à mon dieu **ALLAH** le tout puissant qui m'a aidé à faire ce travail et qui m'a donné le courage, la patience et la santé durant toutes ces longues années d'étude afin que je puisse arriver à ce niveau.

Ce travail de recherche a été effectué au Laboratoire **LATA** « Laboratoire des Applications de la Technologie Avancé » de l'Université de Constantine, sous la direction scientifique de Monsieur le professeur **Abdelouahab ZAATRI**, que je tiens à remercier très vivement de son enthousiasme envers mon travail, de sa disponibilité et de son soutien scientifique et humain. Je le remercie également, pour la confiance qu'il m'a accordée, ses encouragements et ses précieux conseils.

Je suis très honoré que Monsieur **HAMMOUDI Zoheir**, Professeur à l'université de Constantine 1, ait accepté de présider le jury de cette thèse. Je remercie également Messieurs **BELARBI Khaled**, Professeur à l'Ecole Nationale Polytechnique de Constantine, **MAHFOUDI Chawki**, Maître de Conférence à l'université de Larbi Ben Mhidi à Oum El Bouaghi et **CHEMACHEMA Mohamed**, Maître de Conférence à l'université de Constantine, d'avoir accepté de participer à ce jury.

Je remercie aussi toute l'équipe du laboratoire LATA, pour leurs aides, serviabilités et leurs encouragements.

Mes remerciements s'adressent également à tous mes collègues de l'Unité de Recherche en Technologie Industrielle de Annaba, pour leurs soutiens, ainsi que Monsieur **H. MERABTI** le directeur de division Mécatronique et Monsieur **H. MERADI** directeur de l'unité de recherche en technologie industrielle à Annaba.

Mes remerciements s'adressent aussi à Monsieur **O. Khedraoui**, Maître de Conférence à l'université de Oum El Bouaghi, Monsieur **S. Mouissat**, Docteur à l'université de Constantine 1, Monsieur **Benchaker Yassin**, **Houssein Eddine BOULKOUR**, **D. Boucharma**, **H. Nezari**, **Da l'yazid**, **HARIK el Houssein Chouaib**, **A. Boussetla** et **Inel Fouad** pour leurs encouragements, à tous mes amis et tous ceux qui ont contribué de près ou de loin à la réalisation de ce modeste travail.

Enfin je ne pourrais pas terminer ces remerciements sans une pensée à l'ensemble de mes enseignants de toutes les phases scolaire qui sont à l'origine de tout mon savoir et qui m'apprirent à lire et à écrire.

Un grand merci à mes parents et à mes grands-parents.

Table des matières

Introduction générale	1
1 Introduction et Etat de l'art sur l'interaction homme-robot	6
1.1 Introduction	7
1.2 Définitions et terminologie concernant la multimodalité	7
1.3 Définitions et terminologie concernant l'interface graphique.....	9
1.4 Etat de l'art sur les techniques d'interaction haut niveau	11
1.4.1 La commande joystick	11
1.4.2 La commande vocale	12
1.4.2.1 Approche pour la reconnaissance de la parole	13
1.4.2.2 Logiciels existants pour le traitement de la parole	15
1.4.2.3 La bibliothèque VOCE	16
1.4.3 La commande gestuelle	16
1.4.3.1 Les systèmes de capture de gestes	18
1.4.3.2 Le geste graphique.....	19
1.4.4 La commande par pointage sur image.....	22
1.5 Etat de l'art sur les thèmes et l'analyse de l'interaction homme-robot	30
1.5.1 Quelques thèmes choisis dans le domaine de l'IHR	30
a) Le robot dans Le robot dans l'éducation, la thérapie, la réhabilitation et le soutien des personnes âgées	30
b) Interactions multimodales, l'expressivité et compétences de conversation	30
c) L'apprentissage social et l'acquisition de compétences par l'enseignement et imitation	30
d) Coopération et collaboration homme-robot.....	31
e) Identification des Intentions et compréhension des activités humaines	31
1.5.2 Exemples de l'état de l'art sur les études de l'IHR.....	31

1.6	Etat de l'art sur les robots à câbles.....	35
1.7	Conclusion	38
2	Les prototypes expérimentaux réalisés	40
2.1	Introduction	41
2.2	Le robot parallèle à câbles	41
2.2.1	Le robot parallèle à 3 câbles	42
2.2.1.1	Modélisation	42
2.2.1.1.1	Présentation de la structure du robot	42
2.2.1.1.2	Modélisation Géométrique	43
2.2.1.1.3	Modélisation Cinématique	43
2.2.1.1.4	L'analyse statique des forces	44
2.2.1.1.5	Modélisation Dynamique	46
2.2.1.2	Simulation du control du RPCs	48
2.2.1.2.1	La commande PD du robot	49
2.2.1.3	Développement de la plateforme d'étude	51
2.2.1.3.1	Description du fonctionnement	52
2.2.2	Le robot parallèle à 4 câbles	62
2.2.2.1	Modélisation.....	62
2.2.2.1.1	Modélisation Géométrique	62
2.2.2.1.2	Modélisation Cinématique.....	62
2.2.2.1.3	L'analyse statique des forces.....	63
2.2.2.1.4	Modélisation Dynamique.....	64
2.2.2.2	Simulation du control du Robot à 4 câbles	65
2.2.2.3	Réalisation expérimentale du robot parallèle à 4 câbles.....	68
2.2.2.3.1	La structure mécanique du robot.....	68
2.2.2.3.2	La partie électronique	70
2.3	Le robot mobile avec bras manipulateur.....	74
2.3.1	La réalisation mécanique	74
2.3.1.1	Le bras manipulateur	75
2.3.1.2	La base mobile	76
2.3.2	Modélisation géométrique du bras manipulateur.....	78
2.3.2.1	Modèle géométrique inverse	78
2.4	Conclusion	80
3	Implémentation des Modes de Contrôle Haut Niveau	81
3.1	Introduction.....	82

3.2	La commande vocale	82
3.2.1	Détection des limites d'un mot.....	82
3.2.2	La méthode MFCC (Mels Frequency Cepstrum Coefficients)	86
3.2.2.1	L'échelle des Mels.....	86
3.2.2.2	Création d'un cepstres à partir des coefficients MFCC	88
3.2.2.2.1	Prétraitement	89
3.2.2.2.2	Fenêtrage.....	89
3.2.2.2.3	Application d'une fenêtre de pondération.....	89
3.2.2.2.4	La FFT (Fast Fourier Transform)	90
3.2.2.2.5	Banc de filtres Mels.....	91
3.2.2.2.6	Les coefficients Cepstraux	92
3.2.3	Quantification vectoriel.....	93
3.2.3.1	Conception de l'algorithme LBG	94
3.2.3.2	Classifieur.....	95
3.2.3.3	Résultats expérimentaux	96
3.2.4	L'utilisation de la bibliothèque VOCE	97
3.2.4.1	Fichiers de grammaire	97
3.2.4.2	Post-traitement du texte reconnu	98
3.2.4.3	Références des classes pour la librairie VOCE.....	99
3.2.4.4	Les problèmes rencontrés	100
3.3	La commande gestuelle	100
3.3.1	Implémentation de la commande gestuelle	100
3.4	La commande par pointage sur image.....	105
3.4.1	Implémentation de la commande de pointage sur image	105
3.4.1.1	La procédure de calibration de la camera	108
3.4.1.2	Exemples d'applications.....	110
3.5	La commande joystick.....	113
3.6	Conclusion	114
4	La conception de l'interface multimodale Homme - Robot	115
4.1	Introduction.....	116
4.2	Le système interactif global.....	116
4.3	La conception centrée utilisateur appliqué au SMM.....	117
4.3.1	Quatre principes fondamentaux	118
4.3.2	Etapes du processus de conception centrée utilisateur.....	119
4.4	L'implémentation de l'interface graphique multimodale	119
4.4.1	L'environnement de développement Processing.....	120

4.4.2	L'implémentation des modes de commande dans l'interface graphique multimodale	121
4.4.2.1	L'implémentation de la commande vocale	121
4.4.2.2	L'implémentation de la commande joystick	122
4.4.2.3	L'implémentation de la commande pointage sur image.....	124
4.4.2.4	L'implémentation de la commande gestuelle	125
4.4.3	Les routines supplémentaires implémentées	126
4.4.3.1	L'initialisation automatique du robot	126
4.4.3.2	Le mode d'acquisition	128
4.4.3.3	Les zones graphiques	129
4.4.3.4	Le simulateur virtuel.....	131
4.4.4	L'interface multimodale globale.....	132
4.4.5	Développement de l'interface graphique pour le robot manipulateur sous Labview	133
4.5	Conclusion	137
5	Analyse des modes et leurs combinaisons	138
5.1	Introduction	139
5.2	L'analyse de chaque mode de command séparément.....	139
5.2.1	L'analyse du mode vocal.....	139
5.2.2	L'analyse du mode de pointage sur image	143
5.2.3	L'analyse du mode gestuel	147
5.2.4	L'analyse de mode joystick	152
5.3	L'analyse de combinaisons des modes	154
5.3.1	La combinaison du mode vocal avec le mode de pointage sur image	154
5.3.2	La combinaison du mode gestuel avec le mode basé sur image.....	157
5.4	L'analyse de la dépendance par rapport à l'environnement de programmation et le système expérimental utilisé	161
5.5	Conclusion	166
6	Évaluation des performances pour les systèmes de contrôle multimodaux supervisés	168
6.1	Introduction.....	169
6.2	Travaux en relation avec l'AHP	169
6.3	Aperçu sur les ISCS.....	171
6.4	Interfaces multimodale intégrées.....	172
6.4.1	IGU pour robots industriels	172

6.4.2	IGU pour le robot a câbles.....	173
6.4.3	Mesure des performances pour ISCS.....	175
6.4.3.1	Les mesures des performances et les critères.....	175
6.4.3.2	Des expériences d'illustration du processus d'apprentissage	175
6.5	Évaluation de la performance des tâches au moyen d'AHP.....	178
6.5.1	Exemple 1 (robot industriel)	178
6.5.2	Exemple 2 (Robot à 4 câbles)	183
6.6	Conclusion	187
	Conclusion générale	188
	Bibliographie	192

Table des figures

1.1	Les interfaces en ligne de commande.....	9
1.2	L'ordinateur Macintosh avec interface graphique	10
1.3	Joystick	12
1.4	Domaine de la communication parlée homme-machine	14
1.5	Principe de reconnaissance et d'apprentissage de la parole	15
1.6	L'architecture de la bibliothèque VOCE.....	16
1.7	Exemple de gants colorés application de l' MIT	19
1.8	Exemple de gants numérique.....	19
1.9	Sketchpad.....	20
1.10	Appareils d'entrée gestuelle plane classiques.....	21
1.11	Appareils d'entrée gestuelle.....	21
1.12	Le schéma du système de vision.....	24
1.13	Des images à partir de l'interface graphique utilisateur.....	24
1.14	Un exemple de tâche	25
1.15	L'interface graphique d'utilisateur.....	25
1.16	Le système daVinci	26
1.17	L'avion sans pilote "Predator".....	27
1.18	Une interface double écran utilisée	28
1.19	Unmanned Surface vehicles.....	28
1.20	Une capture d'écran pendant la commande de deux USV.....	29
1.21	Le rover Rocky 7.....	29
1.22	Paro, le robot phoque.....	32
1.23	Un groupe de personnes âgées interagissant avec le robot Paro.....	32
1.24	Le système d'enregistrement vidéo.....	33
1.25	Le robot BIRON (Bielefeld Robot Companion)	33
1.26	Interaction homme-robot dans l'environnement BITT	34
1.27	Caméra contrôlée par un mécanisme à câbles (Skycam)	35
1.28	Représentation d'un capteur de force pour les robots à câbles (CaTraSys)	36

1.29	Le system à câbles STRING-MAN.....	36
1.30	Le prototype mécanique de STRING-MAN.....	36
1.31	Interface haptiques de 6 degrés de liberté.....	37
1.32	Robot parallèle à câbles utilisé dans les centrales solaires.....	37
1.33	NeReBot est un robot à câble pour la réadaptation de membre supérieure avec 3 ddl.....	38
1.34	Le robot MariBot.....	38
2.1	Exemple de robot parallèle à câbles.....	41
2.2	Exemple d'un robot parallèle à 3 câbles.....	42
2.3	forces statiques pour le robot 3-câbles.....	45
2.4	Structure de poulie.....	47
2.5	Le schéma de contrôle en boucle fermé.....	50
2.6	Suivi d'une trajectoire circulaire.....	50
2.7	Les longueurs des cables.....	51
2.8	Les tensions appliquées sur les câbles.....	51
2.9	Schémas block de la plateforme d'étude du robot à 3 câbles.....	51
2.10	Les deux signaux de sortie d'un encodeur en quadrature de phase.....	53
2.11	Diagramme d'état des deux encodeurs pour les deux sens.....	53
2.12	L'implémentation des encodeurs en quadrature.....	54
2.13	Le block « QUADRATUR COUNTER ».....	54
2.14	L'implémentation du model géométrique direct.....	55
2.15	L'implémentation du contrôleur PD.....	56
2.16	Les interrupteurs de sélection de mode.....	57
2.17	Le block « Digital Output » de la librairie « Real Time Workshop ».....	57
2.18	Le simulateur virtuel 3D du robot parallèle à 3 câbles.....	58
2.19	Le simulateur Virtuelle 3D dans l'environnement VRML.....	58
2.20	L'interfaçage du simulateur virtuel avec Simulink.....	59
2.21	Suivi d'une trajectoire circulaire en 3D en mode statique.....	59
2.22	L'acquisition de la lettre « A ».....	60
2.23	La structure mécanique configurable.....	60
2.24	Le moteur EMG30.....	61
2.25	La carte de contrôle du robot à 3 câbles.....	61
2.26	Les sources d'alimentation du robot à 3 câbles.....	61
2.27	Le robot parallèle à 4 câbles.....	62
2.28	Forces statique du robot à 4 câbles.....	63
2.29	Le suivi d'une trajectoire circulaire.....	66
2.30	Les tensions appliquées sur les câbles.....	66

2.31	Les longueurs des câbles	66
2.32	Le suivi d'une trajectoire triangulaire	67
2.33	Les tensions appliquées sur les câbles.....	67
2.34	Les longueurs des câbles	67
2.35	L'effet de la vitesse sur la qualité du contrôle	68
2.36	La plateforme d'étude pour le robot à 4 câbles.....	69
2.37	Les configurations possibles du site robot.....	69
2.38	Effecteur final	70
2.39	La partie électronique du robot à 4 câbles	70
2.40	La Carte Mega ADK for Android	71
2.41	Conexion des PIN pour le circuit L298N	72
2.42	Le control bidirectionnel pour un moteur à CC	72
2.43	Le schéma électronique de l'interface de puissance sous PROTEUS.....	73
2.44	Le schéma électronique 3D et le circuit imprimé de l'interface de puissance sous ARES (ISIS)	73
2.45	Le prototype expérimental du robot série.....	74
2.46	Le bras manipulateur plus la base mobile.....	74
2.47	Le bras manipulateur avec pense	75
2.48	Le bras manipulateur sans pense.....	75
2.49	La pièce de fixation des servomoteurs	75
2.50	Le segment du bras.....	76
2.51	La pièce de fixation du pense prismatique	76
2.52	La base mobile de type « ROVER »	77
2.53	Une vue à l'intérieur de la base mobile	77
2.54	La pièce de renforcement + la fixation de la jante	77
2.55	Les roues utilisées	78
2.56	Le bras manipulateur du robot	78
2.57	Configuration du prototype, simulateur 3D.....	79
2.58	Simulation Matlab	79
2.59	Configuration du prototype, simulateur 3D.....	80
2.60	Simulation Matlab	80
3.1	Détection de début et de fin de mot.....	83
3.2	L'algorithme de détection de début et de fin de mot.....	84
3.3	Détection des limites des lettres « A » et « B »	85
3.4	Détection de début et de fin du mot « START »	86
3.5	Diagramme de Fletcher Décrit les seuils de perception et lignes d'isotonies selon les fréquences	87

3.6	La courbe de Mels	88
3.7	Représentation des étapes de création des MFCC	88
3.8	Courbe de la fonction de pondération de Hamming sur 128 points.....	89
3.9	Mise à l'échelle du résultat de l' FFT.....	90
3.10	La FFT du mot A.....	90
3.11	Spectre 3D du mot « A »	91
3.12	Représentation d'un banc de filtres dans l'échelle des Mels.....	92
3.13	La représentation 3D des coefficients Cepstraux du mot « A »	92
3.14	QV 1-dimensionnel	93
3.15	Un exemple de QV de 2-dimensions.....	94
3.16	Le Classifieur	96
3.17	Le mot START détecté traité et compressé	96
3.18	L'architecture de la bibliothèque VOCE.....	97
3.19	La configuration du robot parallèle pour la commande gestuelle.....	100
3.20	Le système de contrôle gestuel pour le robot à 4 câbles	101
3.21	L'algorithme de détection et poursuite des objets colorés.....	102
3.22	Le système interactif en mode gestuel.....	103
3.23	L'acquisition d'un cercle en mode gestuel avec contacte avec l'environnement.....	103
3.24	L'acquisition de la lettre « B » en mode gestuel avec contacte avec l'environnement	104
3.25	L'acquisition d'une commande en mode gestuel sans contact avec l'environnement	105
3.26	L'espace expérimental du robot a 4 câbles.....	106
3.27	Le système de pointage sur image pour le robot parallèle à câbles	106
3.28	Projection perspective	107
3.29	Grille de calibration.....	108
3.30	Montage de calibration	108
3.31	Différentes positions et orientations de la grille sur l'espace de travail.....	109
3.32	La projection des points de l'image sur les points de la grille.....	109
3.33	La reconstruction des prises de vue en utilisant le Toolbox CCT.....	110
3.34	L'état initiale du robot parallèle en mode pointage par image	111
3.35	La sélection d'un objet par le mode de pointage par image	111
3.36	La superposition de l'effecteur final sur l'objet sélectionné.....	112
3.37	Une sélection successive de 3 objets en utilisant le mode de pointage par image.....	112
3.38	Manette de jeu	113
3.39	Le mode continue	113
3.40	Le mode pas à pas	113

4.1	Le système interactif multimodal global	117
4.2	Les activités de conception centrée sur l'opérateur humain	119
4.3	L'environnement de développement Processing	120
4.4	L'implémentation de la commande vocale sur l'IHR	122
4.5	L'état reconnaissance du mot « seven »	122
4.6	L'implémentation de la commande joystick sur l'IHR	123
4.7	Les flèches d'indication de la commande joystick	123
4.8	Combinaison de direction par la commande joystick	123
4.9	La zone de la commande de pointage sur image dans l'interface multimodale	124
4.10	Objet sélectionné en utilisant la commande par pointage sur image	125
4.11	La zone graphique pour le mode gestuel	125
4.12	Un objet coloré sélectionné en mode gestuel	126
4.13	La localisation automatique de l'effecteur final sur l'espace du travail	127
4.14	Initialisation automatique de l'effecteur final	128
4.15	L'acquisition de la lettre « و » avec le mode acquisition	128
4.16	L'acquisition de la lettre « ح » en mode gestuel	129
4.17	La zone graphique en mode contrôle	130
4.18	La zone graphique en mode acquisition	130
4.19	Le simulateur virtuel 2D	131
4.20	L'interface graphique multimodale globale	132
4.21	L'interface graphique d'utilisateur pour commander le robot	134
4.22	La carte Arduino nano	134
4.23	Affectation des angles et espace de travail pour les servomoteurs	135
4.24	Figure 4.24- Implémentation de la scène 3D pour le simulateur virtuel	135
4.25	Figure 4.25- Implémentation du modèle géométrique inverse avec le block « Matlab script »	136
4.26	Figure 4.26- L'interface graphique après l'acquisition d'un geste graphique	136
4.27	Figure 4.27- Diagramme de sous-programme d'acquisition du geste graphique	137
5.1	L'interface multimodale pendant la détection du mot « voice »	142
5.2	L'interface multimodale pendant la détection du mot « Enable »	142
5.3	Objet sélection en utilisant le mode "point and click"	143
5.4	Le pointage d'un objet sélectionné	144
5.5	Le pointage successive d'un deuxième objet sur l'espace de travail	144
5.6	L'acquisition du nombre "2" en utilisant le mode "point and click"	146
5.7	L'acquisition du nombre "3" en utilisant le mode "point and click"	146
5.8	L'initialisation du mode gestuel	148
5.9	La sélection de l'objet coloré	148

5.10	L'acquisition du nombre "3" en utilisant le mode gestuel	149
5.11	L'acquisition de la lettre "ح" en utilisant le mode gestuel	150
5.12	L'acquisition de la lettre "B" en utilisant le mode gestuel	150
5.13	L'acquisition d'une commande gestuelle sans contacte avec l'environnement	152
5.14	La commande par joystick en mode pas à pas	153
5.15	La commande par joystick en mode continue	153
5.16	La détection de la commande vocale "find"	154
5.17	L'affichage du banque d'objets	155
5.18	La détection de la commande "goto"	156
5.19	Le pointage d'un objet en utilisant la commande vocale	157
5.20	La classe spéciale proposée	158
5.21	Une expérience pratique	158
5.22	La phase de l'enseignement par démonstration	159
5.23	Intervention dans un point de blocage	160
5.24	Intervention dans deux points de blocage	161
5.25	Le système interactif pour le bras manipulateur	162
5.26	L'interface multimodale après l'acquisition d'un geste graphique	162
5.27	Mode partagé	163
5.28	Mode combiné en parallèle	164
5.29	Mode indépendant	164
5.30	Mode inclusion	164
5.31	Le degré de complexité	165
5.32	Le modèle général de l'interface multimodale	166
6.1	Vue d'ensemble du premier système expérimental	171
6.2	Vue d'ensemble du deuxième système expérimental	172
6.3	L'interface multimodal	173
6.4	Le mode pointer et cliquer	174
6.5	Le mode gestuel	174
6.6	Le mode vocal et joystick	174
6.7	Montage expérimental (Kuka-361)	176
6.8	L'améliorations des résultats avec les expériences	176
6.9	Montage expérimental (robot à 4 câbles)	177
6.10	L'amélioration des résultats en fonction des expériences	177
6.11	Le modèle AHP	178
6.12	Une expérience illustrative	184
6.13	AHP pour le choix de l'opérateur	186
6.14	AHP pour le choix du système expérimental	186

Liste des tableaux

3.1	Dimensions des matrices MFCC et QV	96
3.2	Le taux de succès	97
3.3	Références des classes pour la librairie VOCE.....	99
3.4	Les attributs statiques privés.....	99
6.1	Valeurs des différents attributs (critères) pour différentes alternatives.....	179
6.2	Table de comparaison par paires.....	179
6.3	Evaluation pour les quatre attributs.....	180
6.4	Valeurs d'évaluation pour le temps d'achèvement.....	181
6.5	Valeurs d'évaluation pour le temps de télémanipulation	181
6.6	Valeurs d'évaluation pour le nombre d'interaction.....	181
6.7	Valeurs d'évaluation pour la facilité d'utilisation	182
6.8	Valeurs de l'indice aléatoire (RI)	182
6.9	Valeurs des différents attributs (critères) pour différentes alternatives.....	184
6.10	Evaluation pour les quatre attributs.....	184
6.11	Valeurs d'évaluation pour le temps d'achèvement.....	184
6.12	Valeurs d'évaluation pour le temps de télémanipulation	185
6.13	Valeurs d'évaluation pour le nombre d'interaction.....	185
6.14	Valeurs d'évaluation pour la facilité d'utilisation	185

Introduction générale

Le contrôle et la commande des systèmes supervisés modernes destinés à la conduite des missions et des tâches complexes requièrent différents niveaux d'interaction avec l'opérateur humain compte tenu des différentes situations rencontrées. Pour assurer de meilleures performances dans la conduite des tâches et des missions assignées; la stratégie de conception des systèmes d'interaction qui est classiquement centrée sur la machine où c'est l'opérateur qui doit s'adapter à celle-ci; est en train de changer de sens pour que la machine s'adapte à l'opérateur (centrée opérateur). Parmi ces types de systèmes supervisés modernes d'interaction Homme-Machine (IHM), nous avons les robots qui sont des agents munis de capacités de perception et d'action pour interagir avec divers environnements du monde physique. Leur utilisation a été généralisée dans les usines, mais aujourd'hui, ils ont tendance à se trouver dans différents endroits et même dans les domaines critiques comme la recherche et le sauvetage, l'art militaire, la détection des mines et bombes, l'exploration spatiale, la chirurgie, le transport, etc. Ces nouveaux domaines d'applications nécessitent une meilleure interaction avec l'utilisateur. Dans ce sens, le concept de proximité doit être pris en considération là où les robots et les humains partagent un espace commun. En 1912, I. Asimov avait déjà décrit les propriétés de l'interaction homme-robot (HR) et l'importance de l'interface HR: « une bonne interface doit être transparente de manière à ce que l'on puisse vivre en symbiose avec la machine, elle doit s'intégrer totalement dans le monde humain » [1]. Ils peuvent aussi coopérer et partager les mêmes objectifs en termes de réalisation des tâches et des missions. Cette interaction étroite a besoin de nouveaux modèles théoriques pour améliorer l'utilité des robots et d'autre part d'évaluer les risques et les avantages de ce nouvel «ami» introduit dans notre société moderne.

Avec le progrès des composants matériels et logiciels, la recherche dans ce domaine se concentre d'une part sur la plus sûre interaction physique mais aussi sur une interaction socialement acceptable. L'objectif est de construire une communication intuitive, et facile avec le robot par la parole, les gestes et les expressions faciales, etc. D'autre part, les robots modernes coopératifs ont besoin de plus de fonctionnalités en matière d'intelligence artificielle (smart robots) comme de reconnaître et classer les objets, localiser l'homme et

sécuriser la relation avec lui. Ces différents besoins ont rendu l'interaction HR un domaine de recherche difficile et multidisciplinaire à l'intersection de la psychologie, les sciences cognitives, les sciences sociales, l'intelligence artificielle, l'informatique, la robotique, l'ingénierie et l'interaction homme-ordinateur.

Le besoin de capacités dynamiques pousse l'IHR à s'inspirer de l'interaction Homme-Homme. Ce besoin requiert l'utilisation de plusieurs techniques d'interaction de haut niveau, qui ressemblent aux moyens naturels par lesquels un homme peut interagir et communiquer avec un robot. Heureusement; la technologie moderne permet d'envisager l'utilisation de la multimodalité et le développement des systèmes multimodaux comme solution pour améliorer l'IHM et les performances du système considéré. Mais si la multimodalité s'avère une solution nécessaire, elle pose des difficultés introduites par la complexité de l'interface qui augmente avec le nombre de modes d'interaction. Cette interface doit aussi faire face à la complexité des systèmes robotiques utilisés, des tâches et missions assignées.

En raison de la nature de cette interaction multimodale, les robots et les humains ont besoin de coordonner leurs activités dans le temps et l'espace en temps réel, tout en étant souvent «face-à-face». Les chercheurs dans ce domaine sont de plus en plus conscients que des nouvelles approches de coexistence doivent être formulées.

Une analyse préliminaire de ces systèmes a montré qu'il n'existe pas à ce jour un modèle général pour la conception de l'interaction multimodale dans les systèmes supervisés. Les systèmes développés restent dépendants de la machine, de l'application envisagée, et de l'environnement de travail.

Contexte et motivation de la Thèse

Dans ce contexte, le travail envisagé consiste à analyser et à développer une interaction multimodale homme-robot en vue de lever les limitations dans le domaine des systèmes supervisés. On se propose donc de concevoir et d'implémenter une interface utilisateur multimodale flexible conçue selon les principes de la conception centrée utilisateur pour contrôler des systèmes robotiques supervisés locaux ou distants. On se propose également de concevoir et réaliser quelques prototypes de systèmes robotiques expérimentaux qui formeront une plateforme d'études qui servira à tester l'interface HR en question. Le système global (interface multimodale et systèmes robotiques) nous permettront d'analyser l'interaction homme-robot, en termes de la dépendance par rapport à l'opérateur, au système opérationnel et à l'environnement.

On s'intéressera dans une première étape à l'évaluation expérimentale des performances du système global en fonction des différents éléments y compris le facteur humain, la tâche, et l'environnement. Dans la deuxième étape, on proposera une approche mathématique pour l'analyse et l'évaluation des performances de ces systèmes.

Les Objectifs

L'objectif de ce sujet de thèse est divisé en plusieurs étapes.

- La première étape consiste à effectuer une analyse de l'interaction homme robot dans les systèmes supervisés multimodaux en se basant sur l'état de l'art. Il est question d'étudier la nature et le nombre de modes à implémenter, l'influence des facteurs humains, les performances du système opérationnel à conduire, le degré de complexité de la tâche ou de la mission ainsi que celui de l'environnement. Le but est la conception d'un modèle d'une interface graphique multimodale utilisateur ou "Multimodal Graphical User Interface" "MGUI" qui se caractérisera par une grande flexibilité par rapport à l'utilisateur et une grande adaptabilité par rapport aux systèmes opérationnels utilisés et à l'environnement.
- la deuxième étape consiste à concevoir et réaliser une plateforme expérimentale qui permettra d'analyser expérimentalement les différents modes séparément ainsi que leurs enchainements. On considérera plusieurs systèmes robotiques pour tester l'adaptabilité et la flexibilité de cette interface multimodale.
- la troisième étape consiste à proposer une approche mathématique permettant l'analyse et l'évaluation des performances du système global en tenant compte des critères quantitatifs et qualitatifs faisant intervenir le facteur humain et la prise de décision.

Contributions

Les contributions majeures de notre travail sont :

- La réalisation d'un nouveau système robotisé qui est le robot parallèle à câbles contrôlé par des commandes de haut niveau dans le but de valider nos études dans le domaine de l'analyse de l'interaction homme-robot sur des systèmes physiques. Les robots parallèles à câbles sont un nouveau type de robots parallèles qui ont la particularité d'utiliser des liaisons constituées de câbles au lieu de vérins. Ces robots présentent des propriétés intéressantes. Munis d'une interface multimodale, ils ouvrent un axe de recherche pour des applications originales.

- Le développement d'une application originale qui est l'utilisation du robot à 4 câbles dans une classe spéciale d'un centre de réhabilitation et d'apprentissage tout en exploitant la multimodalité et les différents avantages pour ce type de robots: un grand espace de travail en comparaison avec les manipulateurs parallèles classiques; En plus, les robots à base de câbles ont une structure légère qui a de bonnes propriétés dynamiques; un rapport masse/ charge utile élevée et un coûts de construction réduit; une bonne portabilité; une structure facile à monter, démonter, stocker et transporter;
- L'implémentation individuelle de plusieurs modes de commande haut niveau: la commande vocale, la commande gestuelle, la commande par pointage sur image (Point-and-click) et la commande par joystick.
- Le développement d'une interface multimodale combinant les différents modes de commandes cités pour permettre à l'opérateur d'interagir avec les prototypes expérimentaux dans le but d'analyser l'interaction homme-robot ;
- L'analyse indépendante de chaque mode de commande dans le but d'effectuer une analyse complète de l'interaction multimodale homme-robot, cette analyse est complétée par l'analyse des combinaisons possible entre les différents modes on tenon compte les différents facteurs intervenants pondant la phase d'interaction et la phase d'exécution des ordres (facteurs humain, complexité de la tâche,...etc.). Nous avons mis en évidence les propriétés; les potentialités et les limites d'application ;
- L'application de la multimodalité à divers systèmes : pour un robot à 3 câbles, un robot à 4 câbles et à un robot sériel porté sur une base mobile. De plus, on effectuera l'analyse de la dépendance et de l'adaptabilité de l'interface multimodale par rapport au système opérationnel utilisé ;
- Evaluation des performances pour les systèmes supervisés multimodaux tenant compte du facteur humain et les paramètres qualitatifs et quantitatifs. Proposition et adaptation d'une méthode d'évaluation et de décision dénommée Analytical Hierarchy Process (AHP) pour permettre l'analyse de l'interaction multimodale en se basant sur la combinaison des différents facteurs, ce qui permet d'optimiser les performances du système global tout en garantissant l'adaptabilité par rapport à l'utilisateur, à la stratégie de combinaisons des modes et au système expérimental utilisé.

Organisation du manuscrit

Dans le chapitre 1, nous présentons les différentes définitions et la terminologie utilisées dans le domaine de l'IHR. Par la suite, nous consacrons une partie à l'état de l'art sur les techniques de contrôle haut niveau ainsi que les différents thèmes de recherche concernant l'interaction homme-robot. Un examen critique de l'état de l'art est présenté.

Le chapitre 2 est consacré à la présentation de la réalisation ainsi que le contrôle bas niveau des différents prototypes expérimentaux réalisés. Nous commençons par le robot parallèle à 3 câbles suivi par le robot parallèle à 4 câbles. A la fin, nous présentons la réalisation d'un bras manipulateur à 5ddl porté par une base mobile.

Dans le chapitre 3, nous présentons les différentes techniques de contrôle haut niveau implémentées suivi de tests préliminaires. On commencera dans l'ordre par la commande vocale, la commande gestuelle, la commande par pointage sur image et la commande par joystick.

Le chapitre 4 est consacré à l'implémentation de l'interface multimodale globale qui regroupe les différents modes de commande présentés ultérieurement et cela sur la base de la technique de conception centrée utilisateur.

Dans le chapitre 5, nous présentons une analyse critique de chaque mode de commande séparément ainsi que leurs enchainement et combinaisons en tenant compte les différents facteurs qualitatifs et quantitatifs. Ensuite, nous étudions la dépendance de l'interface multimodale par rapport à l'environnement de programmation ainsi qu'au système opérationnel utilisé.

Le chapitre 6 est consacré à l'adaptation de la technique AHP en tant que technique d'analyse et d'évaluation pour l'interaction homme-robot. Elle permet d'évaluer les performances du système global en tenant compte des critères qualitatifs et quantitatifs.

On clôturera notre thèse par une conclusion générale qui regroupera de façon succincte les résultats essentiels obtenus.

Chapitre 1

Etat de l'art sur l'interaction homme-robot

Sommaire

1.1	Introduction	7
1.2	Définitions et terminologie concernant la multimodalité	7
1.3	Définitions et terminologie concernant l'interface graphique	9
1.4	Etat de l'art sur les techniques d'interaction haut niveau	11
1.4.1	La commande joystick	11
1.4.2	La commande vocale.....	12
1.4.2.1	Approche pour la reconnaissance de la parole	13
1.4.2.2	Logiciels existants pour le traitement de la parole	15
1.4.2.3	La bibliothèque VOCE	16
1.4.3	La commande gestuelle.....	16
1.4.3.1	Les systèmes de capture de gestes	18
1.4.3.2	Le geste graphique	19
1.4.4	La commande par pointage sur image	22
1.5	Etat de l'art sur les thèmes et l'analyse de l'interaction homme-robot	30
1.5.1	Quelques thèmes choisis dans le domaine de l'IHR	30
1.5.2	Exemples de l'état de l'art sur les études de l'IHR.....	31
1.6	Etat de l'art sur les robots à câbles	35
1.7	Conclusion	38

1.1 Introduction

Dans ce chapitre, nous commençons par présenter les différentes définitions ainsi que la terminologie utilisée dans le domaine de l'IHR et des interfaces graphiques utilisateurs. Par la suite, nous présentons l'état de l'art concernant les techniques de contrôle haut niveau, en particulier, celles que nous allons développer ultérieurement; à savoir : la commande manuelle par joystick, la commande vocale, la commande gestuelle et la commande par pointage sur image (point and click). Nous mettrons en évidence les caractéristiques principales de chaque mode ainsi que quelques domaines d'applications remarquables relatifs à ce mode. Nous présentons aussi quelques travaux concernant les systèmes supervisés qui utilisent des interfaces multimodales remarquables par leurs originalités et leurs domaines d'applications spécifiques. Enfin, nous dégagons quelques thèmes de recherche concernant l'interaction homme-robot vers lesquels la recherche actuelle et future s'oriente.

1.2 Définitions et terminologie concernant la multimodalité

- Mode de communication

D'après le petit Robert, un mode est une norme particulière sous laquelle se présente un fait, s'accomplit une action. En grammaire, le mode est un trait dénotant la manière dont le locuteur présente le procès (on dit plus couramment action). Du point de vue système, un mode représente l'état dans lequel le système se trouve à un moment donné. D'autre part, un mode fait référence aux cinq sens de l'être humain : le toucher, l'ouïe, la vue, l'odorat, le goût (réception d'information), et aux différents moyens d'expression humains : le geste, la parole (émission d'information). Il définit la nature des informations servant pour la communication (mode visuel, mode sonore, mode gestuel etc.)[1, 2].

- Modalités

Une modalité est une forme concrète particulière d'un mode de communication. Par exemple, le bruit, la musique, la parole sont des modalités du mode sonore [3,4].

- Modalités actives

Une modalité d'entrée est qualifiée d'active quand l'utilisateur doit réaliser une action explicite avec un dispositif en vue de spécifier une commande au système comme la parole « microphone, langage pseudo-naturel ». De même dans les interfaces manipulables, la modalité « ordinateur de poche, langage gestuel » constitue une modalité active, l'utilisateur inclinant par exemple l'ordinateur de poche pour faire défiler une liste. Une modalité de sortie est dite active lorsque l'utilisateur doit effectuer une action explicite pour percevoir les informations véhiculées par la modalité, comme regarder l'écran d'un ordinateur de poche [5].

- **Modalités passives**

Une modalité est caractérisée de passive quand le dispositif qui la constitue ne requiert pas l'attention et d'action explicite de l'utilisateur comme la capture par un GPS de la localisation d'un utilisateur, la modalité passive correspondante étant décrite par le couple (GPS, localisation en données GPS). De nombreuses interfaces (interfaces sensibles au contexte et « perceptual user interfaces ») exploitent des modalités d'entrée passives afin de rendre l'interaction plus robuste et efficace. De même de nombreux systèmes de réalité augmentée sur supports mobiles reposent sur des modalités de sortie passives exploitant un casque semi-transparent (principe de l'attention minimale, crucial en situation de mobilité) [5].

- **Multimodalité**

La multimodalité reflète le caractère de multiplicité (préfixe multi) des modalités pour un système interactif. Un système est multimodal s'il dispose d'au moins deux modalités pour un sens donné, entrée ou sortie. Ainsi, le système est qualifié de multimodal en entrée (en sortie) si au moins deux modalités d'entrée (de sortie) sont disponibles [3, 4].

- **Média**

Synonyme de "médium", ce terme concerne les dispositifs physiques permettant d'interagir avec le système, en entrée et en sortie et concerne la capture, le stockage ou la présentation des informations [8, 6]. Il est donc défini par le type de capteur ou d'effecteur qu'il met en œuvre [7].

- **Multimédia**

Un système informatique multimédia est un système capable d'acquérir et/ou de restituer, par l'intermédiaire des médias, des informations de natures et/ou de formes différentes (parole, musique, image vidéo, etc.)[1, 9].

- **Système multimodal**

Théoriquement, un système informatique multimodal est un système capable d'intégrer plusieurs modes de communication. Cependant, on désigne également par ce nom tout système capable d'intégrer plusieurs modalités de communication (même s'il n'intègre qu'un seul mode)[10].

- **Système interactif**

Un système interactif est généralement composé de deux parties : l'interface utilisateur et le noyau fonctionnel. L'interface utilisateur est constituée des éléments logiciels et matériels qui sont mis en œuvre lors de la capture des entrées de l'utilisateur et lors de la restitution des sorties du système. Le noyau fonctionnel représente le système de traitement et de stockage de l'information [10].

- Interface Homme-Machine

Ensemble des dispositifs matériels et logiciels permettant à un utilisateur d'interagir avec un système interactif [11].

- Interaction Homme-Machine

Discipline englobant l'ensemble des aspects de la conception, de l'implémentation et de l'évaluation des systèmes interactifs. Ensemble des aspects d'interaction avec un système comme pousser un bouton, bouger la souris, déplacer les doigts, etc. [11].

1.3 Définitions et terminologie concernant les interfaces graphiques

Nous commençons par la définition des différents termes utilisés dans ce domaine comme (UI, NUI, GUI, HMI, HCI et CHM, DHM, IHM) et les types du langage d'interaction (pseudo naturel, naturel ou analogique).

- Historique sur les interfaces graphiques d'utilisateur

En informatique initialement, une **interface graphique d'utilisateur**, en anglais « GUI » « Graphical User Interface » est un dispositif de dialogue homme-machine, dans lequel les objets à manipuler sont dessinés sous forme de pictogrammes à l'écran, que l'utilisateur peut utiliser en imitant la manipulation physique de ces objets avec un dispositif d'acquisition de commandes. Ces interfaces ont été créées par les ingénieurs du Xerox PARC à la fin des années 1970, pour remplacer les **interfaces en ligne de commande** (figure 1.1) qui sont parfois difficiles à comprendre, puis développé par Apple avec l'ordinateur Macintosh commercialisé en 1984 (figure 1.2)[12].

```

romain@RSLAP15 ~
429 dpkg -l *ibm-lotus*
430 sudo dpkg -P ibm-lotus-notes
431 sudo dpkg -P ibm-lotus-notes-fixpack
432 sudo apt-get -y install libgnomeprintui2.2-0 ia32-libs ttf-xfree86-nonfree t1-xfree86-n
onfree
433 cd ..
434 wget http://usablesoftware.files.wordpress.com/2011/02/getlibs-all-deb.pdf
435 mv getlibs-all-deb.pdf getlibs-all-deb
436 mv getlibs-all-deb getlibs-all.deb
437 sudo dpkg -i getlibs-all.deb
438 sudo getlibs -p libgnomeprintui2.2-0 libgnomeprint2.2-0 libgnomevfs2-0 libgnomeui-0 lib
xkbfile1 libstartup-notification0 libsepol1 libselinux1 libgsf-1-114 libgsf-1-dev librsvg2-2 l
ibrsvg2-common libavahi-client3 libavahi-common3 libavahi-glib1 libbonoboui2-0 libcroco3 libdb
us-1-3 libdbus-glib-1-2 libgnome2-0 libgnomecanvas2-0 libgnome-keyring0 libgnome-menu2 libesd0
gtk2-engines libgnome-desktop-2-17 libmotif3 libmotif4 libgnome-desktop-3-0 libavahi-glib1 gt
k2-engines-oxygen gtk2-engines-aurora gtk2-engines-qtcurve gtk2-engines-murrine gtk2-engines-e
quinox getlibs -p liborbit2 libbonobo2-0 libgconf2-4
439 sudo getlibs -p alsa-base alsa-utils iproute
440 sudo ln -s /usr/lib32/i386-linux-gnu/libavahi-glib.so.1 /usr/lib32/
441 ll
442 cd Work/
443 ls
444 rm *.old
445 rm -rf ibm-lotus-notes-8.5.3.i586/
446 clear
447 ll

```

Figure 1.1- Les interfaces en ligne de commande



Figure 1.2- L'ordinateur Macintosh avec interface graphique

Dans notre étude, nous pouvons partager cette section des définitions des interfaces graphiques utilisateurs en deux catégories. La première catégorie s'intéresse aux types d'interfaces elles-mêmes. La deuxième s'intéresse à la relation entre l'homme et le système interactif.

➤ **Les définitions en relation avec l'interface**

- **IUN « Interfaces utilisateurs naturelles »**

L'interface utilisateur naturelle (abrégée NUI, en anglais, Natural User Interface), dite aussi Post-WIMP, est une interaction informatique émergente qui met l'accent sur les capacités humaines telles que le toucher, la vision, la voix et le mouvement (geste).

L'interface utilisateur naturelle vise à exploiter la richesse des capacités humaines dans ses modalités d'interactions classiques (capacités sensoriels) [13].

- **IU « Interface d'utilisateur »**

Une interface définit la frontière de communication entre deux entités, comme des éléments de logiciels, des composants matériels informatique, ou un utilisateur. Elle se réfère généralement à une image abstraite qu'une entité fournit d'elle-même à l'extérieur. Cela permet de distinguer les méthodes de communication avec l'extérieur et les opérations internes, et autorise à modifier les opérations internes sans affecter la façon dont les entités externes interagissent avec elle, en même temps qu'elle en fournit des abstractions multiples. On appelle aussi interfaces des dispositifs fournissant un moyen de traduction entre des entités qui n'utilisent pas le même langage, comme entre un être humain et un ordinateur [14].

➤ **Les définitions en relation avec la relation entre l'homme et le système interactif**

- **CHM « Communication Homme-Machine »**

La fonction de communication se résume à la manière de présenter des informations ou de comprendre des instructions. Cette fonction se situe dans une relation opérateur-tâche où la machine a un rôle collaboratif [15, 16].

- **DHM « Dialogue Homme-Machine »**

Est la fonction privilégiée par laquelle un opérateur peut à la fois surveiller, communiquer et conduire un système automatisé [15, 16].

- **IHM « Interaction Homme-Machine »**

Ensemble des dispositifs matériels et logiciels permettant à un utilisateur d'interagir avec un système interactif [15, 16].

1.4 Etat de l'art sur les techniques d'interaction haut niveau

Comme l'objectif final des études de l'interaction homme-robot est de concevoir des systèmes multimodaux qui intègrent les capacités naturelles d'interaction entre les humains. Les recherches dans ce domaine sont concentrées sur l'implémentation de plusieurs modes de commande tels que : le mode vocal, le mode gestuel, le mode neuronal, le mode joystick, ... etc.

Nous présentons dans la suite de ce paragraphe, un état de l'art sur les quatre modes de commande haut niveau disponibles dans notre interface multimodale.

1.4.1 La commande joystick

Un joystick est un périphérique informatique constitué d'un manche posé sur un socle. Des boutons presseurs sont également accessibles sur le manche ou le socle. L'utilisateur peut agir sur ce périphérique soit en bougeant le manche dans une direction, soit en pressant les boutons (figure 1.3).

Aujourd'hui, il existe plusieurs types de joystick selon l'application envisagée. Des joysticks adaptés aux simulateurs de vol, aux jeux vidéo et même des joysticks plus robustes pour les applications industriels comme : grues, équipements forestiers, véhicules spéciaux. On trouve aussi des joysticks interactifs, qui agissent sur l'opérateur à l'aide d'un système à retour de force (haptic devices) pour les applications d'interaction homme-machine [17].

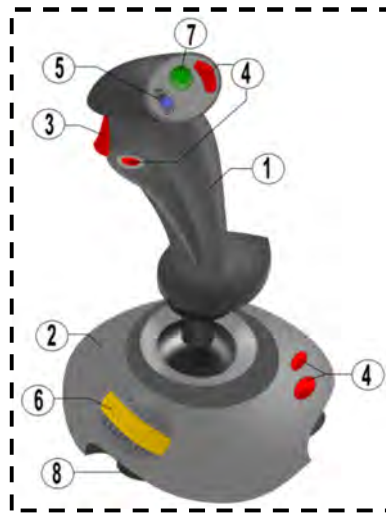


Figure 1.3- Joystick

Il s'agit d'un moyen de télémanipulation ou le joystick effectue l'interfaçage pour isoler l'opérateur des zones de travail dangereuses comme la manipulation des produits radioactifs dans les réacteurs, les zones inaccessibles comme les applications spatiales et la microchirurgie.

1.4.2 La commande vocale

La parole est un moyen naturel de communication entre les hommes. Les efforts entrepris depuis une soixantaine d'années pour mettre au point des systèmes de reconnaissance automatique de la parole ont obtenu un succès limité et ont surtout montré la nature complexe de la communication parlée. L'auditeur humain utilise plusieurs niveaux de traitement pour la reconnaissance de la parole. Le premier niveau est la détermination des caractéristiques du signal de parole lui-même, c'est-à-dire l'analyse acoustique. Viennent ensuite les niveaux phonétique, lexical, syntaxique, sémantique, etc. C'est à dire combien chez l'homme la reconnaissance et la compréhension de la parole sont intimement liées.

Avec l'arrivée des ordinateurs rapides vers les années 1960 et surtout depuis 1970, permettant en particulier le traitement du signal en temps réel, de gros efforts ont été déployés. Le lancement du projet DARPA (Defense Advanced Research Projects Agency) aux Etats-Unis en 1971 a été décisif en ce qui concerne l'évolution du traitement de la parole [18].

Les avantages de la reconnaissance automatique de la parole sont tellement importants que l'on trouve déjà sur le marché des dispositifs à une utilisation limitée, mais néanmoins efficaces. Citons certaines applications qui ont déjà vu le jour : saisie vocale de données, aide aux handicapés, chambre d'hôpital avec possibilités de commandes vocales pour le malade, commande vocale de robot, commande vocale d'une montre portable, etc.

La mise au point d'un système de reconnaissance automatique de la parole rencontre deux types de difficultés : les variations intra-locuteur et les variations interlocuteur. De telles variations sont dues à

la complexité physique de l'appareil vocal qui présente une soixantaine de muscles pouvant participer à la phonation [19].

Les variations intra-locuteur concernent un même locuteur, qui tout en prononçant un même mot peut parler de différentes façons d'une fois à l'autre. Ces variations ont pour causes : le soin apporté à l'articulation, la vitesse d'élocution, le niveau sonore, etc.

Les variations interlocuteur sont dues aux dimensions et proportions d'appareil vocal variables selon l'âge, le sexe et la taille, aux manières de réaliser tel ou tel son (voix) liées à l'anatomie et aux habitudes articulatoires de chaque individu.

Les systèmes qui existent actuellement peuvent reconnaître un ensemble de mots isolés, un ensemble de mots enchaînés et même de la parole continue, mais avec d'importantes limitations {vocabulaire limité, syntaxe rigide, locuteur consciencieux). Les techniques utilisées sont celles de la reconnaissance des formes [20, 21, 22].

Les grandes étapes de la reconnaissance de la parole sont résumées en quelques dates dans [23, 24].

Le domaine de la reconnaissance automatique de la parole (RAP) est divisé en quatre sous-domaines : la reconnaissance des mots isolés, la reconnaissance de mots enchaînés, la reconnaissance et compréhension de la parole continue avec un vocabulaire et une syntaxe limités et enfin, la reconnaissance et la compréhension du langage naturel. Seuls les trois premiers sous-domaines sont envisageables aujourd'hui. La reconnaissance et la compréhension du langage naturel par une machine est encore un rêve dans l'état actuel des connaissances [25].

Pour notre système, comme nous projetons de concevoir et de réaliser un système de commande vocal, nous avons choisi de travailler dans le domaine de la reconnaissance des mots isolés.

2.4.2.1 Approches pour la reconnaissance de la parole

Il existe deux approches permettant d'aborder la reconnaissance de la parole : l'approche globale et l'approche analytique [26].

Dans l'approche globale, l'unité de base est le mot : le mot est considéré comme une entité indivisible. Une petite phrase, de très courte durée, peut aussi être considérée comme un mot. Dans l'approche analytique, on tente de détecter et d'identifier les composantes élémentaires de la parole que sont les phonèmes. L'approche analytique permet de traiter de gros vocabulaires, alors que dans l'approche globale on est limité à des vocabulaires de petite ou moyenne taille.

Mais la méthode analytique a un grand inconvénient : l'extrême variabilité du phonème en fonction du contexte (effets de la coarticulation). C'est pourquoi la méthode globale a été choisie dans ce travail. Elle est décrite par la flèche colorée de la figure 1.4.

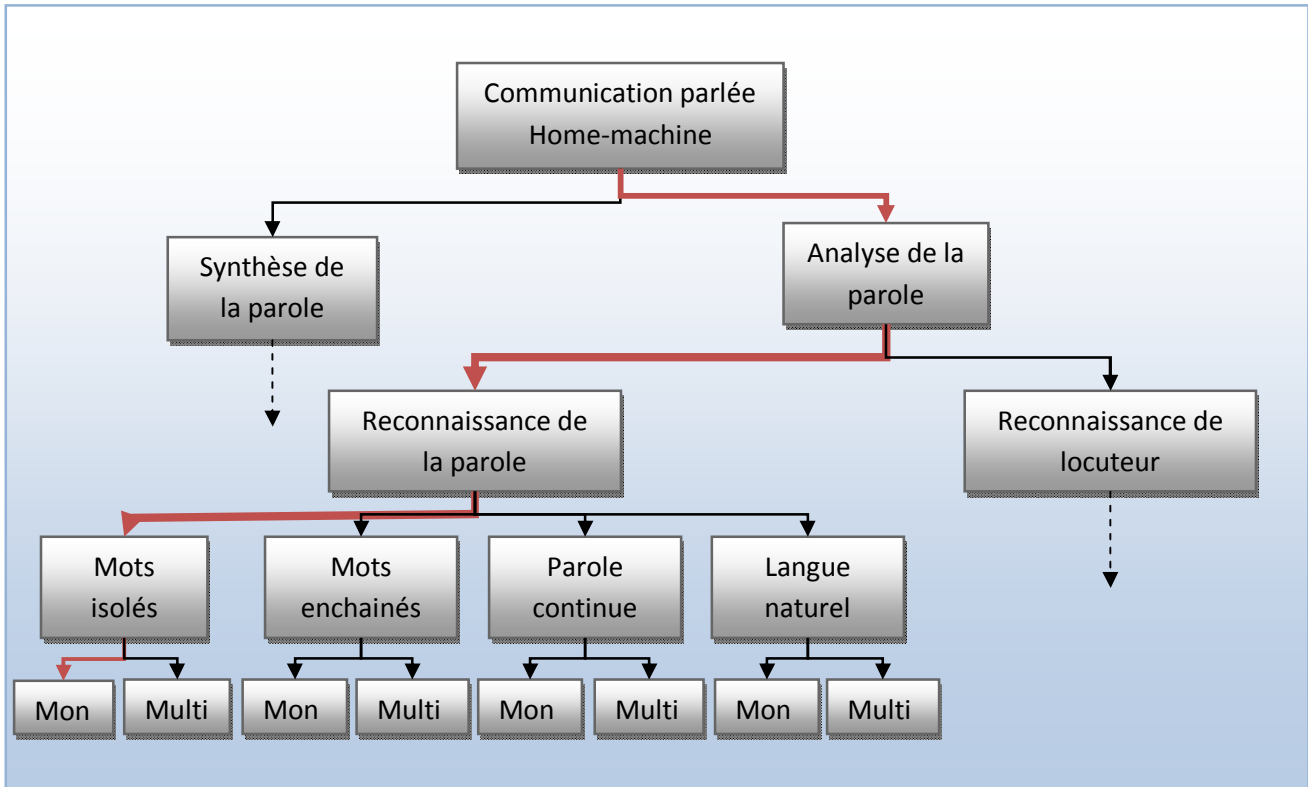


Figure 1.4- Domaine de la communication parlée homme-machine

Le principe utilisé est illustré dans la figure 1.5. Il comprend deux phases :

- la phase d'apprentissage : on crée pour chacun des mots une ou plusieurs formes de référence qui sont rangées dans un dictionnaire (mots de référence).
- la phase de reconnaissance : on identifie un mot inconnu à l'un des mots de référence rangés dans le dictionnaire.

Il y a deux opérations communes à la phase de reconnaissance et d'apprentissage : 1) analyse du signal acoustique, par exemple l'analyse spectrale par banc de filtres ; 2) traitement numérique {détection des limites du mot, compression temporelle, normalisation en amplitude et quantification}.

Le rôle du classifieur consiste à choisir le mot de référence le plus semblable au mot inconnu. Le choix est basé sur le calcul de la distance entre le mot inconnu et tous les mots de référence.

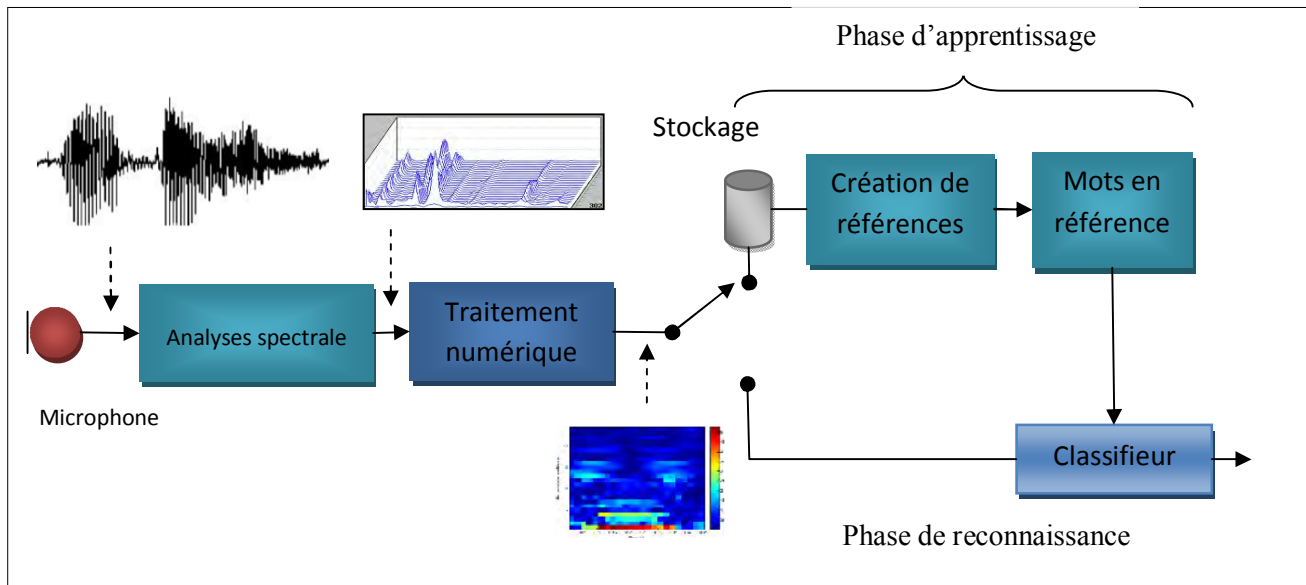


Figure 1.5- Principe de reconnaissance et d'apprentissage de la parole

2.4.2.2 Logiciels existants pour le traitement de la parole

Cette section aborde quelques-uns des outils logiciels les plus notables disponibles pour l'interaction vocale :

- 1- Dragon Naturally Speaking version 8 [27] est en âge mûr, produit commercialisé avec un kit de développement logiciel pour les développeurs d'applications. Toutefois, ce n'est pas gratuit, Open Source, ou disponible sur plusieurs plates-formes (actuellement, il fonctionne uniquement sous Windows).
- 2- Le Microsoft Speech SDK 5.1 [28] est une bibliothèque mature qui est disponible gratuitement. Il n'est disponible que sur Windows et n'est pas Open Source.
- 3- Le projet CMU Sphinx [29] a produit plusieurs bibliothèques gratuites, Open Source (Sphinx-2, Sphinx-3, et le Sphinx-4) destinés à traiter la reconnaissance vocale. Sphinx-4 est la version la plus avancée, conçue principalement comme une plate-forme de recherche avec des composants enfichables. Il est écrit entièrement en Java, ce qui rend facile à transporter sur de multiples plateformes.
- 4- FreeTTS [30] est un logiciel gratuit, bibliothèque pour la synthèse de la parole Open Source. Il travaille également sur plusieurs plates-formes, car il est écrit en Java.

Du point de vue développeur de logiciels, toutes les bibliothèques existantes ont des limites. Certains ne sont pas Open Source, dont certains sont conçus pour un système d'exploitation unique, certains n'ont pas l'API pour les développeurs, et la plupart ne gère pas la reconnaissance vocale et la synthèse à la fois. La bibliothèque la plus prometteuse est Sphinx-4, mais à plusieurs limites (par exemple le soutien d'un langage de programmation simple, pas de synthèse vocale intégrée, et une API complexe).

2.4.2.3 La bibliothèque VOCE

Cette bibliothèque exploite les points forts de logiciels existants. Voce fournit une couche mince entre les bibliothèques sous-jacentes d'interaction vocale et les applications [31].

L'architecture globale de la bibliothèque Voce est conceptuellement simple (figure 1.6). Pour la synthèse vocale, Voce prend des chaînes de texte à partir des applications et les transmet à un synthétiseur TTS qui les convertit en une sortie audio. Pour la reconnaissance vocale elle alloue un microphone Sphinx4 qui écoute en permanence les données audio entrantes du matériel audio de l'utilisateur. Un reconnaiseur Sphinx4 traite en permanence ces données, l'ajout de chaînes reconnues à une file d'attente qui peut être interrogée par l'application. La synthèse et la reconnaissance sont exécutées dans des threads séparés du thread principal de l'application pour éviter que l'application attende la fin de la synthèse ou de la reconnaissance.

La complexité de la technologie sous-jacente est cachée aux développeurs d'applications, ce qui leur permet de traiter l'interaction par la parole comme une simple entrée / sortie. Bas niveau support audio est déjà inclus dans Sphinx-4 et FreeTTS via son API Java, qui fournit un moyen pour faire face aux multiplateformes matériels audio.

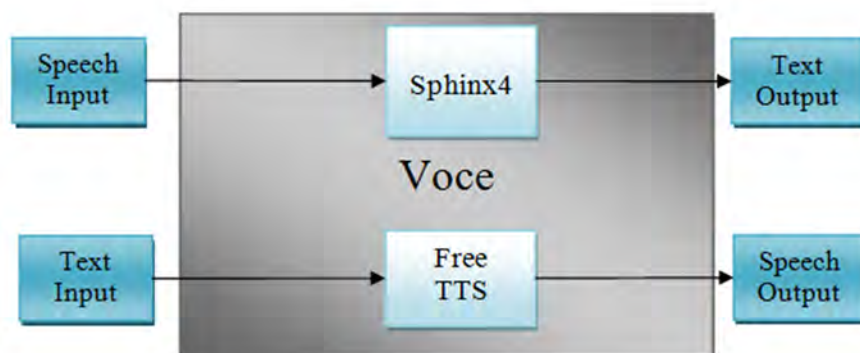


Figure 1.6- L'architecture de la bibliothèque VOCE

1.4.3 La commande gestuelle

Le geste est un moyen naturel de communication entre les humains, qui peut être effectué ou non simultanément avec la parole afin d'illustrer ou compléter la communication vocale (**geste co-verbal**). Dans notre époque, les machines sont contrôlées par des gestes en utilisant des interfaces mécaniques ou électroniques, par exemple : le volant de voiture, la commande d'un bras manipulateur à distance, ou simplement une souris d'ordinateur. C'est pourquoi de plus en plus de travaux de recherche s'intéressent à la vision artificielle pour l'acquisition et l'interprétation des gestes.

De plus, le canal de communication gestuelle fonctionne dans les deux sens, comme moyen d'émission et de réception d'informations. Si on prend l'exemple de la main, on distingue trois fonctions complémentaires et imbriquées [32] :

- **La fonction épistémique**

Dans ce cas, la main joue le rôle d'organe de perception. A l'aide du sens *tactilo-proprio-kinesthésique*, des informations sur l'état de surface, la température, le poids et même la forme ou les mouvements des objets peuvent être obtenues.

- **La fonction ergonomique**

La main joue le rôle dans ce cas d'un organe actionneur sur l'environnement physique. À l'aide d'un ensemble d'os et des muscles, nous pouvons appliquer des forces sur un objet pour provoquer une déformation ou un déplacement.

- **La fonction sémiotique**

Ici, la main joue le rôle d'un organe d'émission d'informations. Elle traite la perception visuelle d'un ou de plusieurs interlocuteurs.

Si aucun intermédiaire matériel n'est utilisé et la fonction sémiotique est utilisée, les fonctions ergonomique et épistémique sont utilisées surtout pour contrôler le mouvement effectué. Dans le cadre de l'analyse de l'interaction homme-machine, nous allons étudier particulièrement la fonction sémiotique.

Dans le domaine de l'interaction homme-machine, nous pouvons distinguer trois grandes catégories d'applications du geste en fonction du mode d'utilisation de l'organe gestuel, en entrée, en sortie ou en entrée-sortie :

- **Les gestes en entrée-sortie**

- Dans le domaine de robotique, les fonctions épistémique et ergonomique de la main humaine sont peut-être simulées à l'aide d'un robot articulé. Dans ce cas, les organes actionneurs permettent la manipulation des objets [33].
- Dans le domaine de la réalité virtuelle, le but est d'utiliser le geste de la main pour manipuler des objets virtuels [34]. Un bon aperçu des différentes problématiques dans le domaine de la réalité virtuelle est présenté dans un numéro spécial de IEEE Computer Graphics & Application, une définition et un historique de la réalité virtuelle est présenté dans [35], un modèle conceptuel basé sur des études psychologiques est présenté dans [36].
- Si nous combinons les trois fonctions, nous avons « **le geste instrumental** » [32]. Ce type de geste est appliqué par l'intermédiaire d'un équipement. Dans ce cas, nous avons une interaction physique entre l'opérateur et l'outil utilisé. Parmi ces outils matériels utilisés nous citons les touches rétroactives [37], ou l'exosquelette qui peut être attaché sur la main pour mesurer les flexions des doigts.

- **Geste en sortie**

Dans le domaine de l'animation de personnages virtuels, où le but est de réaliser des gestes à l'aide des personnes virtuels affichés sur des interfaces graphiques.

Dans [38] un système de création automatique des conversations animées a été implémenté, un système qui traduit des phrases de la langue japonaise en langage des signes est proposé dans [39].

- **Geste en entrée**

Nous pouvons distinguer trois grandes catégories :

- Les interfaces gestuelles, utilisées pour décrire des formes ou des commandes.
- Les applications multimodales.
- Les applications de reconnaissance de gestes des langues des signes.

Ces trois applications exploitent la fonction sémiotique du geste humain.

2.4.3.1 Les systèmes de capture de gestes

L'acquisition des gestes de la main pour les applications de l'interaction homme-machine peut être faite par l'intermédiaire de deux catégories de systèmes :

- **Les systèmes d'acquisition avec caméras vidéo**

Dans ce cas, nous avons besoin de localiser la main dans l'image reçue, puis d'estimer les paramètres 2D ou 3D comme les bouts des doigts et les angles des articulations. Dans le but d'analyser les gestes, on peut faire appel à des gants colorés (figure 1.7) ou des marqueurs sur la main [40, 41].

Ce qui concerne les méthodes d'analyses 2D, elles se limitent à la reconnaissance d'un nombre limité de configurations. Cette limitation est due à l'étape d'apprentissage. Par exemple, dans [42], ils ont développé une application de dessin qui fournit en temps réel la position d'un doigt par corrélation sur le bout du doigt.

En [43], ils ont proposé une approche par modèles déformables utilisant un modèle de distribution de points qui représente la silhouette d'une forme quelconque par un ensemble de points caractéristiques et des modes de variation qui décrivent les déplacements de ces points. Ce modèle est constitué à partir d'un ensemble d'apprentissage par une analyse en composantes principales des différences entre les formes de l'ensemble d'apprentissage et la forme moyenne. La reconnaissance est réalisée à partir du modèle de distribution des points. Cette démarche a également été proposée par Martin et Crowley [44].

Pour les méthodes d'analyse 3D, on remplace la connaissance à priori de l'ensemble des gestes attendus, par une connaissance à priori de l'objet observé « la main ». Donc, les techniques d'analyse 3D sont plus riches en termes d'informations gestuelles que les techniques 2D.

Nous notons que les méthodes d'analyse de geste par ajustement et recalage d'un modèle générique 3D utilisent différents types de modèles : articulés squelettiques [45], volumiques [46], ou de type surfaciques déformable [47].



Figure 1.7- Exemple de gants colorés application de l' MIT

- **Les systèmes avec gants numérique (instrumentées) :**

Les possibilités de communication par le geste avec les machines sont renforcées avec l'apparition des gants instrumentés (figure 1.8), équipés par de nombreux capteurs mesurant les angles des articulations des doigts et la position de la main dans l'espace. Ceux-ci ont été utilisés pour commander des acteurs virtuels, pour désigner et manipuler des objets graphiques [48, 49], ou même pour la reconnaissance de la langue des signes [50].

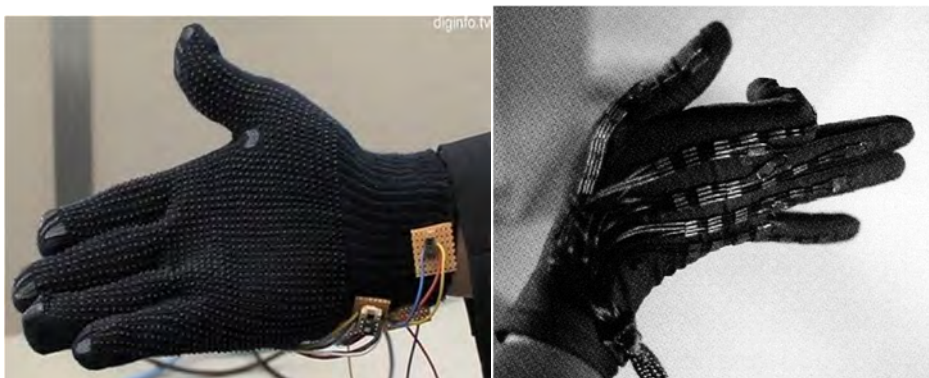


Figure 1.8- Exemple de gants numérique

Dans notre travail, nous nous intéressons à l'étude des méthodes 2D pour l'interaction gestuelle. Comme la métaphore papier-stylo des premiers ordinateurs à écran tactile, et la reconnaissance de l'écriture manuscrite sont à l'origine de l'introduction du geste graphique comme modalité d'interaction. Des études sont menées en vue d'exploiter largement l'interaction par le geste graphique, laquelle se révèle pertinente pour un large spectre d'activités.

2.4.3.2 Le geste graphique

Ce geste est celui qui laisse une trace sur un support, c'est le geste de l'écriture ou du dessin, par opposition au geste 3D (montrer du doigt, faire un signe de la tête ...) [51, 52, 53]. Encouragé par l'essor des Smartphones, et des tablettes tactiles, de plus en plus de systèmes interactifs reposent sur l'interaction gestuelle. La littérature concernant ce domaine étant

particulièrement vaste, nous nous restreindrons aux gestes effectués dans un plan et à la présentation des travaux de recherche majeurs qui leur sont consacrés.

Tout d'abord, l'utilisateur préfère manipuler l'appareil mobile avec une seule main (Karlson et al., 2008), l'autre main étant réservée pour interagir avec l'environnement, tenir un objet ou alors maintenir l'équilibre. De cette manière, seul le pouce est disponible, ce qui rend certaines cibles graphiques difficiles à atteindre et restreint davantage l'espace d'interaction pour effectuer des gestes.

- **Sketchpad (Sutherland, 1963)**

Il s'agit du premier système permettant à l'utilisateur d'entrer un geste, à l'aide d'un stylo optique, pour créer et modifier des formes géométriques par manipulation directe (figure 1.9).

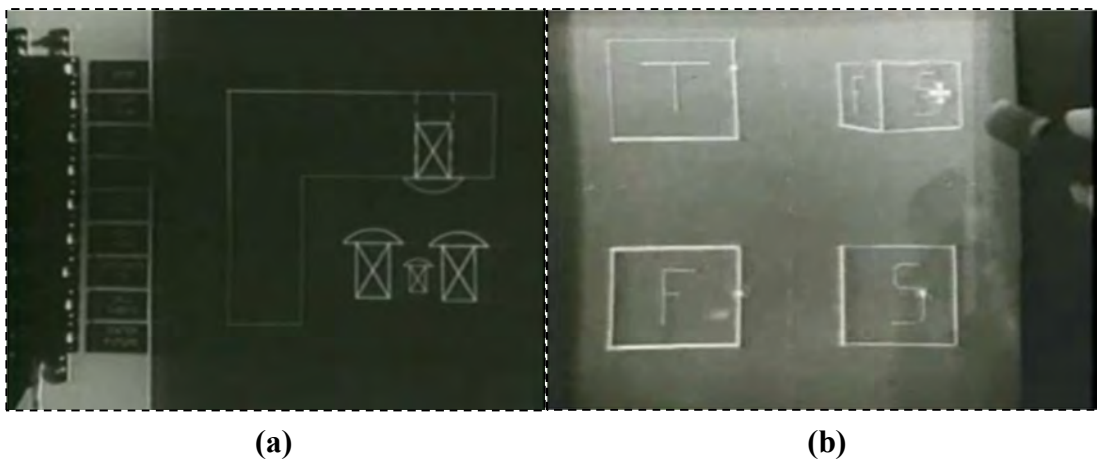
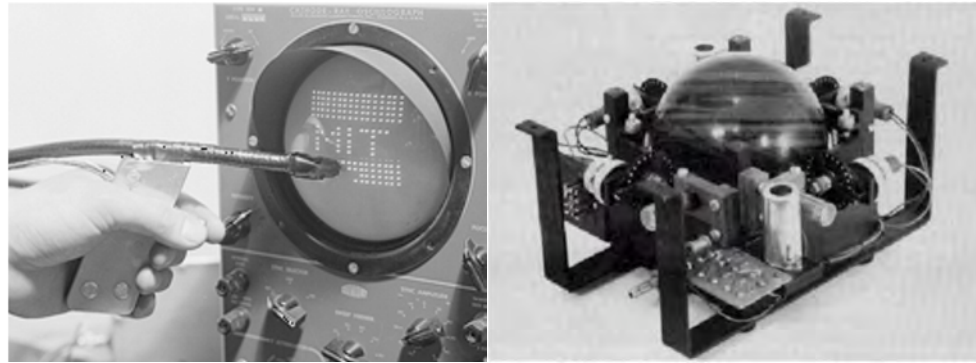


Figure 1.9- Sketchpad(a) Différentes instances d'un objet (ici un rivet) avec des dimensions et orientations variées. (b) Manipulation 3D et vues multiples sur un même objet.

Ce qui suit est un court historique permettant de situer dans l'ordre chronologique l'apparition de nouveaux appareils d'entrée gestuelle plane :

- Vers **1950**, le *Light Gun* (Everett, ~1950) fut le premier périphérique optique capable de lire la position d'un point sur un écran afin d'effectuer des diagnostics.
- En **1952**, le premier *Trackball* (Cranston et Longstaff, 1952) fut inventé et se présentait comme une grande souris inversée. L'utilisateur manipulait la boule, maintenue sur des coussins d'air pour minimiser les frictions, à l'intérieur du socle inamovible.
- En **1957**, le *Lightpen* (Gurley, 1957) adopte la forme d'un stylo et est le premier du genre à pouvoir être utilisé sur un écran.



(a) Light Gun

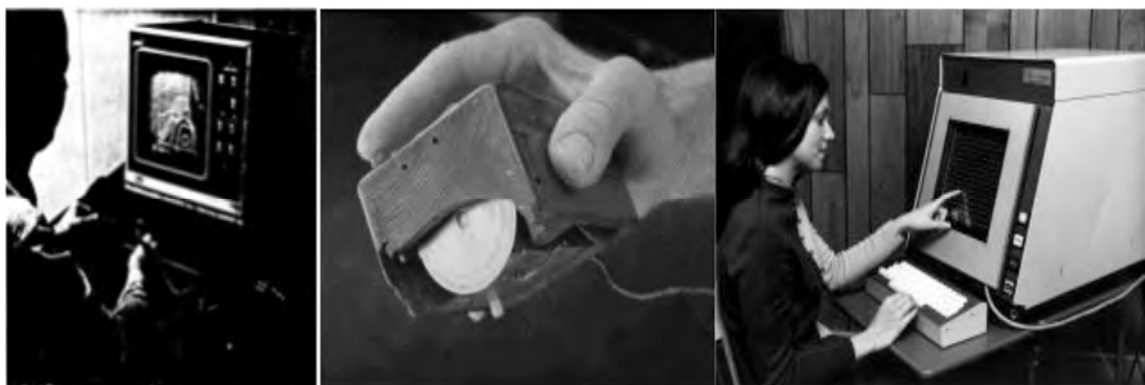
(b) Trackball



(c) Lightpen

Figure 1.10- Appareils d'entrée gestuelle plane classiques

- En 1963, arrive la « Rand Tablet » (Davis et Ellis, 1963), l'ancêtre des tablettes graphiques telles que nous les connaissons aujourd'hui.
- Dès 1964, la souris (English et Engelbart, 1967) fut inventée.
- En 1965, un mécanisme fonctionnel d'écran tactile capacitif est décrit (Johnson, 1965).
- En 1972, le premier écran tactile à être massivement distribué dans l'état d'Illinois voit le jour (Ebeling et al., 1972). Des lumières infrarouges réparties sur le cadre intérieur de l'écran permettant de détecter la présence d'un doigt sur l'écran.



(a) Rand Tablet

(b) Souris

(c) Touch Screen

Figure 1.11- Appareils d'entrée gestuelle

- En 1982, le premier appareil multi-touch (Mehta, 1982) est créé. Il repose sur l'emploi d'une caméra placée derrière une plaque de verre rétro éclairée filmant les ombres créées par la main de l'utilisateur. Le flux vidéo est alors numérisé et transmis à un processeur pour analyse.
- En 1983, ce qui semble être le premier appareil tactile sur le marché, le HP 150, est apparu (HP, 1983).
- En 1984, le premier écran multi-touch avec une nappe de détection transparente est inventée (Boie, 1984).
- Enfin, en 2009, ThinSight, le premier écran LCD doté de capteurs optiques sur toute sa surface (Izadi et al., 2009), permettant d'émettre et de capter de la lumière est réalisé. Il permet alors de lire et même scanner tout objet sur ou proche de la surface de l'écran.

1.4.4 La commande par pointage sur image

Les applications interactives homme-robot nécessitent plus d'autonomie et tendent à utiliser les moyens naturels de communication homme-homme. Dans ce but, il existe une sérieuse demande concernant la disponibilité de plusieurs interactions et techniques de contrôle haut-niveau tel que la télémanipulation, la commande vocale, la commande par image, et la commande geste. Parmi les modes cités ci-dessus, une attention particulière est réservée à la commande basée sur image. Ce mode de commande qui est souvent appelé « point and click ». Il opère seulement par pointage sur image reçue de l'espace de travail en utilisant la souris de l'ordinateur. Le système de pointage sur l'image convertit ce click en commande pour déplacer le robot vers cette emplacement de l'espace de travail réel en 2D ou 3D. Lorsque l'opérateur humain est en dehors de la boucle: ce mode de control peut être nommé « pointer et oublier » ou « point and forget ». Le mode de contrôle « point and click » présente les avantages suivants: Il est très facile à utiliser, demande un temps d'apprentissage minimum, et ne nécessite pas l'implication de l'opérateur dans la boucle de contrôle.

Il est aussi très efficace et est bien adapté pour les applications de la télé-robotique dans des environnements inconnus.

Plusieurs applications qui utilisent l'interaction homme-robot sont effectuées par l'intermédiaire d'un système de commande basé sur image. Dans le but de présenter un état de l'art sur l'utilisation de ce mode de contrôle haut-niveau, quelques travaux sélectionnés dans plusieurs domaines de recherche seront présentés.

Dans la robotique industrielle, Un exemple est le revêtement robotisé où un utilisateur distant spécifie la région de l'espace de travail à revêtir avec un système semi-automatique laser par un simple commande par pointage sur image sur la région d'intérêt telle qu'elle apparaît dans une image de l'interface graphique d'utilisateur [54].

L'automatisation des processus industriels économise une somme substantielle d'argent en réduisant la quantité de travail nécessaire à la production, ou en diminuant le nombre de pièces défectueuses rejetées. En général, les robots dans ce domaine sont utilisés avec un mode de programmation par apprentissage qui est souvent appelé « enseigner-répéter » ou « teach-repeat ». Le mode "enseigner-répéter" désigne un processus dans lequel l'opérateur manuellement, et parfois avec l'aide d'un logiciels, enseigne une série de tâches conjointes pour les robots. Lorsque chacune de ses tâches est exécutée successivement, la séquence fournira la trajectoire désirée (mission ou tâche globale).

Le processus d'apprentissage de la trajectoire désirée du robot peut prendre plusieurs heures à plusieurs jours avant d'être maîtrisé par le robot. Dans certains cas, le paradigme «enseigner-répéter" pour commander le robot pour une tâche donnée devient extrêmement coûteux. Par exemple, si la géométrie des pièces en cours de traitement change fréquemment ou si la pièce ne peut être repositionnée identiquement par rapport à son prototype. Le temps et les ressources impliqués dans l'enseignement de la tâche affectée au robot et la construction des appareils appropriés pour fabriquer la pièce peuvent représenter un pourcentage important du coût global de la production de la pièce. Dans l'exemple suivant, nous allons présenter un cas d'application où le mode de commande par pointage sur image peut servir à optimiser le processus de programmation par apprentissage.

Un exemple de tâche industrielle qui nécessite une dextérité mécanique et un niveau de précision élevée pour le positionnement relative en trois dimensions est le « plasma-spray operation » ou « l'opération de pulvérisation par plasma ». Une solution est proposée dans [54] pour résoudre les problèmes de la méthode « teach-repeat ». Le problème de l'orientation et l'emplacement de la pièce à usiner peut s'effectuer arbitrairement dans l'espace de travail pour la tâche de revêtement par plasma. Dans cette article, une technique de contrôle de haut-niveau basée sur un système de vision équipé d'un pointeur laser qui utilise la méthode « camera-space manipulation » « manipulation de l'espace de la caméra » est présentée (figure 1.12). Nous déduisons après étude de ce mode « camera-space manipulation » est en fait le mode "point and click". L'opérateur à l'aide d'une interface graphique d'utilisateur sélectionne la région d'intérêt par un simple clic de la souris sur l'image reçue de la pièce arbitrairement située et orientée par rapport au robot et au système de vision (figure 1.13). Elle fournit un niveau élevé de précision dans le positionnement et l'orientation du manipulateur par rapport à la pièce à usiner de géométrie qui peut être inconnue et située arbitrairement par rapport au robot et aux capteurs de vision. Dans ce cas, la méthode « teach-repeat » qui est une méthode manuelle, est remplacée par la technique de contrôle haut-niveau "point and click ».

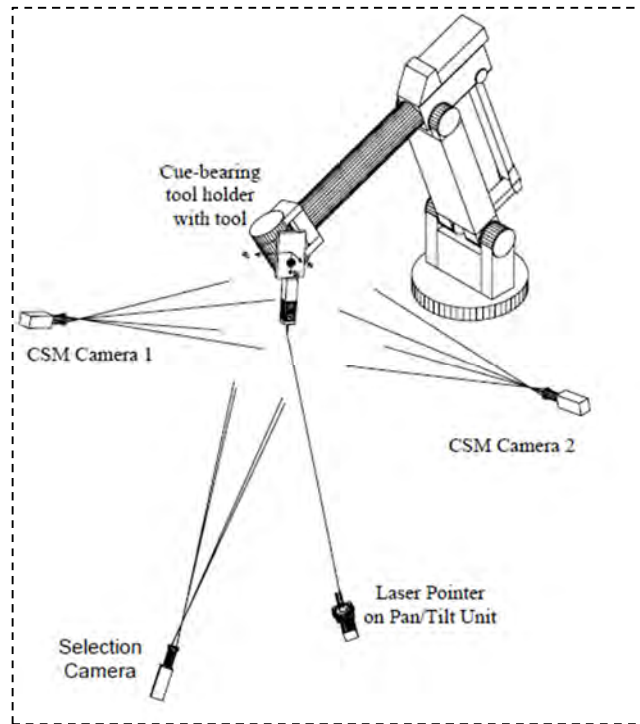


Figure 1.12- Le schéma du système de vision

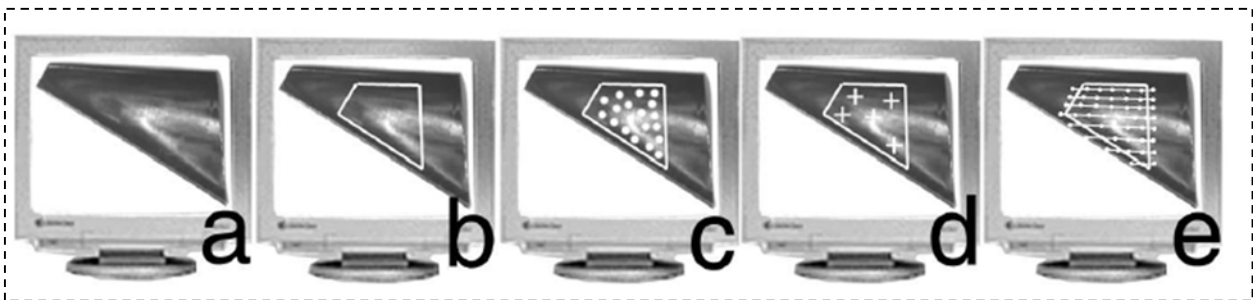


Figure 1.13- Des images à partir de l'interface graphique utilisateur

Un autre exemple d'application de la technique « point and click » est présenté dans l'article [55]. Le système présenté met en œuvre le concept d'autonomie partagée (shared autonomy) entre le robot et l'opérateur humain pour aider les pompiers dans les applications de sauvetage. Deux modes d'interaction ont été mis en œuvre pour commander le robot, à base d'images d'une caméra et d'une vue d'une carte de l'environnement locale présentée online « online local map », via une interface d'utilisateur graphique. Le premier est (i) « pointer and click » « point-and-click » sur la carte ou dans l'image pour indiquer les emplacements où le robot devrait aller. (ii) « point-and-click » sur les objets intéressants vus dans l'image de la caméra pour les inclure dans la carte par « online local map ».

La tâche présentée dans ce document concerne la situation où un pompier doit commander et guider à distance un robot de sauvetage afin d'identifier et de localiser les tubes de gaz dans un environnement intérieur après l'incendie (figure 1.14).



Figure 1.14- Un exemple de tâche: commander à distance le robot (a gauche) pour identifier et localiser les tubes de gaz présentant un risque d'explosion (a droite)

L'utilisateur est présenté avec l'interface représentée sur la figure 1.15. La fenêtre à gauche affiche la carte construite par le robot; l'opérateur a cliqué sur un point dans le couloir gauche (point rouge) pour donner au robot la consigne d'entrer dans ce couloir. La fenêtre à droite montre les images prises par la caméra du robot. L'utilisateur peut pointer-et-cliquer sur cette fenêtre pour indiquer l'emplacement où le robot devrait aller. Le point cliqué est traduit en coordonnées globales et inclus dans la carte de navigation du robot. L'utilisateur aura la possibilité de cliquer sur les objets spécifiques dans l'image afin de les introduire dans la carte globale de l'espace de navigation.

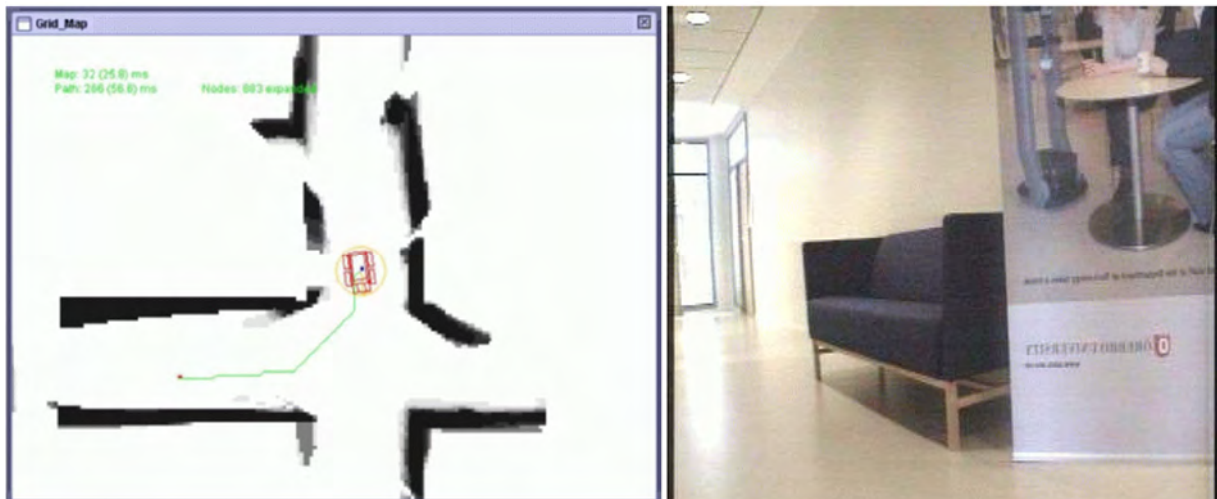


Figure 1.15- L'interface graphique utilisateur

Un autre champ d'application de la technique de pointage sur image est la robotique chirurgicale [56]. Les robots chirurgicaux ont commencé à apparaître sur le marché au cours des dernières années. Ils ont commencé à remplir les salles d'opération dans les grands centres médicaux. Nous prévoyons que les robots deviendront des outils standards de la salle d'opération. Les demandes

initiales d'une poignée de robots ont déjà démontré leur potentiel chirurgical. Comme la technologie évolue, les robots pourraient non seulement améliorer les performances dans des procédures minimalement invasives, mais aussi améliorer d'autres procédures existantes, ou même permettre entièrement de nouveaux types de chirurgies.

Les principaux avantages de la manipulation robotique d'outils chirurgicaux sont généralement : a) un enregistrement précis aux images médicales; b) mouvement cohérent sans fatigue et vibration, c) la capacité de travailler dans des environnements d'imagerie hostiles aux chirurgiens humains, et d) la capacité à repositionner les instruments rapidement et avec précision grâce à des trajectoires complexes ou sur des cibles multiples.

En robotique chirurgicale la tâche peut soit être prédéfinie par le chirurgien sur la base de données d'intervention / préopératoire ou, pour des procédures plus complexes, être définie comme la chirurgie progressive dans la salle d'opération. Sur la base de cette distinction, les robots chirurgicaux peuvent être classés en deux groupes principaux: Les systèmes guidés par image (Image-Guided Systems) et les systèmes pilotés par chirurgien (Surgeon-Driven Systems) (1.16). Ces catégories font appel tous deux à l'utilisation des compétences complémentaires / avantages du chirurgien et du robot mais ils le font de différentes manières. L'idée des systèmes de robotique guidés par l'image est de permettre au chirurgien de "pointer et cliquer" sur une image médicale à un emplacement à l'intérieur d'un organe, d'approuver son choix et engager le robot pour placer un outil (une aiguille par exemple) à la position physique de la position sélectionnée sur l'image médicale.

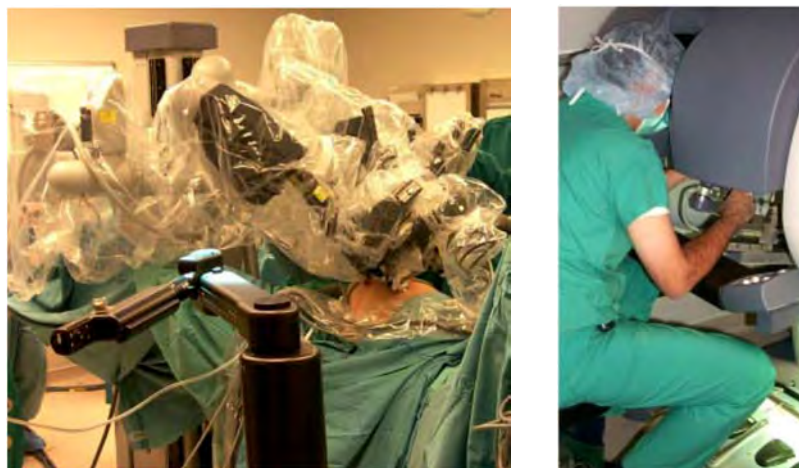


Figure 1.16- Le système daVinci (background - à gauche, et la console de chirurgien - droite)

Un autre champ d'application de cette technique de pointage sur image concerne les systèmes sans pilotes. Au cours des dernières années, il y a une forte augmentation de la demande des systèmes aériens sans pilotes pour une grande variété d'applications. Une classe particulière de ce type de véhicule qui a acquis une grande popularité dans la communauté est l'UAV (Unmanned Aerial Vehicle) (figure 1.17). Petits et facilement portables, elles sont idéales pour les applications civiles et

militaires allant de la surveillance à courte portée pour les études environnementales à des natures dangereuses pour des applications de recherche et de sauvetage [57].



(a) La station de contrôle

(b) Predator MQ-1

Figure 1.17-L'avion sans pilote "Predator"

Une étude sur l'influence des facteurs environnementaux sur l'**état de vigilance** « situation awareness » dans les missions des avions sans pilote est présenté dans [58]. L'interface utilisateur pour le contrôle du Drone, comme on le voit sur la figure 1.18, est composée d'un seul PC portable dual-écran connecté via Ethernet à un serveur de données acquises en temps réel ou préenregistrées. Sur un écran, en temps réel le flux vidéo est acquis par l'UAV. Sur l'autre écran, une fenêtre de la carte affiche une image aérienne au-dessus la zone d'essai. L'emplacement actuel du drone, tel que mesuré par le GPS embarqué est représenté graphiquement par un cercle rouge sur la carte, et l'historique de l'emplacement précédente est visible dans la piste bleue. Dans la fenêtre de la carte, les utilisateurs peuvent ajouter, déplacer et supprimer des cibles à l'aide de simples opérations de type point-and-click à l'aide de pointeurs (souris).

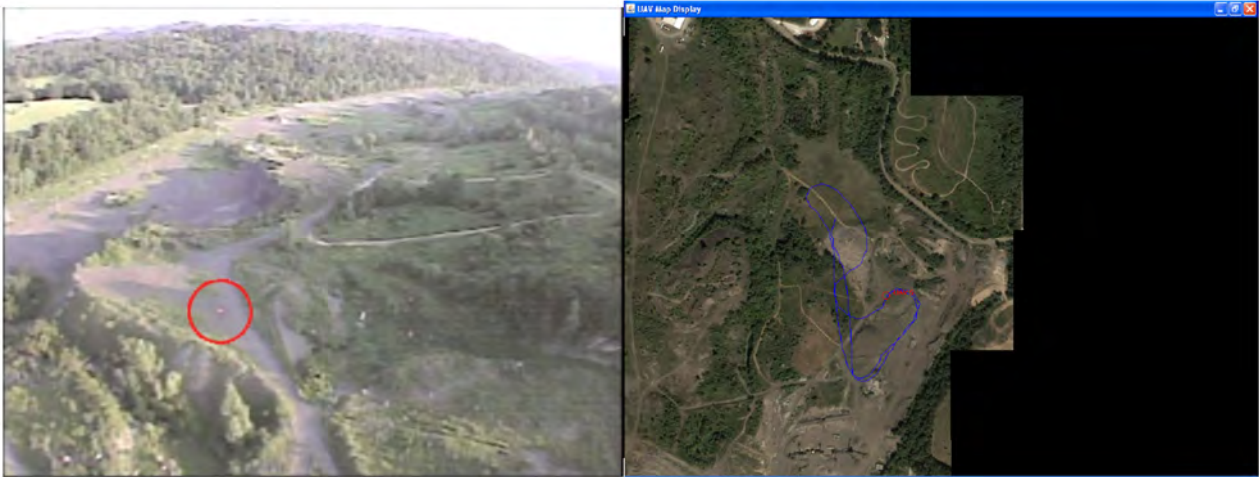


Figure 1.18- Une interface double écran utilisée, dans l'écran de gauche le flux vidéo à partir de l'UAV, dans l'écran de droite une carte avec des images aériennes du site de test recouverte avec la trace GPS du drone.

La technique de pointage sur image est presque inévitable pour les systèmes robotique télécontrôlés. Les véhicules terrestres sans pilote ou UGV (Unmanned Ground Vehicle) sont un exemple de ces systèmes (figure 1.19.a). Elles utilisent une trajectoire de déplacement créé par la concaténation de plusieurs points sélectionnés par l'opérateur. Comme remarqué dans [59], une grande partie de la technologie développée pour les UGV est applicable directement sur les véhicules de surface sans pilote « USV » ou " Unmanned Surface Vehicles" (Figure 1.19.b).



(a) MPRS URBOT w/ GPS Evaluation Antenna (a) SSC-San Diego Unmanned Surface Vessel

Figure 1.19- Unmanned Surface vehicles

Cette technique de pointage sur images est bien adaptée pour la création et la planification de trajectoires en utilisant les via-points sur des cartes. La figure 1.20 est une image de MOCU « The Multi-robot Operator Control Unit » configuré pour être utilisé avec l'USV. On constate sur la carte de la figure 1.20 la création d'une trajectoire à travers les points intermédiaires (via-points) colorés en vert.

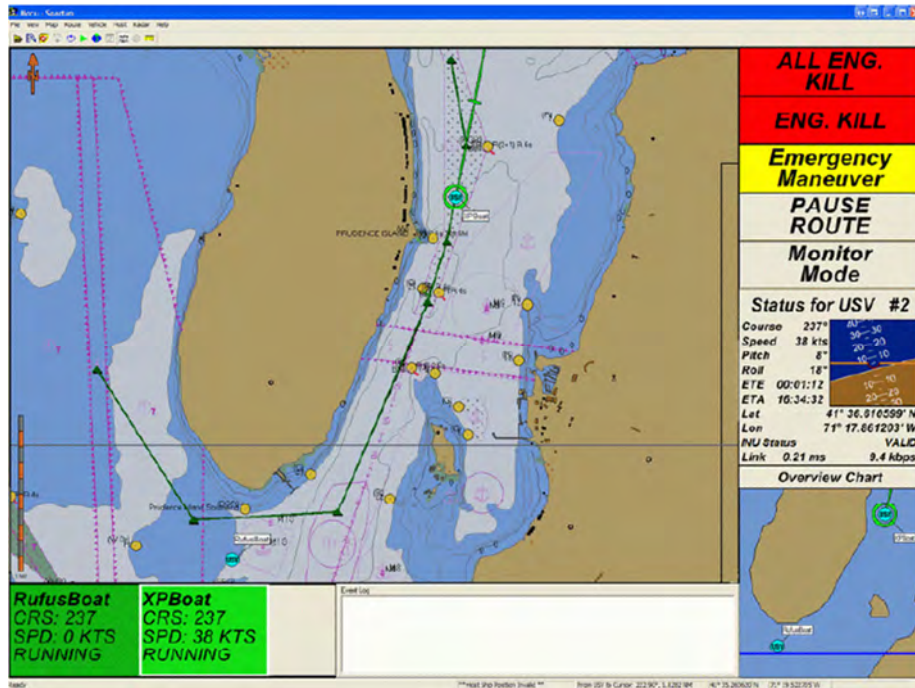


Figure 1.20- Une capture d'écran pendant la commande de deux USV

Comme l'exploration de Mars tient une place importante dans les programmes scientifiques d'exploration du système solaire. Un projet appelé « Le retour d'échantillon et le réseau martien » ou « **Mars Sample Return** » a été lancé par la NASA (figure 1.21). Ce projet a pour objectifs d'analyser des échantillons du sol martien dans des laboratoires sur Terre. L'utilisation de type spéciale de robots mobiles appelés « Rover » dispose d'un système de prélèvement permettant d'extraire des roches. Dans ce projet, l'opérateur dans la salle de commande terrestre utilise le mode "point and click" pour désigner les parties des roches à extraire ou pour engager le robot de se déplacer vers un endroit donné [60].

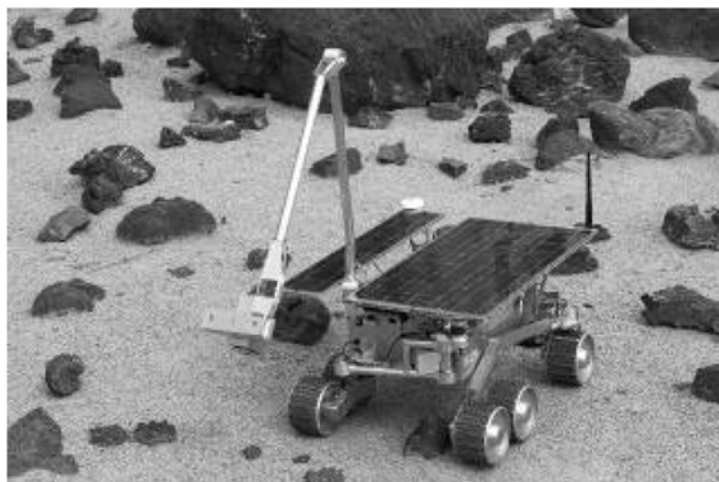


Figure 1.21- Le rover Rocky 7

1.5 Etat de l'art sur l'analyse de l'interaction homme-robot

Dans cette partie, nous présentons un état de l'art sur l'IHR suivi par la sélection de quelques thèmes et exemples de recherche dans ce domaine.

1.5.1 Quelques thèmes choisis dans le domaine de l'IHR

Il existe de nombreux thèmes de recherche qui entrent dans le domaine de recherche de l'IHR. Nous citerons quelques-uns [61]:

a) Le robot dans l'éducation, la thérapie, la réhabilitation et le soutien des personnes âgées

Aider les personnes âgées à vivre de façon autonome plus longtemps est certainement un objectif très intéressant. La robotique d'assistance est un domaine croissant d'applications pour les robots de service. Il implique une sécurité critique et des questions éthiques. Les robots qui prennent en charge les personnes âgées sont étudiés dans de nombreux pays qui sont confrontés au défi d'une société vieillissante. C'est le cas où les robots prennent un rôle d'aide pour des personnes vulnérables ou ayant des besoins spéciaux.

b) Interactions multimodales, l'expressivité et compétences de conversation

L'évolution et l'introduction dans les espaces généralement réservés aux humains des robots de service et des robots humanoïdes ouvrent de nouveaux champs de recherche. En effet, les chercheurs visent à concevoir des robots avec des caractéristiques et des qualités humaines pour pouvoir communiquer et interagir avec les humains. Les études concernent, notamment, la communication gestuelle, l'interaction kinesthésique ainsi que l'organisation des espaces sociaux qui sont importants pour l'interaction homme-homme et qui sont de plus en plus appliqués pour l'interaction homme-robot. Nous notons aussi le cas des expressions faciales qui sont destinés à exprimer des émotions en conjonction avec d'autres signaux verbaux et non verbaux. Un autre champ de recherche très important concerne la communication linguistique et le dialogue homme-robot pour fournir une interface naturelle (plus naturel qu'un clavier ou un bouton poussoir) pour les applications nécessitant une communication directe entre les humains et les robots. L'objectif est de pouvoir faire interagir et faire coexister ensemble des humains avec des robots qui disposeront de moyens de communication multimodaux et capable de présenter des caractéristiques pouvant les identifier avec leurs personnalités comme les humains.

c) L'apprentissage social et l'acquisition de compétences par l'enseignement et imitation

Ce thème comprend les recherches sur le développement des robots qui peuvent s'adapter à des environnements et exigences dynamiques. Pour cela, ils doivent disposer d'un niveau de compétence et de connaissances qui soient améliorables par des techniques d'apprentissage par

démonstration ou par programmation. Ces robots pourraient également transférer les compétences nouvellement acquises à d'autres robots et qui pourraient donner lieu à une (encore futuriste) **culture des robots**.

d) Coopération et collaboration homme-robot

Dans ce domaine, la recherche se concentre sur des applications où les robots et les humains ne vivent pas seulement côte à côte, mais ils ont aussi besoin de travailler la main dans la main. C'est le cas, par exemple, des applications spatiales où les astronautes humains et les robots ont besoin d'assembler en collaboration des pièces ou d'effectuer des tâches. Cette coopération est fortement influencée par les caractéristiques d'une collaboration au sein d'une équipe humaine, mais qui fait face aussi à des défis particuliers. Notamment parce que les humains utilisent des modèles mentaux avec prédiction sur le comportement futur de leur partenaire humain, ce qui est difficile dans des situations face à un robot où on ne sait pas quelles sont les compétences que le robot possède ou ne possède pas.

e) Identification des intentions et compréhension des activités humaines

La perception de l'environnement par les robots est encore un problème ouvert. Cela va de la perception des objets à saisir ou des sites utilisés pour la navigation à la reconnaissance de ce que les humains font dans le but de coordonner les interactions homme-robot. En interaction humain-humain, des indices souvent subtiles et inconsciemment produits et perçus sont utilisés pour réguler et améliorer l'interaction. Il n'est pas encore clair dans quelle mesure la reconnaissance de ces indices peut être réalisée avec des systèmes de perception du robot.

1.5.2 Exemples de l'état de l'art sur les études de l'IHR

Afin d'exemplifier quelques-uns des thèmes du domaine de l'analyse de l'interaction homme-robot, le paragraphe suivant présente quelques exemples de travaux. Nous attirons l'attention que ces travaux qui sont sélectionnés ne couvrent pas le domaine de la recherche dans l'HRI.

L'article [62] est un exemple d'analyse de l'interaction homme-robot de longue durée en IHR spécifique du domaine de la réhabilitation et du soutien des personnes âgées (figure 1.22). Il présente un cas d'étude où un groupe de personnes âgées interagit deux fois par semaine en un mois avec le robot « Paro » (figure 1.23). Cette interaction a été effectuée dans des conditions réelles dans une maison de soins infirmiers. Le but de l'étude est d'analyser le comportement des malades par rapport au robot Paro dans l'environnement naturel de l'hôpital. La question la plus importante dans ce problème c'est l'interprétation réciproque des intentions à partir des réactions et des comportements des malades et du robot.

Cette étude préliminaire montre que la signification et l'interprétation lors de l'interaction dépend des caractéristiques physiques et fonctionnelles non seulement du robot, mais aussi, et surtout, du contexte spécifique de l'interaction.



Figure 1.22-Paro, le robot phoque



Figure 1.23- Un groupe de personnes âgées interagissant avec le robot Paro

Une autre étude à long terme pour une durée de plus de deux mois avec le même robot (Paro) a été effectuée en [63]. Cet article présente les résultats de cette expérience de thérapie assistée dans une maison de soins. Deux robots de types « Paro » ont été utilisés pendant plus de 9 heures par jour pour interagir avec les résidents. Afin d'étudier les effets psychologiques et sociaux des robots, les activités des résidents dans les zones publiques ont été enregistrées par des caméras vidéo pendant la journée pour plus de 2 mois (figure 1.24). Les résultats montrent que Paro a encouragé les résidents à communiquer les uns avec les autres et apporté des améliorations psychologiques en eux.



Figure 1.24- Le système d'enregistrement vidéo

Ces deux travaux soulignent l'importance des études à long terme pour les robots d'assistance, et montrent également la nécessité des **techniques d'évaluation quantitative** ainsi que des **techniques d'évaluation qualitative** pour l'analyse de l'interaction homme-robot [61]. Ces exemples montrent qu'une profonde compréhension de la dynamique d'interaction homme-robot est encore un domaine de recherche jeune et pas suffisamment exploré.

La détection et la compréhension des activités humaines est un élément important dans de nombreuses études de l'IHR. L'article [64] présente le développement d'un système pour le dialogue multimodal qui utilise une variété de capteurs (vision, scanner laser, .. etc). Afin de faciliter la détermination et le suivi du sujet d'interaction, une nouvelle approche pour le suivi du sujet basé sur le dialogue multimodal est présentée pour fournir un robot mobile avec un niveau élevé de conscience. Le robot utilisé dans cette expérience est le robot BIRON « Bielefeld Robot Companion» (Figure 1.25), il est capable de réaliser le suivi dynamique du sujet d'interaction « DynamicTopic Tracking (DTT) » dans une conversation parlé naturelle Homme-robot.



Figure 1.25- Le robot BIRON (Bielefeld Robot Companion)

Ce scénario d'interaction homme-robot est décomposé de séquences de moyenne taille pour 29 sujets d'interaction qui décrivent une chambre pour le robot BIRON. L'espace de travail est une salle

d'environnement domestique qui contient plusieurs domaines thématiques. Une empreinte d'une interaction entre un humain et le robot BIRON est présentée par la figure 1.26.

Les chercheurs dans ce travail ont noté que les performances du système de suivi de sujet d'interaction peuvent être améliorées en tenant compte d'autres informations contextuelles et caractéristiques supplémentaires provenant de différents modalités (**nombre de modes disponibles**).



Figure 1.26- Interaction homme-robot dans l'environnement BITT

Nous avons remarqué que, en raison de la nature très innovante des études de l'IHR, les chercheurs ne peuvent pas concevoir leurs expériences en accord sur des méthodologies bien établies. Il n'existe pas jusqu'à présent, une formulation générale de méthodes et de méthodologies pour la conception de l'IHR. Il est difficile de reproduire des séries d'expériences compte tenu de la variabilité du comportement humain et du contexte. La recherche se poursuit sur l'application ou non, des méthodes statistiques quantitatives nécessitant des expériences à grande échelle, ou à poursuivre une approche qualitative des études à petite échelle concentrées sur des études de un cas précis.

Afin de faire progresser le domaine de l'analyse de l'interaction homme-robot, il est important de «faire la paix» entre les partisans de l'une ou l'autre méthode ou approche [61]. Cependant, il est important de comprendre que la nature de l'interaction homme-robot est liée mais différente de l'interaction homme-ordinateur ou humain-humain. Par conséquent, au lieu de simplement suivre les connaissances des sciences sociales, l'adaptation et le développement des nouvelles méthodes sont demandés. Il est important d'être précis sur les approches méthodologiques utilisées dans les études de l'IHR, mais en même temps il faut être conscient qu'il n'y a pas de solution générale à ce jour pour l'IHR.

Nous avons remarqué aussi dans la synthèse de l'état de l'art que sous la pression de diverses applications, les recherches avancées dans les systèmes d'interaction homme-machine s'intéressent à une nouvelle génération de systèmes qui peuvent être nommés "les systèmes de contrôle intégrés et

supervisés ou SCIS". ISCS sont des systèmes complexes, HMIS synergique. Ils tentent d'intégrer des techniques multimédias, des outils de simulation et techniques modernes de programmation de haut niveau. Ils offrent divers modes de contrôle telles que la télémanipulation, la vision, le geste, la parole, etc. Ces systèmes sont destinés à effectuer des tâches et des missions dans divers environnements complexes. Actuellement, les ISCS sont présents dans plusieurs domaines industriels importants tels que les centrales nucléaires, les missions spatiales, des opérations militaires, l'industrie aéronautique, marine, télé-chirurgie, etc. Une première évaluation de l'état de l'art et les tendances dans ce domaine peut être déduite par l'étude de certains ISCS qui ont été décrits dans la littérature, par exemple [65-69].

1.6 Etat de l'art sur les robots à câbles

Bien que les études sur les robots parallèles soient récentes, différentes applications ont été envisagées, notamment en aviation (simulateurs d'avions), dans l'interface haptique, pour le soulèvement de charges, ainsi que pour la réhabilitation des handicapés. Afin d'illustrer l'état de l'art, nous présentons quelques exemples d'applications typiques.

L'application la plus connue est sans doute la Skycam, une caméra contrôlée par un mécanisme à câbles (Figure 1.27). Elle est utilisée dans les stades pour la télédiffusion des parties de football professionnel [70].

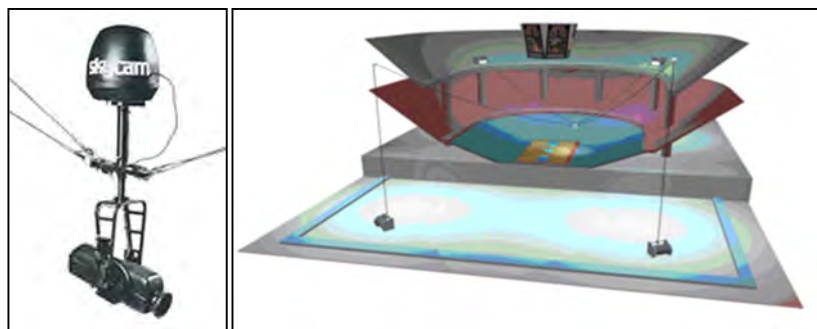


Figure 1.27- Caméra contrôlée par un mécanisme à câbles (Skycam).

Un autre champ d'intérêt dans les applications biomédicales est le suivi du mouvement. On citera l'exemple de CaTraSys (Cassino Tracking System) (Figure 1.28) qui a été utilisé pour l'identification des paramètres cinématiques et la mobilité de l'homme [71]. Le CaTraSys est un système de mesure qui a été conçu et construit au LARM (laboratoire de robotique et mécatronique) à Cassino (Italie). Il a été utilisé pour déterminer la position de l'extrémité des membres pendant les mouvements et en plus il peut mesurer les forces et les couples qui sont exercés par/sur le membre.



Figure 1.28- Représentation d'un capteur de force pour les robots à câbles (CaTraSys)

Le STRING-MAN est système à câbles pour la réadaptation de jambe [72] présenté par la(Figure 1.29). C'est un manipulateur à câbles pour soutenir le corps humain. Sa conception a été inspirée par le principe des cordes de marionnettes. La structure mécanique, de la figure 1.29, a une configuration avec dix câbles. Les câbles sont reliés au tronc et au bassin humain au moyen d'une interface utilisateur ; Le corps humain est l'organe terminal de robot. Il y a un tapis roulant qui permet le mouvement du patient.



Figure 1.29- Le system à câbles STRING-MAN

La figure 1.30 présente le prototype mécanique de STRING-MAN.



Figure 1.30- Le prototype mécanique de STRING-MAN

Une autre application est relative à une interface haptique réalisée sur la base d'un robot à huit câbles à l'université de l'Ohio [73]. Le but est de créer un dispositif d'entrée/sortie qui fournit des forces et des moments en six degrés de liberté (6ddl), qui agissent sur l'opérateur humain en réalité virtuelle dans les applications distantes. La structure est représentée dans la figure 1.31.



Figure 1.31- Interface haptique de 6 degrés de liberté

Une autre application intéressante a trait au robot parallèle à câbles IPAnema [74] (figure 1.32). Il est contrôlé par un système industriel en temps réel NC-contrôleur. La possibilité de le programmer en G-code (DIN 66025) assure un fonctionnement facile. En option, on dispose de l'intégration d'un contrôleur logique programmable (PLC) qui permet une intégration facile dans un environnement de production existant.

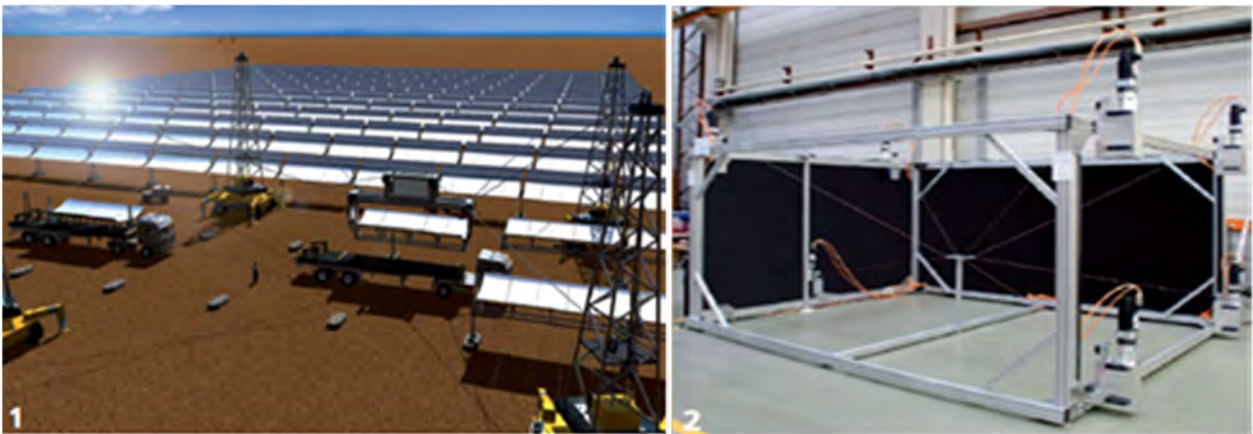


Figure 1.32- Robot parallèle à câbles utilisé dans les centrales solaires

Nerebot (Robot de neuro réhabilitation) et un robot basé sur les câbles. Il a été développé par le laboratoire de robotique du département de l'innovation mécanique et management à l'université de (PADUA) en Italie (figure 1.33). Il dispose de trois degrés de liberté et a été conçu pour la réhabilitation post-choc [75].

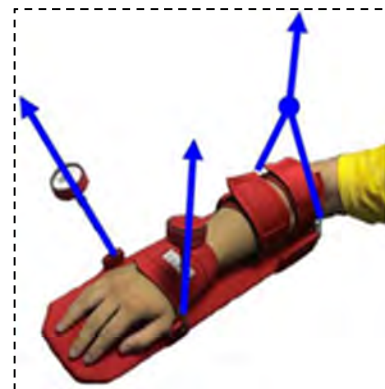


Figure 1.33- NeReBot est un robot à câble pour la réadaptation de membre supérieure avec 3 ddl

En particulier MariBot est un perfectionnement de NeReBot (neuro- robot de réadaptation) (figure 1.34.a), qui est un dispositif à base de câbles destinée à la réadaptation de membre supérieur. La figure 1.34.b représente le robot a câbles constitué de trois fils en nylon qui contrôlent le mouvement du bras du patient [75].



a- Une vue du système complet



b- Un arrangement de la plate-forme pour les bras supérieurs.

Figure 1.34- Le robot MariBot

1.7 Conclusion

Nous avons effectué une analyse extensive sur l'état de l'art concernant les techniques de contrôle haut niveau et l'IHR. Nous avons présenté quelques systèmes et thèmes d'interaction homme-robot typiques. Nous avons remarqué que cette interaction est située au cœur de la technologie moderne et future, et que l'interaction multimodale d'inspiration humaine est la réponse appropriée pour la conduite des systèmes complexes du futur. Nous avons mis en évidence un certain nombre de

questions concernant la conception des interfaces multimodales qui essentiellement conduisent à réduire leur complexité.

Cette analyse de l'état de l'art dans l'interaction des systèmes multimodaux nous a permis de conclure qu'il existe un problème de méthodologie pour la conception et le développement de ces systèmes. Nous avons remarqué aussi que les chercheurs dans ce domaine ne disposent pas de méthodologies définitivement bien établies et généralisables en dehors de leur contexte particulier. Nous avons également remarqué que la plupart des systèmes IHR son monomodes ou bi-modes et qu'ils restent à un niveau d'étude expérimental. D'autre part, nous avons remarqué que pour estimer les performances des systèmes supervisés multimodaux, il y'a lieu de tenir compte des critères quantitatifs mais également des critères qualitatifs qui dépendent du facteur humain. Pour cette raison, il est nécessaire de proposer une approche qui puisse combiner les deux types de critères.

Chapitre 2

Les Prototypes Expérimentaux Réalisés

Sommaire

2.1	Introduction	41
2.2	Le robot parallèle à câbles	41
2.2.1	Le robot parallèle à 3 câbles	42
2.2.1.1	Modélisation	42
2.2.1.1.1	Présentation de la structure du robot.....	42
2.2.1.2	Simulation du control du RPCs.....	48
2.2.1.2.1	La commande PD du robot.....	49
2.2.1.3	Développement de la plateforme d'étude.....	51
2.2.1.3.1	Description du fonctionnement	52
2.2.2	Le robot parallèle à 4 câbles	62
2.2.2.1	Modélisation	62
2.2.2.1.1	Modélisation Géométrique	62
2.2.2.1.2	Modélisation Cinématique.....	62
2.2.2.1.3	L'analyse statique des forces.....	63
2.2.2.1.4	Modélisation Dynamique	64
2.2.2.2	Simulation du control du Robot à 4 câbles	65
2.2.2.3	Réalisation expérimentale du robot parallèle à 4 câbles	68
2.2.2.3.1	La structure mécanique du robot.....	68
2.2.2.3.2	La partie électronique	70
2.3	Le robot mobile avec bras manipulateur	74
2.3.1	La réalisation mécanique.....	74
2.3.1.1	Le bras manipulateur	75
2.3.1.2	La base mobile	76
2.3.2	Modélisation géométrique du bras manipulateur	78
2.3.2.1	Modèle géométrique inverse.....	78
2.4	Conclusion	80

2.1 Introduction

Compte tenu des applications envisagées qui consistent à concevoir et à réaliser une interface multimodale adaptable à différents systèmes robotiques ainsi qu'à leurs combinaisons; nous présenterons dans ce chapitre un certain nombre de systèmes robotiques que nous avons conçu et réalisé pour tester et analyser les capacités de notre Interface multimodale. Plus précisément, nous allons présenter la conception, la simulation, la réalisation, et le fonctionnement des robots suivants: un robot à 3 câbles; un robot à 4 câbles et un robot mobile portant un bras manipulateur.

Pour les robots à câbles, nous présenterons également l'implémentation des contrôleurs bas-niveau de type proportionnelle-dérivée pour l'asservissement des moteurs électriques à courant continu qui actionnent les câbles.

La conception et la réalisation des robots à câbles sont motivées par leur nouveauté dans le domaine de la robotique et par les caractéristiques spécifiques qu'ils offrent. Ils sont potentiellement intéressants pour le développement de nouvelles applications, notamment en réhabilitation.

Nous appliquerons le contrôle monomodal pour tester le bon fonctionnement de ces systèmes et les possibilités d'adaptabilité des modes de commande avant d'appliquer le contrôle multimodal qui sera le sujet du chapitre 5.

2.2 Les robots parallèles à câbles

Un robot parallèle à câbles "RPC" est un type particulier de robot utilisant des liaisons constituées de câbles au lieu de segments et vérins métalliques. Ces robots sont principalement constitués d'une base, d'un effecteur mobile (organe terminal), de câbles reliant en parallèle la plate-forme à la base et d'un ensemble de poulies motorisés. La Figure 2.1 montre un exemple de robot à quatre câbles développé dans notre laboratoire. Généralement, la base est fixe et chaque câble est attaché à une d'extrémité de la plate-forme. Sous l'effet des couples moteurs, le câble s'enroule ou se déroule autour de la poulie permettant de contrôler la position et l'orientation de la plate-forme mobile.

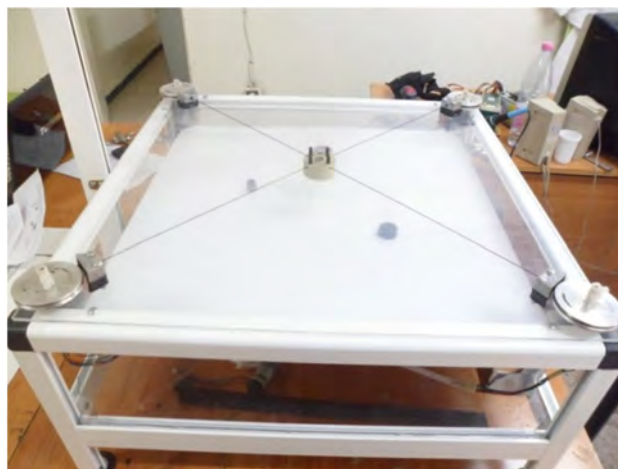


Figure 2.1- Exemple de robot parallèle à câbles

Les robots parallèles à câbles présentent plusieurs avantages [76] :

- Un grand espace de travail en comparaison avec les manipulateurs parallèles classiques
- Les robots à câbles ont une structure légère qui présente de bonnes propriétés dynamiques
- Un rapport charge utile/masse élevé et un prix de construction réduit
- Une bonne portabilité et une construction économique
- Les robots parallèles à câbles, ont une structure facile à monter, à démonter, à stocker, à transporter et donc facile à utiliser

2.2.1 Le robot parallèle à 3 câbles

Nous présentons dans cette section l'étude et la réalisation d'un robot parallèle à 3 câbles.

2.2.1.1 Modélisation de robot à 3 câbles

2.2.1.1.1 Présentation de la structure du robot

Le robot étudié est constitué de trois câbles parallèles (figure 2.2). La structure mécanique de ce robot est constituée par:

- Une base (cadre) fixe de forme triangle.
- Une plateforme mobile, qui porte l'organe terminal.
- La base et la plateforme mobile sont liées par le biais de trois câbles.
- Chaque câble est lié à la base par un actionneur (moteurs).

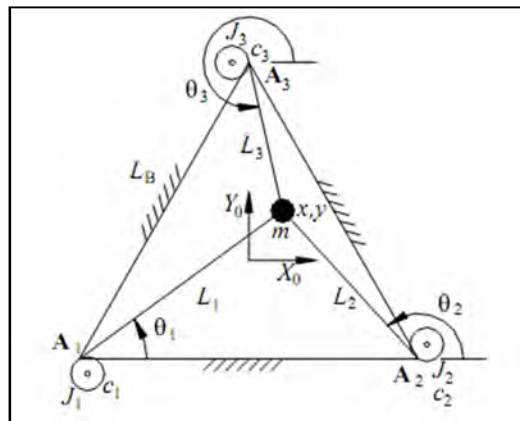


Figure 2.2- Exemple d'un robot parallèle à 3 câbles

Les éléments intervenants dans le schéma de la figure 2.2 sont :

r : Le rayon de la poulie de chaque moteur

m : La masse de l'organe terminal

C : Les coefficients d'amortissement visqueux de chaque arbre moteur

J : Les inerties du rotor et de la poulie de chaque moteur

$\tau(i)$: Les couples appliqués par les moteurs aux câbles

A_i : Les points de fixation des câbles aux moteurs

L_b : La longueur du côté du triangle limitant l'espace de travail.

Θ_i : L'angle de rotation des câbles par rapport à l'axe X

L_i : Les longueurs des câbles

2.2.1.1.2 Modélisation Géométrique

Dans cette section, nous allons établir le modèle géométrique direct et le modèle géométrique inverse pour le robot 3 câbles.

- **Modèle Géométrique Inverse (MGI)**

Ce modèle a pour but de déterminer les longueurs des câbles « L_i » et les angles « θ » entre l'axe X et les câbles en fonction de la position $\{X, Y\}$. Le modèle géométrique inverse peut s'exprimer par les relations suivantes [77] :

$$L_i = \sqrt{(x - A_{ix})^2 + (y - A_{iy})^2} ; i=1,2,3 \quad (2.1)$$

$$\theta_i = \arctang\left(\frac{y - A_{iy}}{x - A_{ix}}\right) ; i=1, 2, 3 \quad (2.2)$$

- **Modèle Géométrique Direct (MGD)**

Le MGD exprime la position de l'effecteur $M(x, y)$ en fonction des longueurs des câbles L_i . Pour les manipulateurs parallèles, le modèle géométrique directe est difficile à résoudre à cause de sa structure fermée (les expressions des angles θ_i sont mathématiquement liées avec les longueurs des câbles L_i). La relation entre la position $X=(x, y)$ et les coordonnées généralisées est non linéaire.

Ce problème peut être simplifié en déplaçant le repère $R(O,X,Y)$ en point A_1 ce qui nous donne des nouvelles coordonnées des points $A_1=(0,0)^T$ et $A_2=(L_b,0)^T$. Alors la solution du modèle géométrique direct est l'intersection de deux cercles, un de centre A_1 et de rayon L_1 , et l'autre de centre A_2 et de rayon L_2 . La position de l'end-effecteur est exprimée par les relations suivantes [77]:

$$x = \frac{L_b^2 + L_1^2 - L_2^2}{2L_b} \quad (2.3)$$

$$y = +\sqrt{L_1^2 - x^2} \quad (2.4)$$

2.2.1.1.3 Modélisation Cinématique

Dans cette partie, nous présentons la modélisation inverse et directe de la vitesse du robot parallèle plan à 3-câbles.

- **Modèle Cinématique Inverse**

Pour calculer le modèle cinématique inverse, nous considérons le i^{eme} vecteur suivant :

$$x = Aix + L_i \cos(\theta_i) \text{ Et } y = Aiy + L_i \sin(\theta_i) \quad (2.5)$$

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} Aix + L_i \cos(\theta_i) \\ Aiy + L_i \sin(\theta_i) \end{pmatrix} \quad (2.6)$$

L'inverse de la dérivé du vecteur de position par rapport au temps est donné par:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} \cos(\theta_i) & -L_i \sin(\theta_i) \\ \sin(\theta_i) & L_i \cos(\theta_i) \end{pmatrix} \begin{pmatrix} \dot{L}_i \\ \dot{\theta}_i \end{pmatrix} \text{ pour } i = 1,2,3 \quad (2.7)$$

A partir de cette expression, nous déduisons:

$$\begin{pmatrix} \dot{L}_i \\ \dot{\theta}_i \end{pmatrix} = \begin{pmatrix} \cos(\theta_i) & \sin(\theta_i) \\ -\sin(\theta_i) & \cos(\theta_i) \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} \text{ pour } i = 1,2,3 \quad (2.8)$$

Comme nous nous intéressons à la longueur des câbles en fonction de la position de l'effecteur, nous pouvons extraire la première ligne de l'équation (2.8) pour obtenir le modèle cinématique inverse du robot.

$$\begin{pmatrix} \dot{L}_1 \\ \dot{L}_2 \\ \dot{L}_3 \end{pmatrix} = \begin{pmatrix} \cos(\theta_1) & \sin(\theta_1) \\ \cos(\theta_2) & \sin(\theta_2) \\ \cos(\theta_3) & \sin(\theta_3) \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} \quad (2.9)$$

- **Modèle Cinématique Direct**

Pour obtenir le modèle cinématique direct, on doit inverser l'équation (2.9) qui nous donne

$$\dot{X} = M^{-1} \dot{L} \quad (2.10)$$

Donc la solution exige le calcul de la matrice jacobéenne inverse. En raison de redondance d'actionnement, M n'est pas carré mais de dimension (nx2) ce qui crée un problème de calcul de M^{-1} . Alors pour résoudre ce problème, nous avons utilisé le pseudo inverse de Moore-Penrose :

$$\dot{X} = M^+ \dot{L} \text{ où } M^+ = (M^T M)^{-1} M^T \quad (2.11)$$

Où :

\dot{L} : Le vecteur de vitesse des trois câbles.

M^+ : Le pseudo inverse de Moore-Penrose.

$\dot{X} = (\dot{x}, \dot{y})^T$: Le vecteur de vitesse de l'organe terminal.

2.2.1.1.4 L'analyse statique des forces

A l'équilibre statique, la somme des forces extérieures exercées sur l'effecteur par les câbles doit être égale à la force résultante F_R externe exercée sur l'environnement. La figure 2.3 montre la statique du corps libre pour 3 câbles [77].

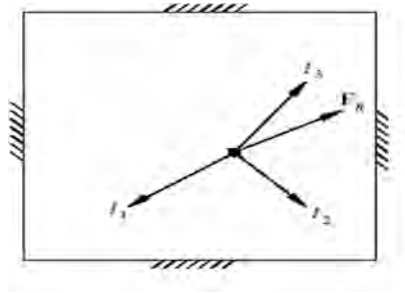


Figure 2.3- Forces statiques pour le robot 3-câbles

L'équation suivante exprime cette relation :

$$\sum_{i=1}^3 t_i = -\sum_{i=1}^3 t_i \vec{L}_i = F_R \quad (2.12)$$

$$\text{Où :} \quad L_i = (\cos(\theta_i) \sin(\theta_i))^T$$

Dans ce cas, la gravité est ignorée par ce qu'elle est supposée perpendiculaire au plan du RPC.

L'équation (2.12) peut être écrite sous la forme :

$$ST = F_R \quad (2.13)$$

Où $S = (-\vec{L}_1, -\vec{L}_2, -\vec{L}_3)^T$ est une matrice de dimension (2×3) , est la transposée de la matrice jacobéenne. Cette matrice dépend des caractéristiques physiques du manipulateur et $T = (t_1, t_2, t_3)^T$ est le vecteur des tensions des câbles.

$$\begin{pmatrix} -\cos\theta_1 & -\cos\theta_2 & -\cos\theta_3 \\ -\sin\theta_1 & -\sin\theta_2 & -\sin\theta_3 \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix} = \begin{pmatrix} f_x \\ f_y \end{pmatrix} \quad (2.14)$$

Cette équation est sous contrainte, ce qui signifie qu'il y a une infinité de solutions du vecteur des tensions des câbles T pour exercer la force F_R . Pour inverser cette équation (exprimer les tensions des câbles T en fonction de F_R), nous utilisons la notion de solution particulière et homogène.

$$T = S^+ F_R + (I_3 - S^+ S)z \quad (2.15)$$

Où I_3 est la matrice d'identité de dimension (3×3) , z est un vecteur arbitraire de dimension (3) , et S^+ est le pseudo inverse de S calculé par la méthode de Moore-Penrose de dimension (3×2) .

Le premier terme est la solution particulière, et le deuxième terme est la solution homogène. Une expression équivalente à l'équation (2.15) est :

$$T = \begin{pmatrix} t_{p1} \\ t_{p2} \\ t_{p3} \end{pmatrix} + \alpha \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} \quad (2.16)$$

La solution particulière est le premier terme de l'équation (2.15) et la solution homogène est exprimée comme le noyau du vecteur N multiplié par un scalaire arbitraire α . Afin de déterminer si un point donné se trouve dans l'espace de travail statique pour un simple RPC il faut assurer que les tensions t_i ($i = 1,2,3$) sont positives pour toutes les forces exercées. Alors il est nécessaire et suffisant que toutes les composantes du vecteur de noyau N ont le même signe ($\forall n_i, n_i > 0$, ou, $n_i < 0$) $i = 1,2,3$. Si l'une de ces deux conditions n'est pas respectée, il existe un scalaire α peut être introduit pour que toutes les tensions deviennent positives. Mais si $\exists n_i = 0$ alors le point donné n'appartient pas à l'espace de travail. Pour calculer les composants de vecteur N nous avons : $n_i = (-1)^{i+1} |S_i|$ où $|S_i|$ est le facteur déterminant de sous matrice de S, ce qui donne :

$$N = \begin{pmatrix} \sin(\theta_3 - \theta_2) \\ \sin(\theta_1 - \theta_3) \\ \sin(\theta_2 - \theta_1) \end{pmatrix} \quad (2.17)$$

Les intervalles admissibles des angles des câbles sont [77] :

$$0 \leq \theta_1 \leq 60^\circ, 120^\circ \leq \theta_2 \leq 180^\circ, \text{ et } 240^\circ \leq \theta_3 \leq 300^\circ$$

Et les valeurs possibles de trois composants de N sont :

$$60^\circ \leq \theta_3 - \theta_2 \leq 180^\circ, -300^\circ \leq \theta_2 - \theta_1 \leq -180^\circ, \text{ et } 60^\circ \leq \theta_2 - \theta_1 \leq 180^\circ$$

Pour assurer les tensions optimales il faut que :

$$t_{opt} = \begin{pmatrix} t_{p1} \\ t_{p2} \\ t_{p3} \end{pmatrix} + \alpha_i \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} \geq t_{min} \quad (2.18)$$

$$\alpha_i = (t_{min} - t_{pi})/n_i, \quad \alpha_i > 0 \Rightarrow n_i > 0 \quad (2.19)$$

$$t_{min} = \text{Max} \left\{ J \left(\frac{d}{dt} \left(\frac{\partial \beta}{\partial \dot{X}} \right) \dot{X} + \frac{\partial \beta}{\partial X} \ddot{X} \right) + C \frac{\partial \beta}{\partial X} \dot{X}, 0 \right\} \quad (2.20)$$

Cette équation doit être vérifiée pour que toutes les tensions soient positives.

2.2.1.1.5 Modélisation Dynamique

• Modèle Dynamique de l'effecteur final

Le modèle dynamique de l'effecteur final est exprimé par la relation suivante [78] :

$$m\ddot{X} = F_R \quad (2.21)$$

Où la matrice cartésienne de masse $m = \begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix}$, $\ddot{X} = (\ddot{x}, \ddot{y})^T$ le vecteur d'accélération de l'organe terminal et $F_R = (F_{Rx}, F_{Ry})^T$ la force résultante de toutes les tensions des câbles appliquées sur l'organe terminal.

- **Le modèle dynamique des actionneurs**

Le comportement dynamique de moteur est exprimé par l'équation :

$$J\ddot{\beta} + C\dot{\beta} = \tau - rT \quad (2.22)$$

Avec :

$$J = \begin{bmatrix} J_1 & 0 & 0 \\ 0 & J_2 & 0 \\ 0 & 0 & J_3 \end{bmatrix}, \text{ et } C = \begin{bmatrix} C_1 & 0 & 0 \\ 0 & C_2 & 0 \\ 0 & 0 & C_3 \end{bmatrix}$$

Sont des matrices diagonales qui représentent les inerties et les coefficients d'amortissement visqueux de chaque moteur. Nous considérons que tous les rayons de la poulie sont identiques ($r_i = r$; $i=1, 2, 3$), $\tau = (\tau_1, \tau_2, \tau_3)^T$ est le vecteur des couples appliqués par les moteurs, $T = (t_1, t_2, t_3)^T$ est le vecteur des tensions des câbles, $\beta = (\beta_1, \beta_2, \beta_3)^T$ est le vecteur des angles de rotation des poulies (figure 2.4).

Donc, nous pouvons écrire que :

$$T = \frac{1}{r}(\tau - J\ddot{\beta} - C\dot{\beta}) \quad (2.23)$$

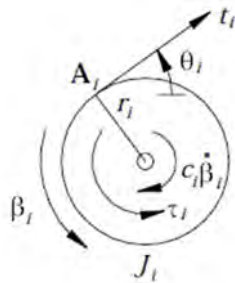


Figure 2.4- Structure de poulie

- **Modèle dynamique du système**

Le modèle dynamique global du système est obtenu par combinaison entre les modèles dynamiques de l'organe terminal et des moteurs. On considère que les angles des poulies sont nuls quand la position de l'organe terminal est au centre de polygone $X = (0,0)^T$. Par rapport à cette position les relations entre les angles de rotation des poulies β_i et les variations des longueurs des câbles ΔL_i sont :

$$\beta_i r = -\Delta L_i \quad \text{et} \quad \Delta L_i = L_i - L_{i0} \quad (2.24)$$

$$\text{Où} \quad L_i = \sqrt{(x - Aix)^2 + (y - Aiy)^2} ; i = 1, 2, 3 \quad (2.25)$$

Les longueurs initiales des câbles sont : $L_{i0} = \sqrt{(Aix)^2 + (Aiy)^2}$

Donc :

$$\beta = \begin{pmatrix} \beta_1(X) \\ \beta_2(X) \\ \beta_3(X) \end{pmatrix} = \frac{1}{r} \begin{pmatrix} L_{10} - L_1 \\ L_{20} - L_2 \\ L_{30} - L_3 \end{pmatrix} \quad (2.26)$$

En dérivant successivement (2.26) par rapport au temps, on obtient :

$$\ddot{\beta} = \frac{1}{r} \left[\begin{pmatrix} \dot{\theta}_1 \sin(\theta_1) & -\dot{\theta}_1 \cos(\theta_1) \\ \dot{\theta}_2 \sin(\theta_2) & -\dot{\theta}_2 \cos(\theta_2) \\ \dot{\theta}_3 \sin(\theta_3) & -\dot{\theta}_3 \cos(\theta_3) \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} - \begin{pmatrix} \cos(\theta_1) & \sin(\theta_1) \\ \cos(\theta_2) & \sin(\theta_2) \\ \cos(\theta_3) & \sin(\theta_3) \end{pmatrix} \begin{pmatrix} \ddot{x} \\ \ddot{y} \end{pmatrix} \right] \quad (2.27)$$

Par substitution de l'équation (2.27) en (2.23) on obtient :

$$T = \frac{1}{r} \left(\tau - J \left(\frac{d}{dt} \left(\frac{\partial \beta}{\partial \dot{x}} \right) \dot{x} + \frac{\partial \beta}{\partial x} \ddot{x} \right) - C \frac{\partial \beta}{\partial x} \dot{x} \right) \quad (2.28)$$

Par combinaison de (2.27) et (2.13), l'ensemble des équations du modèle dynamique peut être exprimé dans une forme standard pour les systèmes robotiques [78] :

$$M(X)\ddot{X} + N(X, \dot{X}) = S(X)\tau \quad (2.29)$$

$$\text{Où :} \quad M = rm + S(X)J \frac{\partial \beta}{\partial \dot{x}} \quad \text{et} \quad N(X, \dot{X}) = S(X) \left(J \frac{d}{dt} \frac{\partial \beta}{\partial \dot{x}} + C \frac{\partial \beta}{\partial \dot{x}} \right) \dot{x}$$

$$M = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \quad \text{et} \quad N(X, \dot{X}) = \begin{pmatrix} N_1(X, \dot{X}) \\ N_2(X, \dot{X}) \end{pmatrix}$$

2.2.1.2 Simulation du contrôle du RPC

Dans cette section, nous présentons notre architecture de contrôle pour le robot à 3 câbles en nous basant sur l'équation dynamique du mouvement (2.29). Les entrées pour le système est le vecteur des tensions τ . Chaque composante de τ doit être positive ou égale à zéro. Dans le but de résoudre ce problème, nous commençons par établir la représentation d'état du système.

L'équation dynamique du système est :

$$\ddot{X} = -M^{-1}(X)N(X, \dot{X}) + M^{-1}(X)S(X)\tau \quad (2.30)$$

$$\text{Ou : } M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \quad N = \begin{bmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{bmatrix} \quad \tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix}$$

A partir de l'équation dynamique nous effectuons les changements suivants :

$$\begin{cases} x_1(t) = x(t) \\ x_2(t) = \dot{x}_1(t) \\ x_3(t) = y(t) \\ x_4(t) = \dot{x}_3(t) \end{cases} \Rightarrow \begin{cases} \dot{x}_1(t) = x_2(t) \\ M_{11}\dot{x}_2(t) + M_{12}\dot{x}_4(t) = u_1(t) - N_{11}x_2(t) - N_{12}x_4(t) \\ \dot{x}_3(t) = x_4(t) \\ M_{21}\dot{x}_2(t) + M_{22}\dot{x}_4(t) = u_2(t) - N_{21}x_2(t) - N_{22}x_4(t) \end{cases}$$

Nous obtenons la représentation d'état suivante :

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & M_{11} & 0 & M_{12} \\ 0 & 0 & 1 & 0 \\ 0 & M_{21} & 0 & M_{22} \end{bmatrix} \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -N_{11} & 0 & -N_{12} \\ 0 & 0 & 0 & 1 \\ 0 & -N_{21} & 0 & -N_{22} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ u_1(t) \\ 0 \\ u_2(t) \end{bmatrix}$$

que nous pouvons écrire sous la forme suivante:

$$\dot{X}_{2d}(t) = f(X, X_{2d}) + g(X, X_{2d})U(t) \quad (2.31)$$

Avec :

$\dot{X}_{2d}(t)$: Le vecteur d'état du système

$f(X, X_{2d}), g(X, X_{2d})$: Sont des fonctions non linéaires

$U(t)$: Le vecteur de commande

2.2.1.2.1 La commande Proportionnelle-Dérivée du robot

La commande proposée en boucle fermée pour le robot est la commande classique proportionnelle dérivée (PD). L'établissement de la loi de commande (U_x, U_y) selon x et selon y est :

$$\begin{cases} U_x = K_p e_x + K_d \dot{e}_x \\ U_y = K_d e_y + K_d \dot{e}_y \end{cases} \quad (2.32)$$

K_p Et K_d sont les coefficients de proportionnalité et de dérivation de contrôleur. (e_x, e_y) représentent les erreurs selon x et y , (\dot{e}_x, \dot{e}_y) représentent les vitesses des erreurs selon x et y . La figure 2.5 représente le schéma du système en boucle fermée.

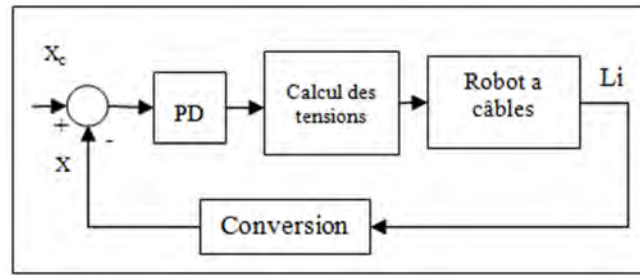


Figure 2.5- Le schéma de contrôle en boucle fermé

Nous avons simulé le contrôleur PD en tenant compte des contraintes à satisfaire sur les tensions des câbles, à savoir que les tensions doivent toujours être positives et comprises entre T_{min} et T_{max} . Les coefficients K_p et K_d sont estimés par essais et erreurs en simulation et valent :

$$K_p = 25000, \quad K_d = 100$$

Nous avons effectué la simulation de la commande point à point ainsi que celle de suivi de différentes trajectoires. Pour illustrer le rôle de notre contrôleur nous présenterons l'exemple du suivi d'une trajectoire circulaire par l'effecteur final (figure 2.6). Nous avons utilisé les paramètres suivants : $m = 0.01$ kg; J_i ($i = 1:3$) = 0.0008 kgm²; C_i ($i = 1:3$) = 0.01 Nms; $r_i = r = 0.013$ m; $L_b = 0.6$ m.

Les résultats de simulation obtenus confirment la possibilité d'utiliser effectivement le contrôleur PD pour commander notre robot à câbles.

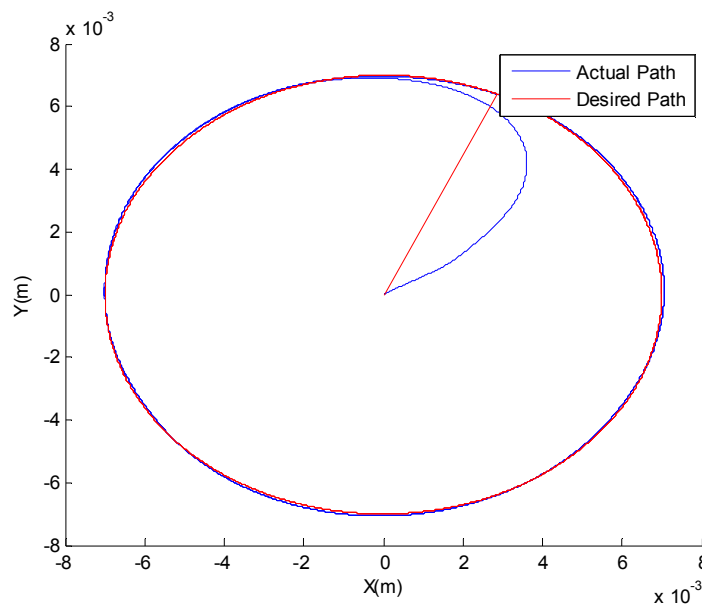


Figure 2.6- Suivi d'une trajectoire circulaire

Les figures 2.7 et 2.8 montrent l'évolution temporelle des longueurs des câbles et des tensions nécessaires pour assurer le suivi de la trajectoire circulaire.

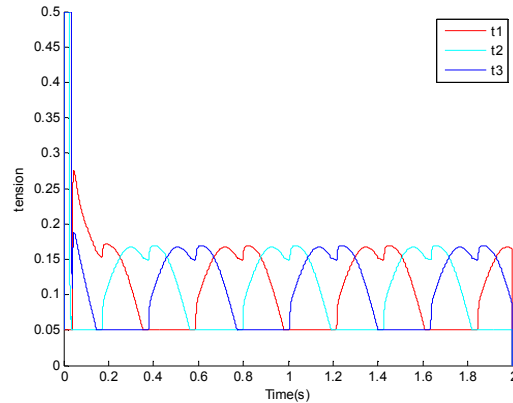
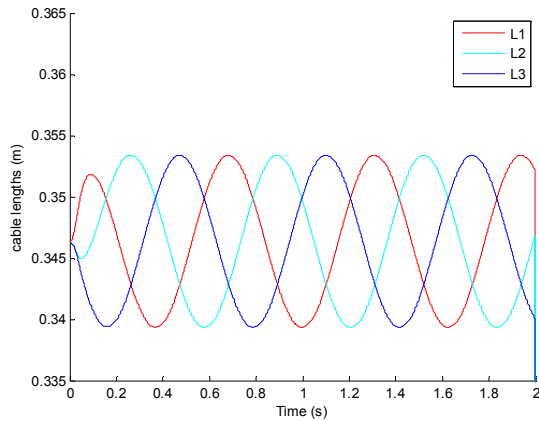


Figure 2.7- Les longueurs des câbles

Figure 2.8- Les tensions appliquées sur les câbles

2.2.1.3 Développement de la plateforme d'étude

Dans le but d'analyser les robots à câbles, une plateforme d'étude générique des robots à câbles a été conçue. Elle est constituée de deux parties (figure 2.9):

- La partie software comprend le programme principal constitué par les modèles géométrique, cinématique, et dynamique du robot étudié ainsi que les lois de commandes (PD, Mode Glissant, ...), avec un modèle virtuel 3D du robot en question.
- La deuxième partie comprend le robot expérimental reconfigurable selon le nombre de câbles envisagés.

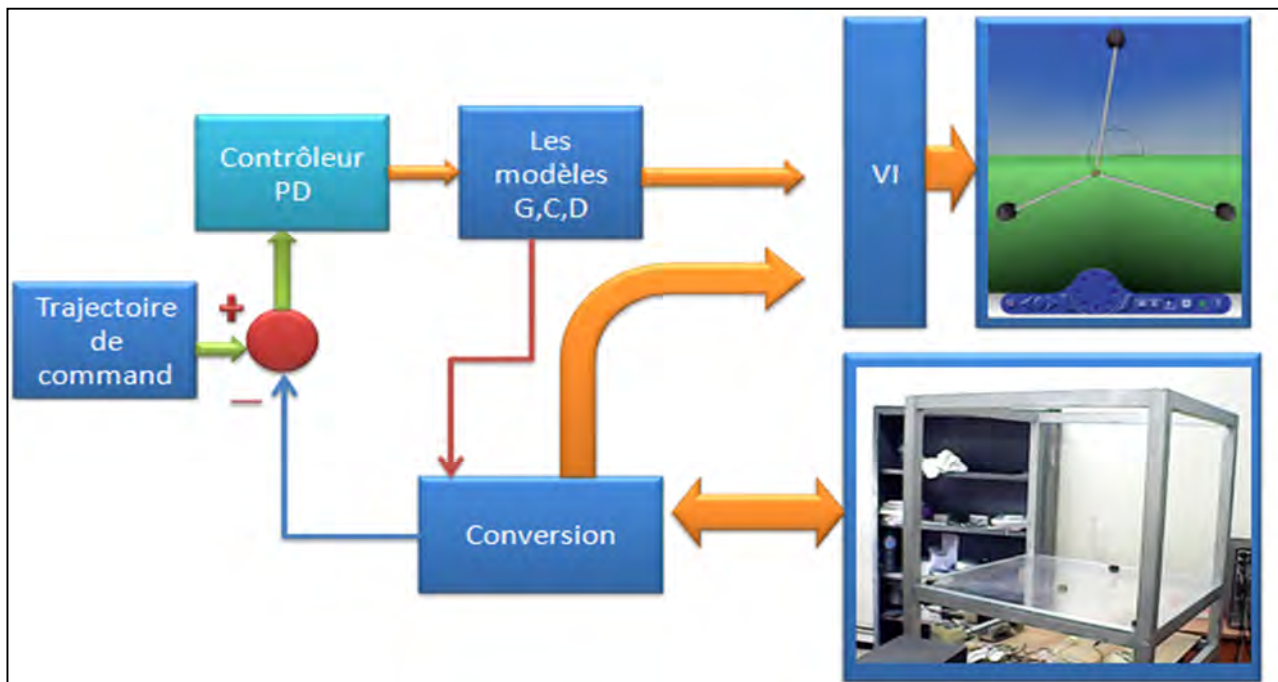


Figure 2.9- Schéma-block de la plateforme d'étude du robot à 3 câbles

Chaque partie est constituée de plusieurs blocks essentiels :

- Block de génération de trajectoires de référence : au niveau de ce block nous avons plusieurs sources de commande. Relativement au mode de commande utilisé nous pouvons choisir des trajectoires à partir d'un dictionnaire des trajectoires réalisés et sauvegardés pendant la phase d'acquisition (figure 2 .9).
- Block contrôleur : on peut sélectionner au niveau de ce block différent contrôleur (PD, mode glissant, ..).
- Block modèles Géométrique, Cinématique et Dynamique : nous avons implémenté au niveau de ce block les différentes modèles du système tel que le modèle géométrique, cinématique et dynamique, pour pouvoir contrôler le robot soit en mode réalité virtuelle ou augmenté.
- Block VI (Virtual Interface) : est utilisé pour interfacier l'environnement MATLAB/SIMULINK avec l'environnement VRML pour pouvoir commander le modèle virtuelle 3D. Nous avons développé ce simulateur de réalité virtuelle 3D du robot à 3 câbles au moyen du langage VRML (figure 2 .9). Il est interfacé avec Simulink-Matlab ce qui permet d'acquérir des signaux provenant du prototype expérimental par des capteurs associés à l'être humain afin de déterminer les mouvements des articulations. Ceci permettra d'effectuer l'apprentissage ou d'étudier l'évolution des capacités de la personne à réhabiliter.
- Block conversion : dans le but de comparer la trajectoire réalisée par le prototype expérimental avec la trajectoire de consigne, il faut que ces deux derniers soient de même types. Comme les trajectoires sauvegardées dans le dictionnaire de commande sont en fonction de la variable planaire $X_{\text{consigne}}(x_{\text{Consigne}}, y_{\text{Consigne}})$, il faut transformer les variables polaires (θ_i, L_i) qu'on reçoit à travers la carte d'acquisition en variables planaire $X(x, y)$.

2.2.1.3.1 Description du fonctionnement

- **La partie software**

Nous avons commencé par les signaux provenant de prototype expérimental, comme signal de feedback. Dans le but de contrôler le robot en boucle fermé, les moteurs utilisés au niveau des poulies sont équipés par des encodeurs en quadrature, pour assurer le calcul des longueurs des câbles dans les deux sens. Dans ce cas ; nous notons que les informations de position sont récupérées à travers des encodeurs en quadrature (figure 2.10) à effet hall (pas de rebonds) intégrés sur les moteurs de type EMG30. Ce type d'encodeurs donne la possibilité de déterminer la position et le sens de rotation des moteurs.

- Encodeur en quadrature de phase :

Les deux signaux de sortie d'un encodeur optique ont les formes suivantes :

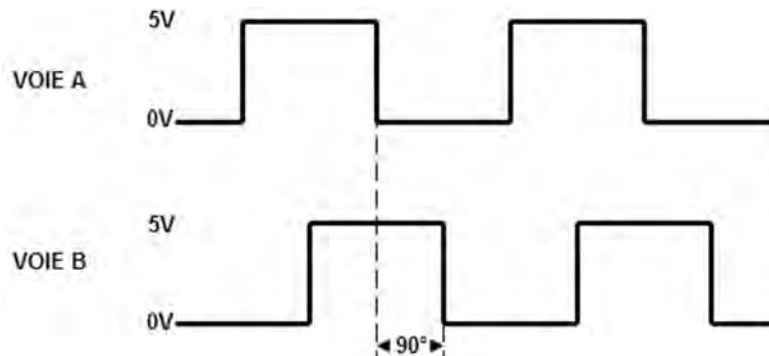


Figure 2.10- Les deux signaux de sortie d'un encodeur en quadrature de phase

Ce signal est intéressant car il est déphasé de 90° et permet de déterminer le sens de déplacement. En effet, si la voie A est en avance par rapport à la voie B, on se déplace dans un sens, dans le cas contraire on se déplace dans l'autre sens [79].

Pour l'implémentation du compteur, nous avons deux possibilités, une implémentation software ou hardware.

- Implémentation software du compteur

Pour notre application, nous avons implémenté le compteur en software. Le schéma suivant présente les états possibles en fonction du niveau logique issu du capteur. Chaque cycle peut compter jusqu'à 4 incréments ou décréments. Il est possible de modifier le programme de façon à prendre en compte que 1 ou 2 incréments par cycle en fonction de la précision que l'on désire (figure 2.11) [79].

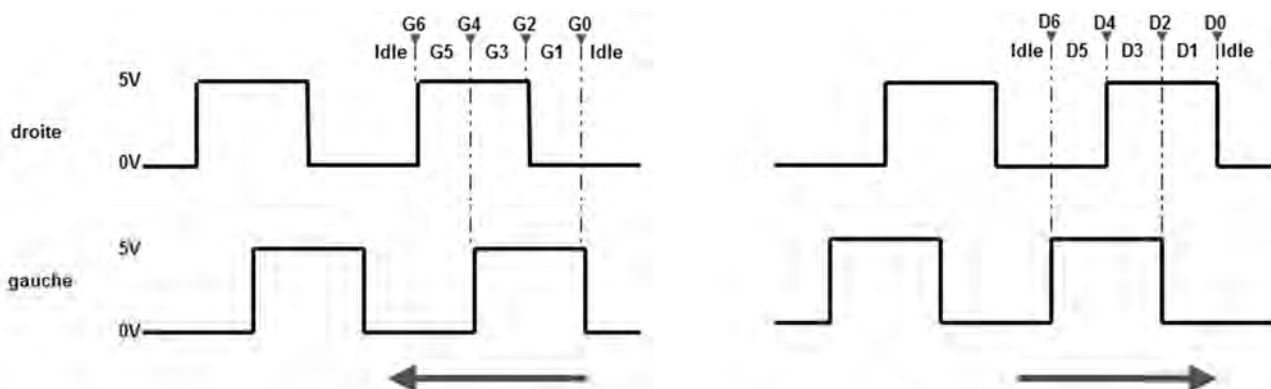


Figure 2.11- Diagramme d'état des deux encodeurs pour les deux sens

Les signaux des encodeurs sont récupérés à travers le block « Digital Input » de la librairie « Real Time Workshop » de l'environnement Simulink sous Matlab (figure 2.12), ce block est configuré pour les ports parallèles installés sur les slots PCI de l'ordinateur pour les différentes adresses suivantes : 378h pour le moteur 1, F600h pour le moteur 2 et F800h pour le moteur 3.

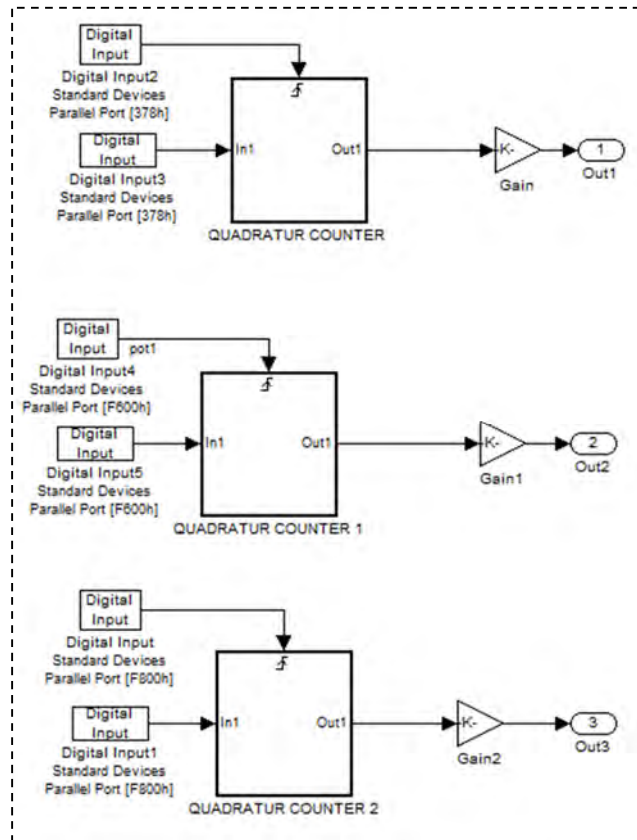


Figure 2.12- L'implémentation des encodeurs en quadrature

Le block « QUADRATUR COUNTER », est un sous-système qui implémente l'encodeur en quadrature présenté dans la figure 2.13. Le calcul des impulsions est déclenché par un événement trigger pour augmenter la précision et assuré en temps réel.

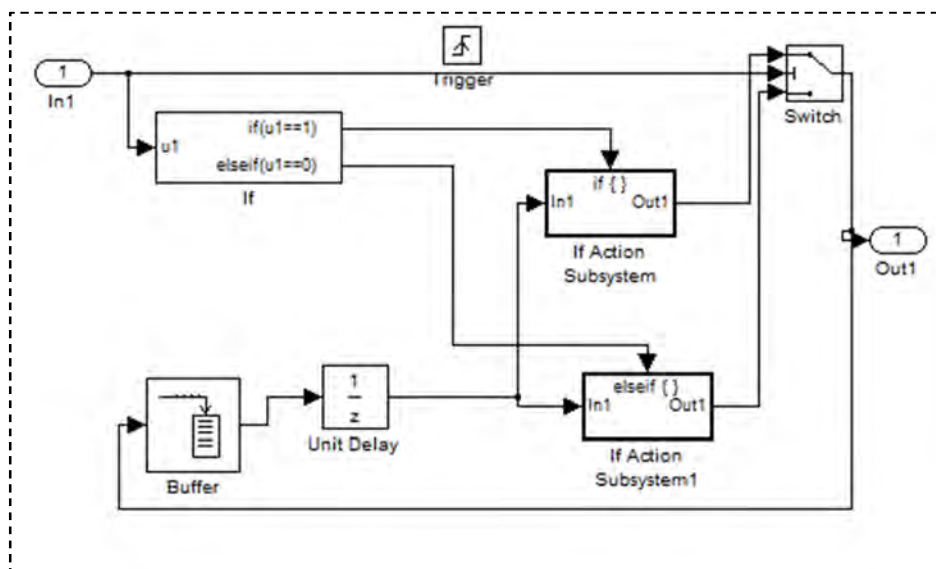


Figure 2.13-Le block « QUADRATUR COUNTER »

Dans le but de comparer la trajectoire réalisée par le prototype expérimental avec la trajectoire de consigne, il faut que ces deux derniers soient de même types. Comme les trajectoires sauvegardées

dans le dictionnaire de commande sont en fonction de la variable planaire $X_{\text{consigne}}(x_{\text{Consigne}}, y_{\text{Consigne}})$, il faut transformer les variables polaires (θ_i, L_i) reçus à travers la carte d'acquisition en variables planaires $X(x, y)$ en utilisant le modèle géométrique direct, qui est implémenté par le block « Embedded Matlab Function » (figure 2.14).

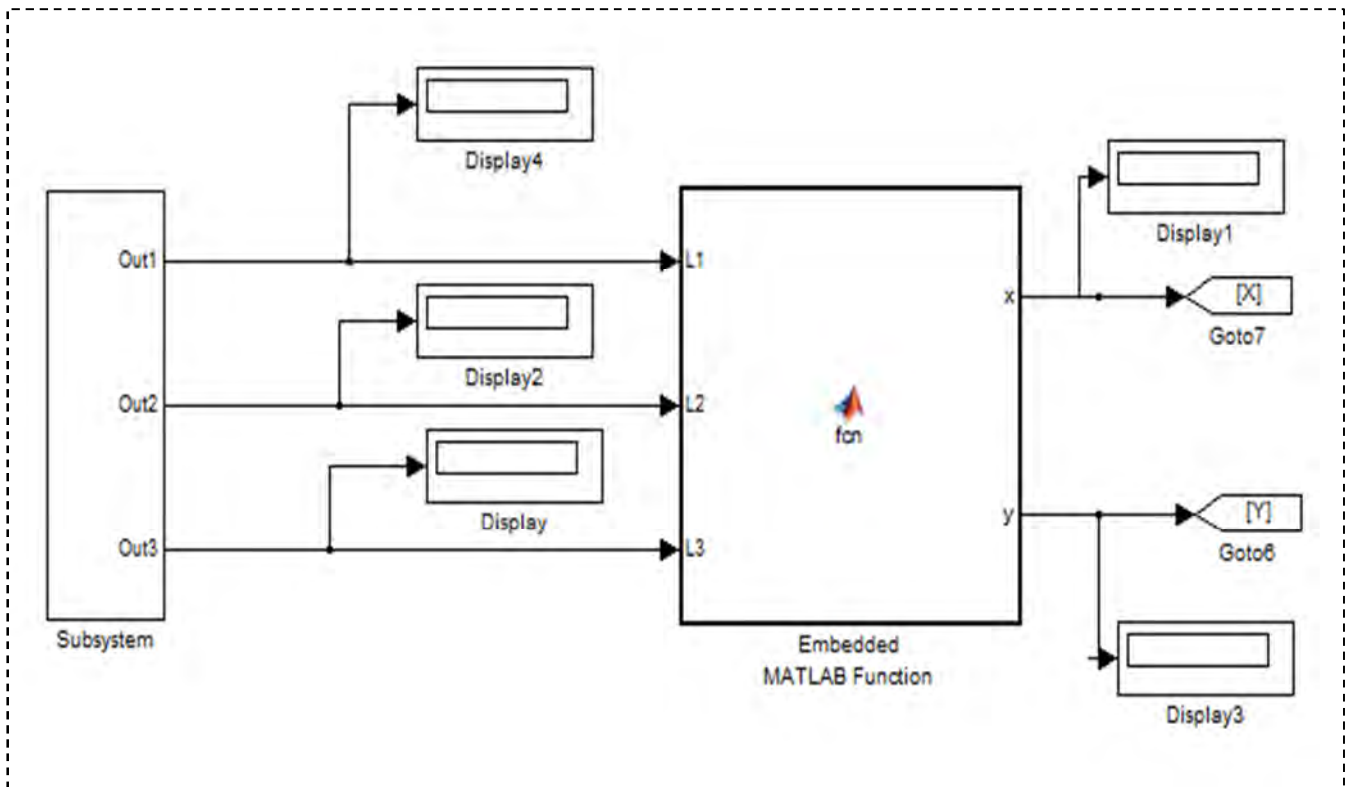


Figure 2.14- L'implémentation du modèle géométrique direct

Après l'étape de conversion, le contrôle de position est effectué par le biais d'un contrôleur PD implémenté en utilisant le block « Embedded Matlab Function » (figure 2.15). Le contrôleur PD calcule selon les deux lois de commande (PD_x sur X et PD_y sur y) la force résultante qu'il faut appliquer sur l'effecteur final. Le block « calcul des tensions » détermine selon le modèle cinématique, les différentes tensions qu'il faut appliquer sur les câbles. En utilisant les expressions mathématiques du modèle géométrique, on calcule les longueurs des câbles ainsi que les angles pour configurer le simulateur virtuel 3D.

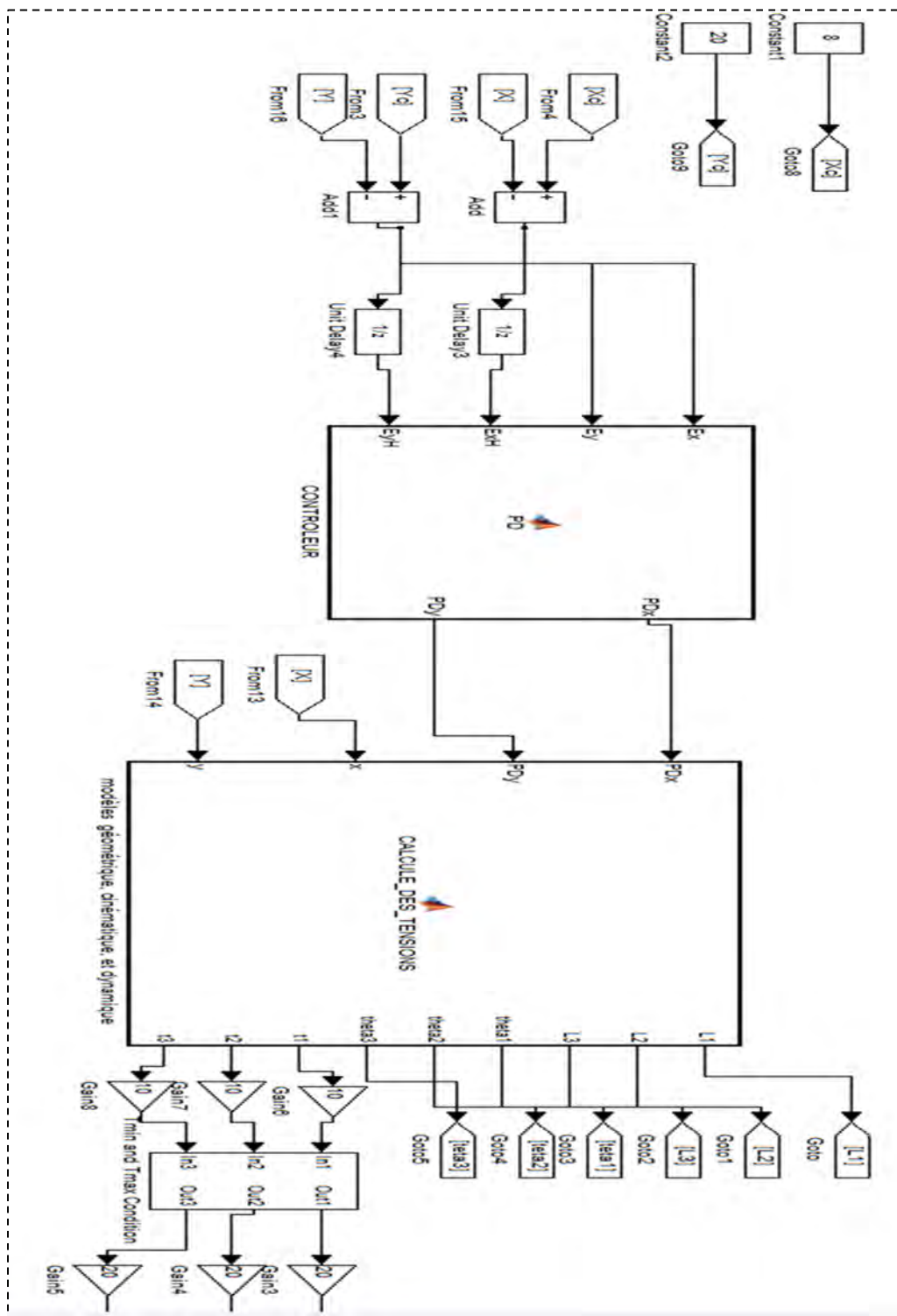


Figure 2.15- L'implémentation du contrôleur PD

Nous avons ajouté des interrupteurs pour pouvoir utiliser le système soit en mode contrôle ou en acquisition (figure 2.16). Pour le mode contrôle, les tensions envoyées au port parallèle sont celles calculées par le contrôleur PD. Dans le cas du mode acquisition, nous envoyons une tension constante égale à la tension minimale afin que nous puissions déplacer l'effecteur final, tout en assurant que tous les câbles soient tendus.

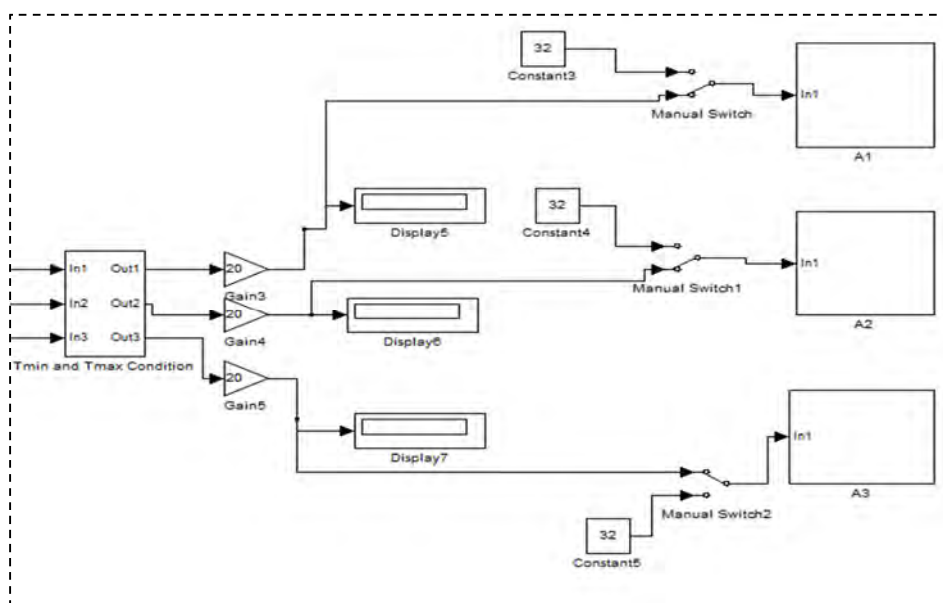


Figure 2.16- Les interrupteurs de sélection de mode

Pour envoyer les trois signaux de commande sur la carte d'interfaçage ; nous avons utilisé les blocks A1, A2 et A3 qui sont des sous-systèmes (figure 2.17). Dans chaque sous-système nous avons implémenté la conversion binaire en utilisant le block « Integer to Bit Converter » ainsi que l'écriture sur les pins du port parallèle en utilisant le block « Digital Output » de la librairie « Real Time Workshop ».

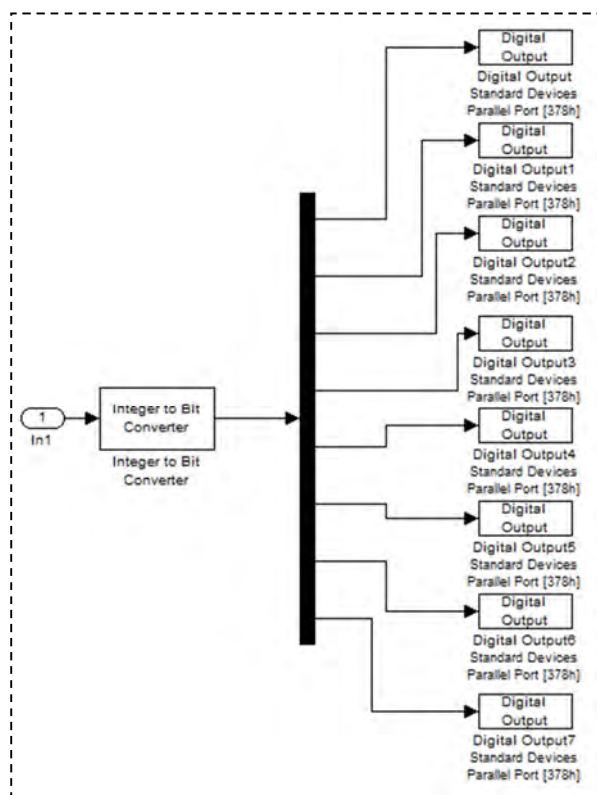


Figure 2.17- Le block « Digital Output » de la librairie « Real Time Workshop ».

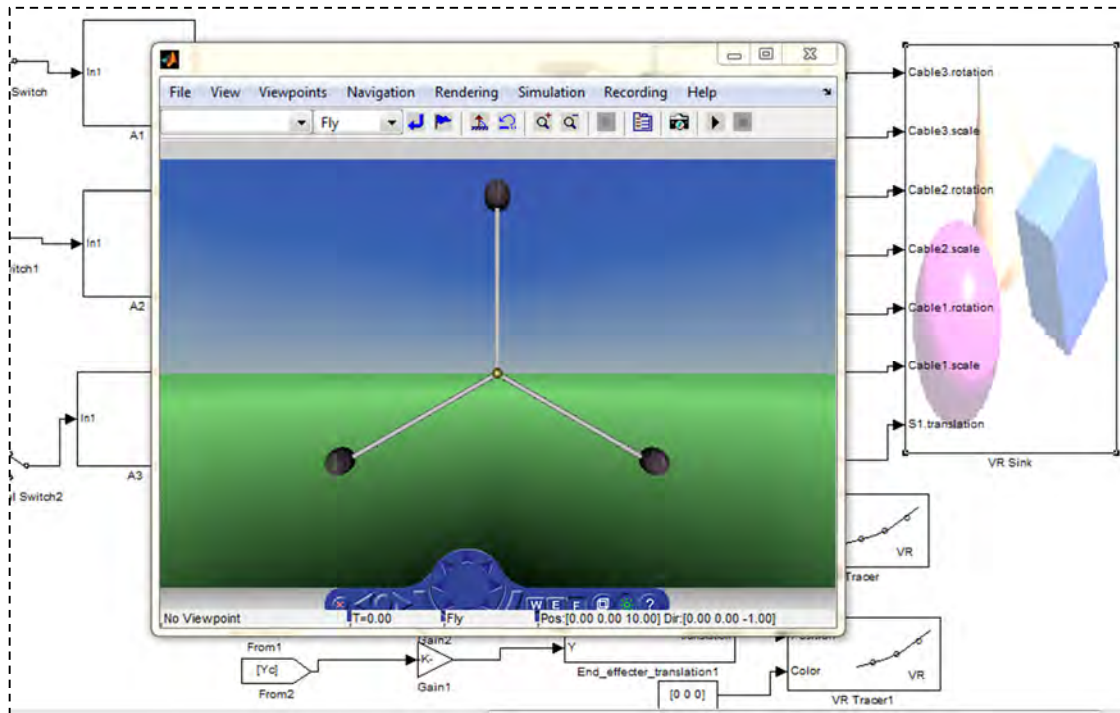


Figure 2.18- Le simulateur virtuel 3D du robot parallèle à 3 câbles

La figure 2.18 présente le simulateur virtuel 3D d'un robot parallèle à 3 câbles dans son état initial. Nous avons développé ce simulateur de réalité virtuelle au moyen du langage « VRML » « Virtual Reality Modelling Language » (figure 2.19). Il est interfacé avec Simulink-Matlab ce qui nous permettra d'acquérir des signaux provenant du prototype expérimental par des capteurs associés à l'être humain afin de déterminer les mouvements des articulations.

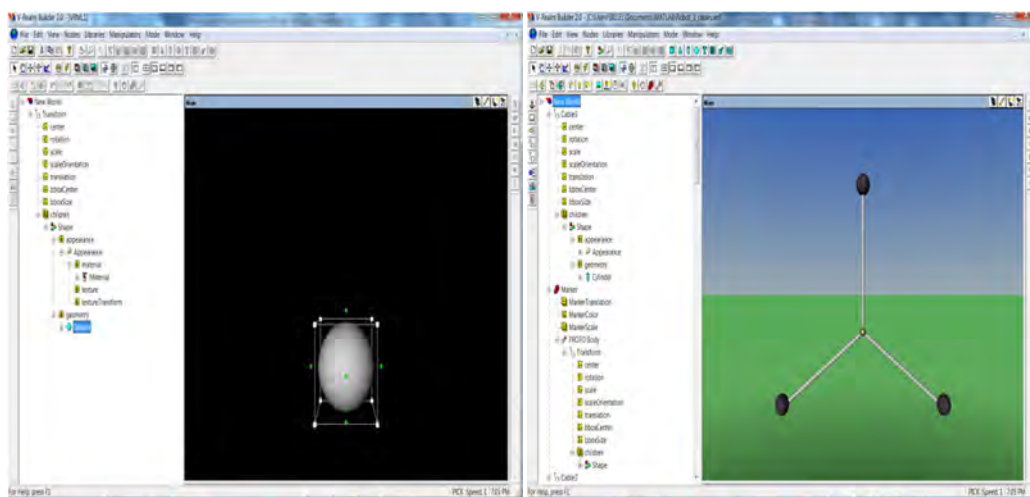


Figure 2.19- Le simulateur Virtuelle 3D dans l'environnement VRML

L'environnement VRML est interfacé avec l'environnement Simulink en utilisant le block « VR Sink » (figure 2.20). le block « Conversion Scale » permet de convertir l'échelle et les dimensions 2D à des dimensions 3D.

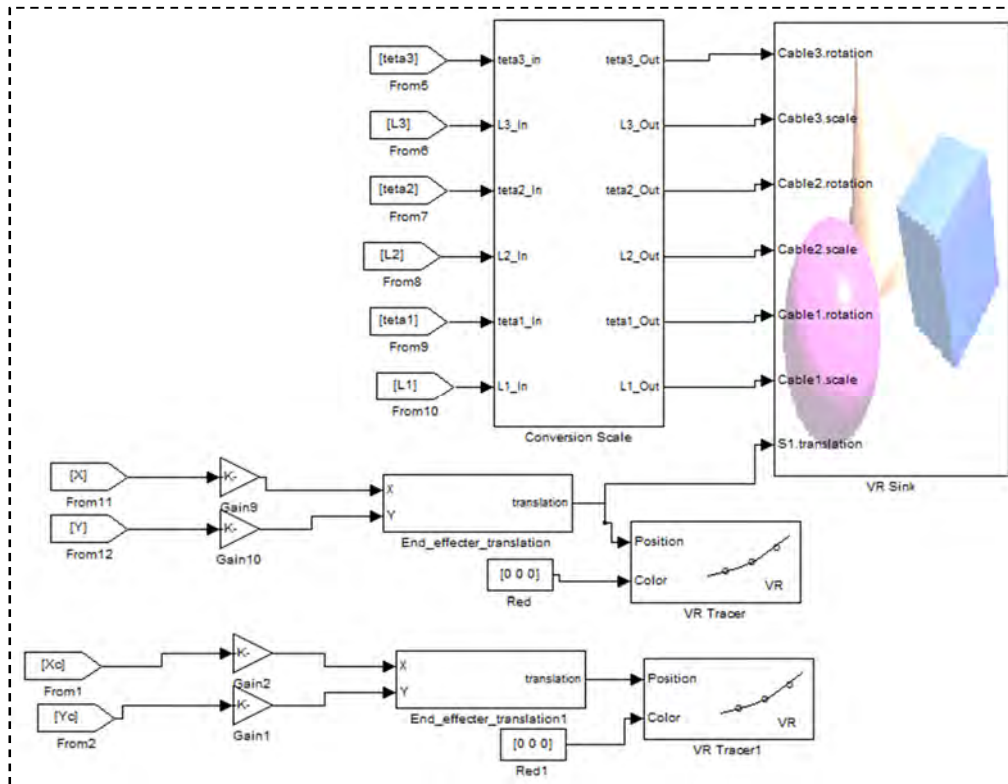


Figure 2.20- L'interfaçage du simulateur virtuel avec Simulink

La Figure 2.21 illustre le suivi d'une trajectoire circulaire en 3D en mode statique, ainsi que les longueurs des 3 câbles.

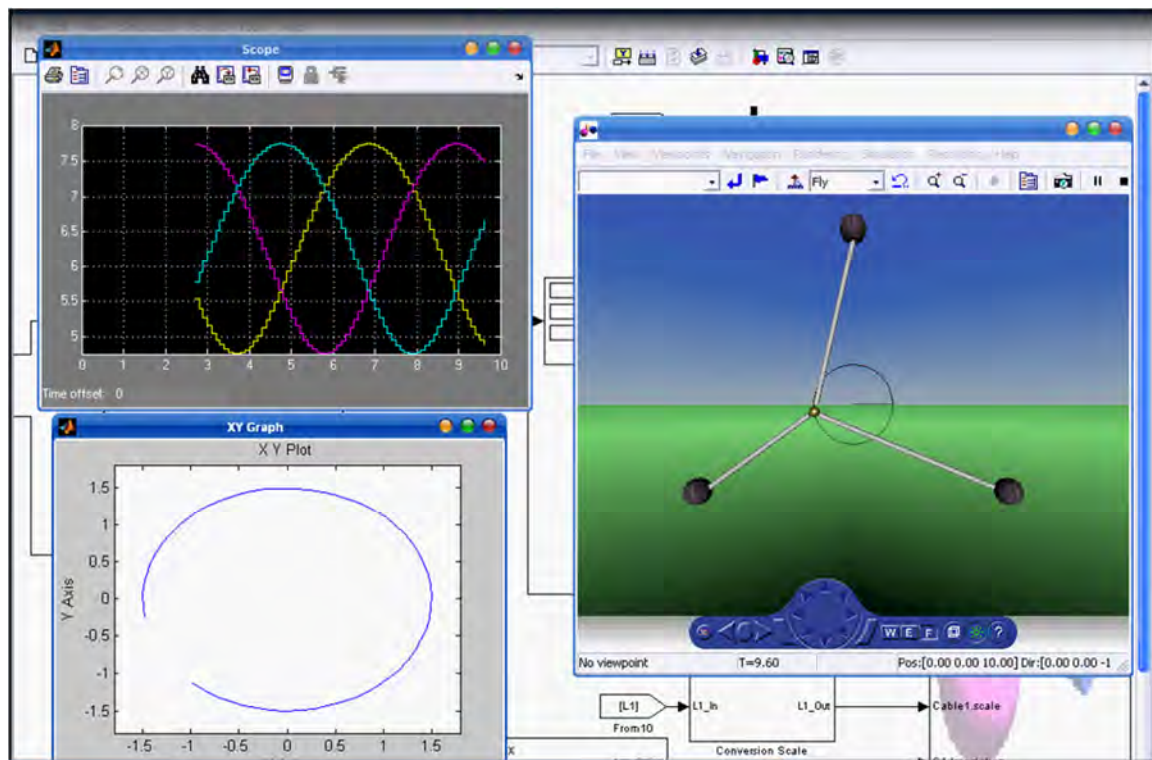


Figure 2.21- suivi d'une trajectoire circulaire en 3D en mode statique

- **Mode acquisition**

Pour le mode d'acquisition, la figure 2.22 présente l'exemple d'acquisition de la lettre « A » ; dans ce cas, le robot est en mode « équilibre dynamique » ce qui nous donne la possibilité de déplacer l'effecteur final pour l'acquisition des données pour la reproduction du geste en mode virtuel.

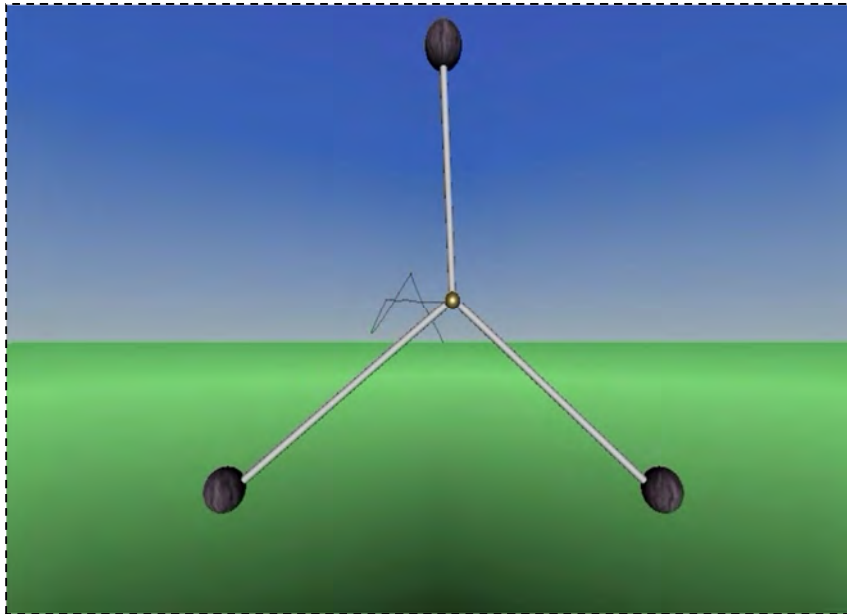


Figure 2.22- L'acquisition de la lettre « A »

- **La partie hardware**

Le prototype expérimental est divisé en 5 éléments essentiels :

- La structure mécanique qui a la forme d'un cube, configurable selon le nombre de câbles et le degré de liberté demandé (figure 2.23).



Figure 2.23- La structure mécanique configurable

- Des moteurs de types EMG30 sont utilisé pour motoriser les poulies équipé par deux encodeurs en quadrature pour pouvoir déterminer le sens de rotation des poulies, avec

une précision de 360 impulsions par tour et un facteur de réduction égale à 30 (figure 2.24).

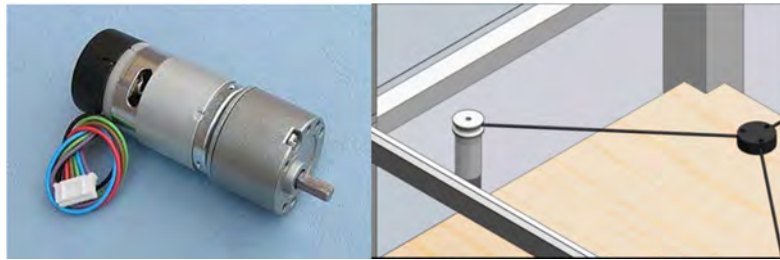


Figure 2.24- Le moteur EMG30

- Une carte d'interface de puissance à base de L293D.
- Une carte pour le contrôle de la PWM des moteurs à base des microcontrôleurs 16F877A pour chaque moteur interfacé avec l'environnement Matlab (figure 2.25).

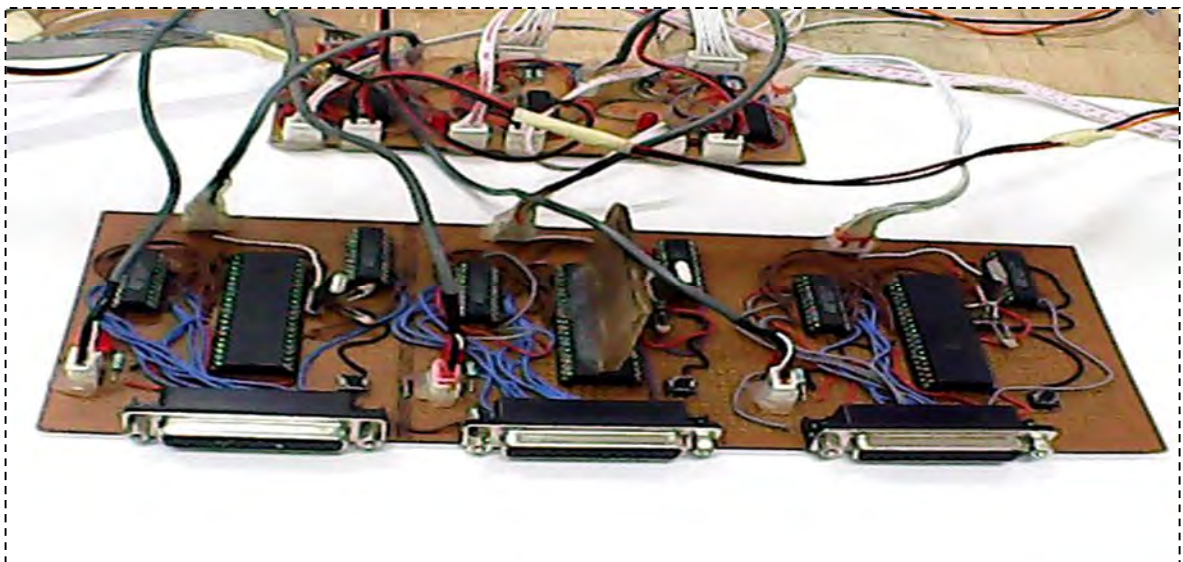


Figure 2.25- La carte de contrôle du robot à 3 câbles

- Des alimentations stabilisées de 5V pour les cartes de basse tension et 12V pour un courant de démarrage max de 9A pour les moteurs (figure 2.26).



Figure 2.26- Les sources d'alimentation du robot à 3 câbles

2.2.2 Le robot parallèle à 4 câbles

Dans cette section, nous présenterons la conception et la réalisation d'un robot parallèle à 4 câbles. Cette configuration permet d'augmenter l'espace du travail par rapport au robot à 3 câbles. D'autre part, puisque la configuration du robot à 4 câbles est planaire, cela implique une redondance dans les forces de contrôle de l'effecteur final que nous mètrons à profit pour sélectionner les solutions.

De la même façon que pour le robot parallèle à 3 câbles, nous allons procéder à l'étude du robot parallèle à 4 câbles présenté par la figure 2.27:

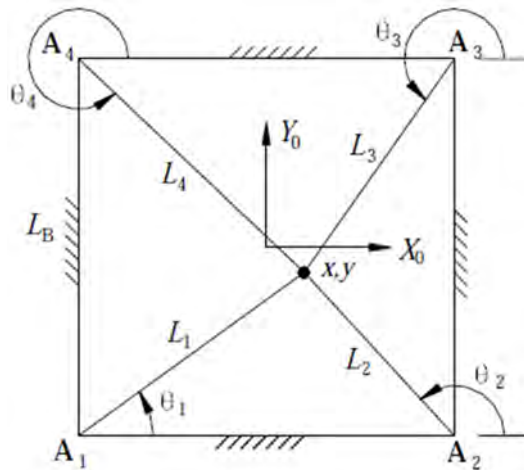


Figure 2.27- Le robot parallèle à 4 câbles

2.2.2.1 Modélisation

2.2.2.1.1 Modélisation Géométrique

En utilisant la symétrie de la structure triangulaire et rectangulaire pour la modélisation, on obtient les expressions qui donnent les modèles géométriques inverse et directe.

- Modèle Géométrique Inverse (MDI) :

$$L_i = \sqrt{(x - A_{ix})^2 + (y - A_{iy})^2} \quad \text{avec } i = 1, \dots, 4 \quad (2.33)$$

$$\theta_i = \arctang\left(\frac{y - A_{iy}}{x - A_{ix}}\right) \quad \text{avec } i = 1, \dots, 4 \quad (2.34)$$

- Modèle Géométrique Direct (MGD) :

$$x = \frac{Lb^2 + L_1^2 - L_2^2}{2Lb} \quad (2.35)$$

$$y = +\sqrt{L_1^2 - x^2} \quad (2.36)$$

2.2.2.1.2 Modélisation Cinématique

- Modèle Cinématique Inverse (MCI) :

$$\begin{pmatrix} \dot{L}_1 \\ \dot{L}_2 \\ \dot{L}_3 \\ \dot{L}_4 \end{pmatrix} = \begin{pmatrix} \cos(\theta_1) & \sin(\theta_1) \\ \cos(\theta_2) & \sin(\theta_2) \\ \cos(\theta_3) & \sin(\theta_3) \\ \cos(\theta_4) & \sin(\theta_4) \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} \quad (2.37)$$

- Modèle Cinématique Direct (MCD) :

$$\dot{X} = M^+ \dot{L} \quad \text{où} \quad M^+ = (M^T M)^{-1} M^T \quad (2.38)$$

Où :

\dot{L} : Le vecteur de vitesse des 4 câbles

M^{-1} : La matrice Jacobéenne inverse de RPCs

$\dot{X} = (\dot{x}, \dot{y})^T$: Le vecteur de vitesse de l'organe terminale

2.2.2.1.3 L'analyse statique des forces

A l'équilibre statique, la somme des forces extérieures exercées sur l'effecteur par les câbles doivent être égales à la force résultante F_R externe exercée sur l'environnement. La figure 2.28 montre la statique du corps libre pour 4 câbles [77].

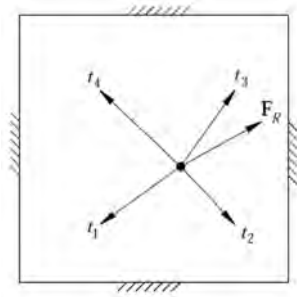


Figure 2.28- Forces statique du robot à 4 câbles

$$\sum_{i=1}^4 t_i = - \sum_{i=1}^4 t_i \vec{L}_i = F_R \quad (2.39)$$

$$\text{Où :} \quad L_i = (\cos(\theta_i) \sin(\theta_i))^T$$

Comme pour le robot à 3 câbles, la gravité est ignorée parce qu'elle est supposée perpendiculaire au plan du RPCs.

L'équation (2.39) peut être écrite sous la forme :

$$ST = F_R \quad (2.40)$$

Nous utilisons la notion de solution particulière et homogène.

$$T = S^+ F_R + (I_3 - S^+ S) z \quad (2.41)$$

Le premier terme est la solution particulière, et le deuxième terme est la solution homogène. Pour la redondance une expression équivalente à l'équation (2.41) est :

$$T = \begin{pmatrix} t_{p1} \\ t_{p2} \\ t_{p3} \\ t_{p4} \end{pmatrix} + \alpha \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{pmatrix} \quad (2.42)$$

$$N = \begin{pmatrix} \cos(\theta_2 - \theta_3 + \theta_4) - \cos(\theta_2 - \theta_3 - \theta_4) + \sin(\theta_2 - \theta_3 + \theta_4) - \sin(\theta_2 + \theta_3 - \theta_4) \\ \cos(\theta_1 - \theta_3 - \theta_4) - \cos(\theta_1 + \theta_3 - \theta_4) + \sin(\theta_1 + \theta_3 + \theta_4) - \sin(\theta_1 - \theta_3 + \theta_4) \\ \cos(\theta_1 + \theta_2 - \theta_4) - \cos(\theta_1 - \theta_2 - \theta_4) + \sin(\theta_1 - \theta_2 - \theta_4) - \sin(\theta_1 - \theta_2 + \theta_4) \\ \cos(\theta_1 - \theta_2 + \theta_3) - \cos(\theta_1 + \theta_2 - \theta_3) + \sin(\theta_1 - \theta_2 + \theta_3) - \sin(\theta_1 - \theta_2 - \theta_3) \end{pmatrix}$$

Les intervalles des angles des câbles admissibles sont:

$$0 \leq \theta_1 \leq 90^\circ, 90^\circ \leq \theta_2 \leq 180^\circ, 180^\circ \leq \theta_3 \leq 270^\circ, \text{ et } 270^\circ < \theta_4 < 360^\circ$$

2.2.2.1.4 Modélisation Dynamique

- Modèle Dynamique de l'effecteur final :

$$m \ddot{X} = F_R \quad (2.43)$$

Où la matrice cartésienne de masse $m = \begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix}$, $\ddot{X} = (\ddot{x}, \ddot{y})^T$ le vecteur d'accélération de l'organe terminal et $F_R = (F_{Rx}, F_{Ry})^T$ la force résultante de toutes les tensions des câbles appliquées sur l'organe terminal.

- Le modèle dynamique des actionneurs :

$$J \ddot{\beta} + C \dot{\beta} = \tau - rT \quad (2.44)$$

$$\text{Avec : } J = \begin{bmatrix} J_1 & 0 & 0 & 0 \\ 0 & J_2 & 0 & 0 \\ 0 & 0 & J_3 & 0 \\ 0 & 0 & 0 & J_4 \end{bmatrix}, \text{ et } C = \begin{bmatrix} C_1 & 0 & 0 & 0 \\ 0 & C_2 & 0 & 0 \\ 0 & 0 & C_3 & 0 \\ 0 & 0 & 0 & C_4 \end{bmatrix}$$

Nous considérons que tous les rayons de la poulie sont identiques ($r_i = r$; $i = 1, \dots, 4$), $\tau = (\tau_1, \tau_2, \tau_3, \tau_4)^T$ est le vecteur des couples appliqués par les moteurs, $T = (t_1, t_2, t_3, t_4)^T$ est le vecteur des tensions des câbles. $\beta = (\beta_1, \beta_2, \beta_3, \beta_4)^T$ est le vecteur des angles de rotation des poulies.

Donc :

$$T = \frac{1}{r} (\tau - J \ddot{\beta} - C \dot{\beta}) \quad (2.45)$$

- Modèle dynamique du système :

$$T = \frac{1}{r} \left(\tau - J \left(\frac{d}{dt} \left(\frac{\partial \beta}{\partial \dot{X}} \right) \dot{X} + \frac{\partial \beta}{\partial X} \ddot{X} \right) - C \frac{\partial \beta}{\partial \dot{X}} \dot{X} \right) \quad (2.46)$$

Les équations du modèle dynamique peuvent être exprimées dans une forme standard pour les systèmes robotiques [78] :

$$M(X)\ddot{X} + N(X, \dot{X}) = S(X)\tau \quad (2.47)$$

Où :

$$M = rm + S(X)J \frac{\partial \beta}{\partial \dot{X}} \quad \text{et} \quad N(X, \dot{X}) = S(X) \left(J \frac{d}{dt} \frac{\partial \beta}{\partial \dot{X}} + C \frac{\partial \beta}{\partial \dot{X}} \right) \dot{X}$$

$$M = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \quad \text{et} \quad N(X, \dot{X}) = \begin{pmatrix} N_1(X, \dot{X}) \\ N_2(X, \dot{X}) \end{pmatrix}$$

2.2.2.2 Simulation du contrôle du robot à 4 câbles

La commande proposée en boucle fermée pour le robot est la commande classique proportionnelle dérivée (PD). L'établissement de la loi de commande (U_x, U_y) selon x et selon y est :

$$\begin{cases} U_x = K_p e_x + K_d \dot{e}_x \\ U_y = K_d e_y + K_d \dot{e}_y \end{cases}$$

Nous avons simulé le contrôleur PD en tenant compte des contraintes à satisfaire sur les tensions des câbles, à savoir que les tensions doivent toujours être positives et comprise entre T_{min} et T_{max} . Les coefficients K_p et K_d sont estimés par essais et erreurs en simulation et valent :

$$K_p = 25000, \quad K_d = 100$$

Nous avons effectué la simulation de la commande point à point ainsi que celle du suivi de différentes trajectoires. Pour illustrer le rôle de notre contrôleur, nous présenterons l'exemple du suivi d'une trajectoire circulaire par l'effecteur final (figure 2.29). Nous avons utilisé les paramètres suivants : $m = 0.1$ kg; J_i ($i = 1:4$) = 0.0008 kgm²; C_i ($i = 1:4$) = 0.01 Nms; $r_i = r = 1$ cm; $L_b = 0.6$ m. Les figures 2.30 et 2.31 montrent l'évolution temporelle des longueurs des câbles et les tensions nécessaires pour assurer le suivi de la trajectoire circulaire.

Un autre exemple du suivi d'une trajectoire triangulaire est présenté par la figure 2.32. Les figures 2.33 et 2.34 montrent l'évolution temporelle des longueurs des câbles et les tensions nécessaires pour assurer le suivi de la trajectoire triangulaire.

Les résultats de simulation obtenus confirment la possibilité d'utiliser effectivement ce type de contrôle bas niveau pour commander notre robot à câbles.

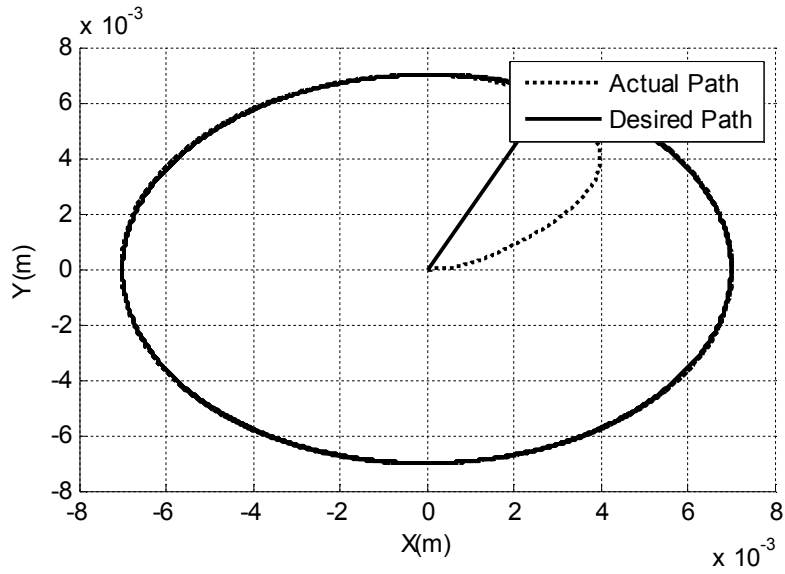


Figure 2.29- Le suivi d'une trajectoire circulaire

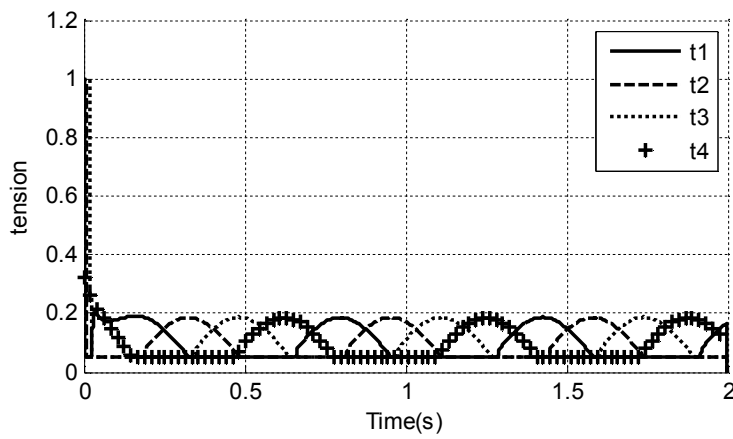


Figure 2.30- Les tensions appliquées sur les câbles

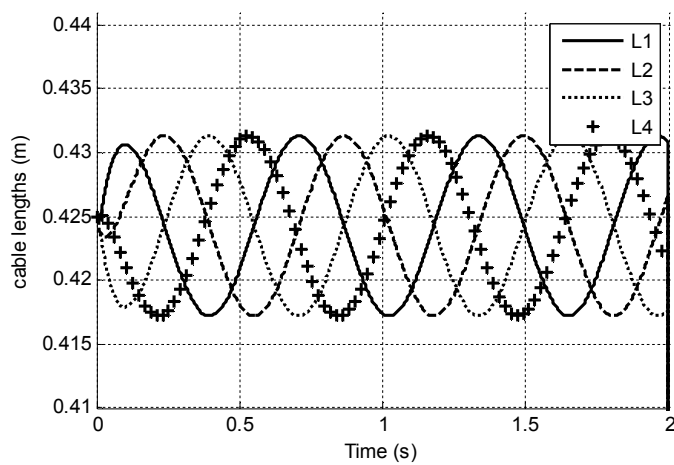


Figure 2.31- Les longueurs des câbles

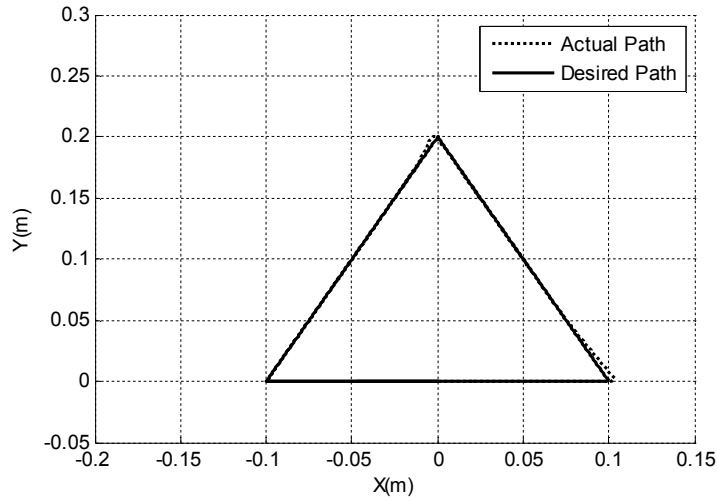


Figure 2.32- Le suivi d'une trajectoire triangulaire

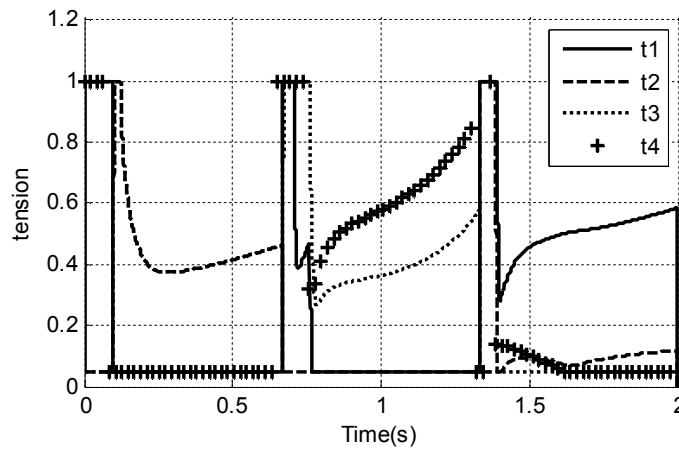


Figure 2.33– Les tensions appliquées sur les câbles

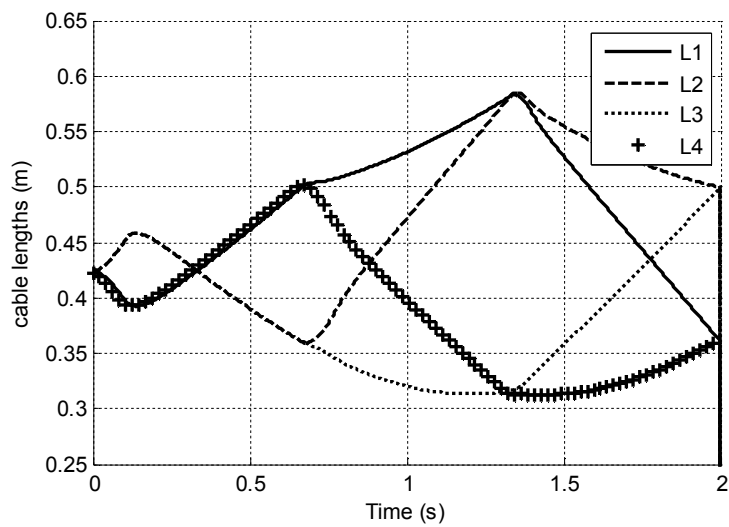


Figure 2.34- Les longueurs des câbles

L'analyse des résultats de simulation présentés précédemment confirme la possibilité d'utiliser ce contrôleur pour le suivi de la trajectoire dans le cas d'une vitesse relativement faible. Pour des vitesses relativement élevées, les tensions requises par le contrôleur ne peuvent pas être satisfaites, conduisant à retarder la réponse du système et provoquant un décalage entre les trajectoires réelles et souhaitées.

Pour illustrer ce dernier cas, nous comparons le suivi des deux courbes sinusoïdales (figure 2.35 (a) et 2.35 (b)). Nous remarquons que le suivi de la trajectoire de la figure 2.35 (a) a été effectuée avec succès en raison de la basse fréquence du signal ($\omega = 20 \text{ rad / s}$). En contraire le suivi de la courbe sinusoïdale de la Figure 2.35 (b) n'a pas été effectuée avec succès en raison de la vitesse relativement élevée ($\omega = 80 \text{ rad / sec}$).

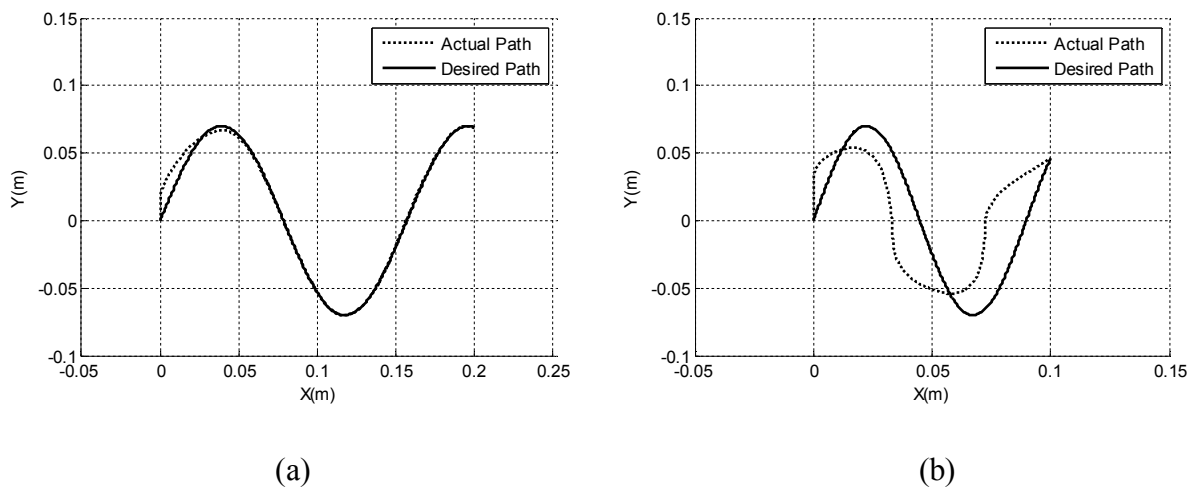


Figure 2.35-L'effet de la vitesse sur la qualité du contrôle

2.2.2.3 Réalisation expérimentale du robot parallèle à 4 câbles

Dans cette partie, nous allons présenter la conception et la réalisation d'une plateforme d'étude du robot parallèle à 4 câbles, destinée pour l'analyse de l'interaction homme robot en utilisant des modes de commande haut niveau. La réalisation du système électronique et mécanique est effectuée au niveau du laboratoire des applications de la technologie avancées LATA.

2.2.2.3.1 La structure mécanique du robot

Cette plateforme d'étude du robot parallèle à 4 câbles a été conçue principalement pour nous permettre d'effectuer une analyse sur l'interaction homme-robot pour les systèmes supervisés multimodaux. Pour cet objectif, la nécessité d'avoir implémenté plusieurs modes de commande haut niveaux (vocal, gestuel, basé sur image, ...) exige que le système interactif soit équipé de périphériques d'entrée/sortie appropriés.

Nous avons divisé cette plateforme en deux sites ; le site robot et le site operateur. La figure 2.36 présente notre plateforme d'étude (système interactif) pour le robot à 4 câbles. Nous avons placé deux caméras CCD: la première sur le site opérateur pour le mode de commande gestuel et la deuxième sur le site robot pour la commande de pointage sur image.

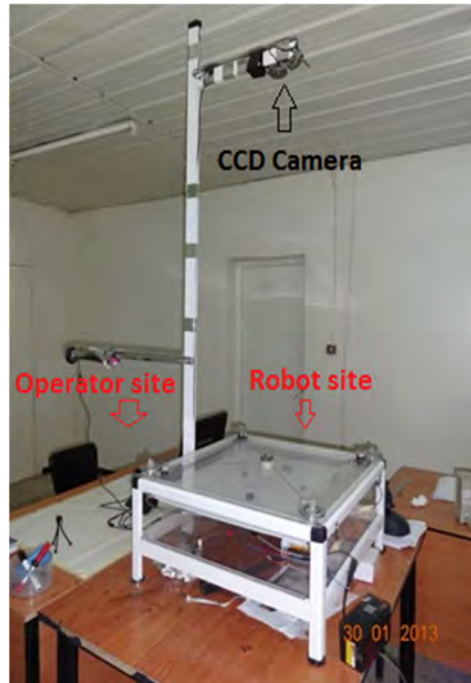


Figure 2.36- La plateforme d'étude pour le robot à 4 câbles

Nous avons adapté deux configurations possibles pour l'espace du travail. La première est présentée par la figure 2.37.a pour le mode gestuel où le plan de l'espace du travail du robot dispose d'une feuille effaçable et l'effecteur final est équipé d'un marqueur. Pour le mode de pointage sur image, nous pouvons utiliser la configuration présentée par la figure 2.37.b. Nous avons décomposé l'espace du travail pour le site robot en deux plans superposés et les objets à pointer de la banque du pointage sont placés sur le plan inférieur.



(a)

(b)

Figure 2.37- Les configurations possibles du site robot: (Commande gestuel + Commande pointage)

La figure 2.38 présente la forme choisie pour l'effecteur final qui peut être équipé d'un stylo pour le mode gestuel ou par un pointeur laser pour le mode de pointage sur image.



Figure 2.38- Effecteur final

2.2.2.3.2 La partie électronique

En ce qui concerne l'implémentation électronique, nous pouvons la diviser en deux parties : la partie interface de puissance et la partie faibles signaux.

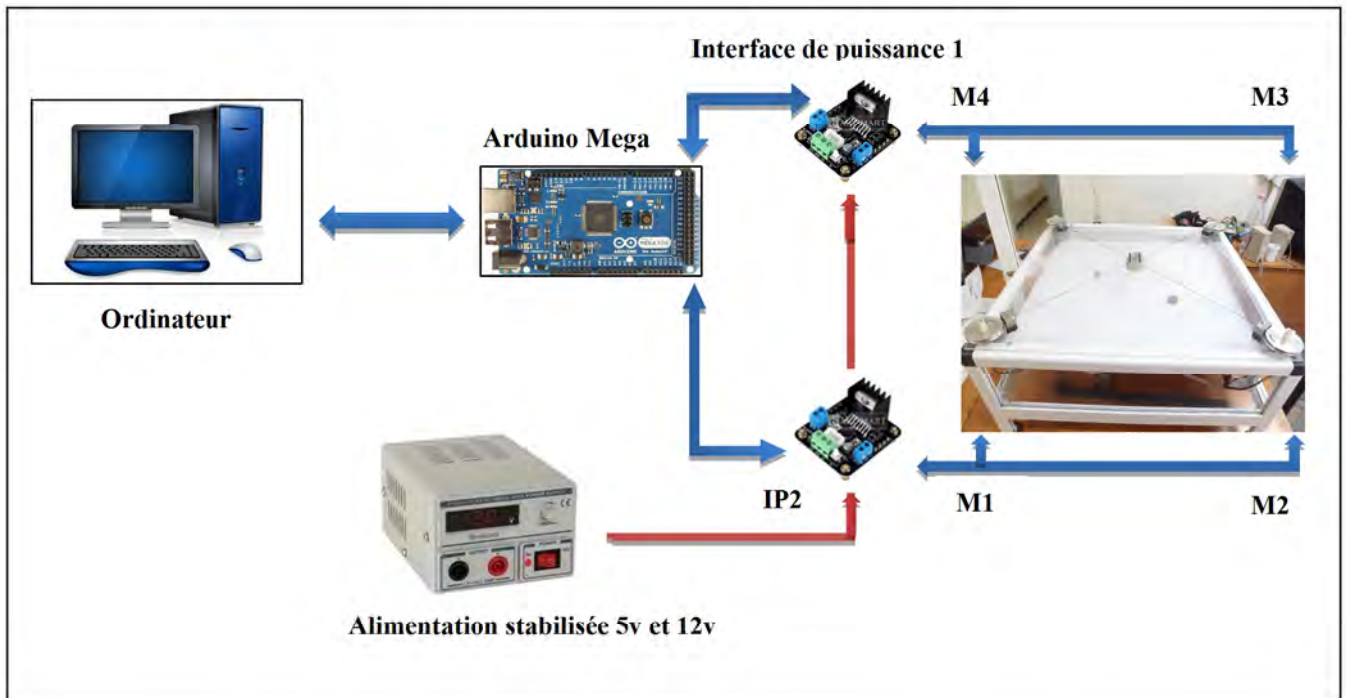


Figure 2.39- La partie électronique du robot à 4 câbles

- **La partie faibles signaux**

Le système expérimentale est piloté par un ordinateur équipé par un micro-processeur de type « CORE i7 », les processeurs multi-cœur favorisent un véritable fonctionnement multitâche. À cause de la limitation de porte d'entrées sortie du pc en termes de nombre de pins, nous avons interfacé le pc avec une carte de commande et de contrôle programmable qui s'appelle « Arduino Mega ADK » [80].

- La Carte Mega ADK pour Android

Arduino est une marque de cartes électroniques open-source basée sur une carte d'entrée-sortie et un environnement de développement qui est basé sur le langage Processing. Dans notre application nous avons utilisé une carte Arduino de type MEGA ADK (figure 2.40). L'Arduino MEGA ADK est basé sur le microcontrôleur ATmega2560 qui lui confère une puissance et un espace mémoire bien supérieur. Cette carte est réellement une carte affichant des caractéristiques impressionnantes si l'on se réfère au nombre d'entrées/sorties disponibles.

La principale différence entre une carte Arduino Mega ADK et une carte Arduino Mega2560 est que l'ADK propose en plus une interface USB compatible avec les téléphones Android. ADK signifie Android Development Kit, c'est une norme de matériel compatible avec Android éditée par Google.

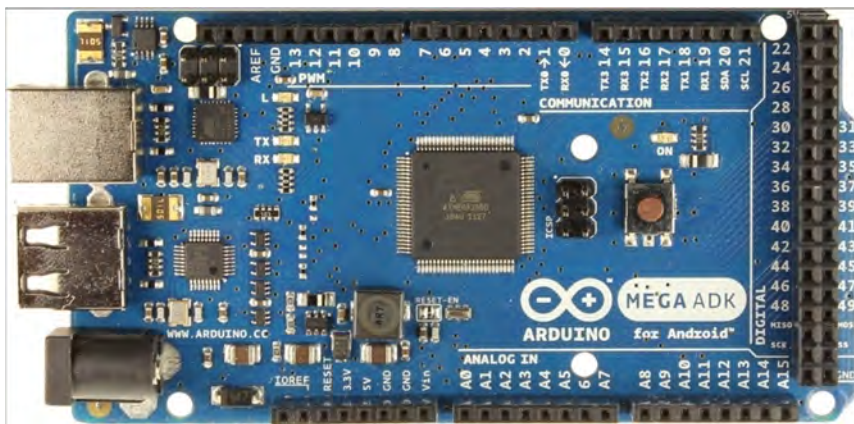


Figure 2.40- La Carte Mega ADK for Android

Pour notre application, nous avons utilisé les sorties analogiques comme sorties PWM pour contrôler les vitesses des moteurs. Les entrées numériques sont utilisées pour lire les données des capteurs. Nous pouvons noter que les informations de position sont récupérées à travers des encodeurs en quadrature à effet hall (pas de rebonds) intégrés sur les moteurs de type EMG30, nous avons détaillé le fonctionnement des encodeurs dans la partie précédente.

• **Interface de puissance pour le robot a 4 câbles :**

La carte de commande Arduino Mega ADK est interfacé avec les moteurs EMG30 par le biais des cartes de puissances. Ces cartes sont basées sur un L298N (figure 2.41) (Double pont H). Il peut contrôler deux moteurs courant continu ou un moteur pas-à-pas 4 fils 2 phases. Il est conçu pour supporter des tensions élevées, Les deux types de moteurs peuvent être contrôlés aussi bien en vitesse (PWM) qu'en direction. Toutes les sorties en puissance sont protégées par des diodes anti-retour (figure 2.42).

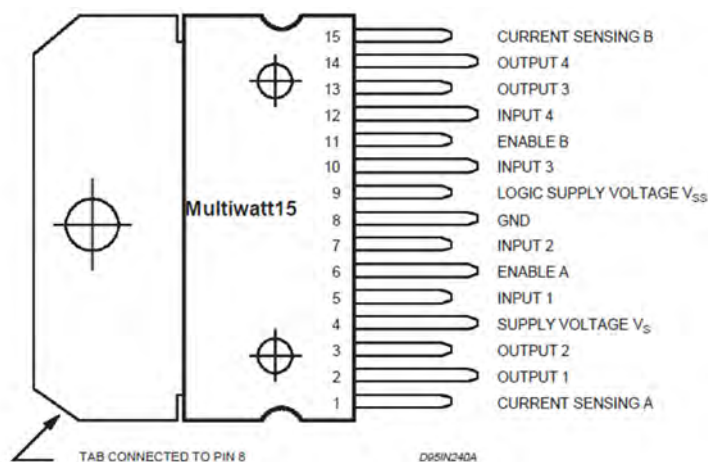


Figure 2.41- Conection des PIN pour le circuit L298N

La figure 2.41 présente les pins de connexion de L298N les pins de commande (INPUT 1, INPUT 2, INPUT 3, INPUT 4), les pins de sortie (OUTPUT 1, OUTPUT 2, OUTPUT 3, OUTPUT 4) , les deux pins d'activation (ENABLE A et ENABLE B) ainsi que l'alimentation pour le niveaux logique du circuit (Vss) et l'alimentation de l'étage de puissance (Vs).

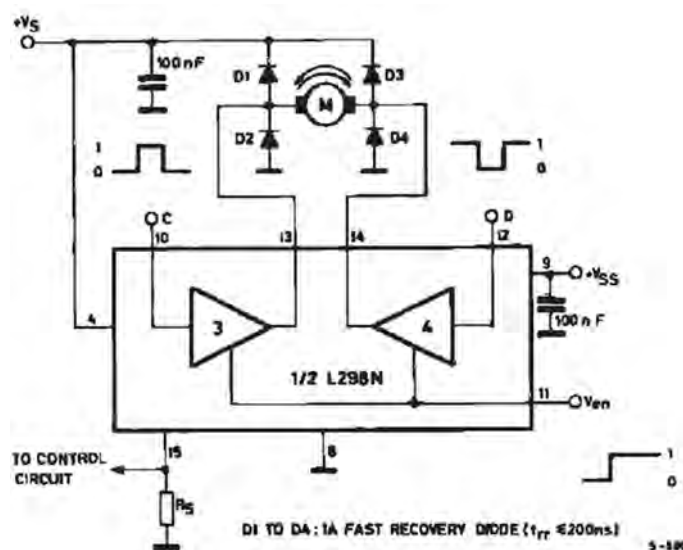


Figure 2.42- Le control bidirectionnel pour un moteur à cc

La figure 2.43 présente le câblage électronique pour contrôler un moteur à courant continu, à travers le L298N avec un pont de diodes pour la roue-libre. Ces diodes sont pour la commutation rapide.

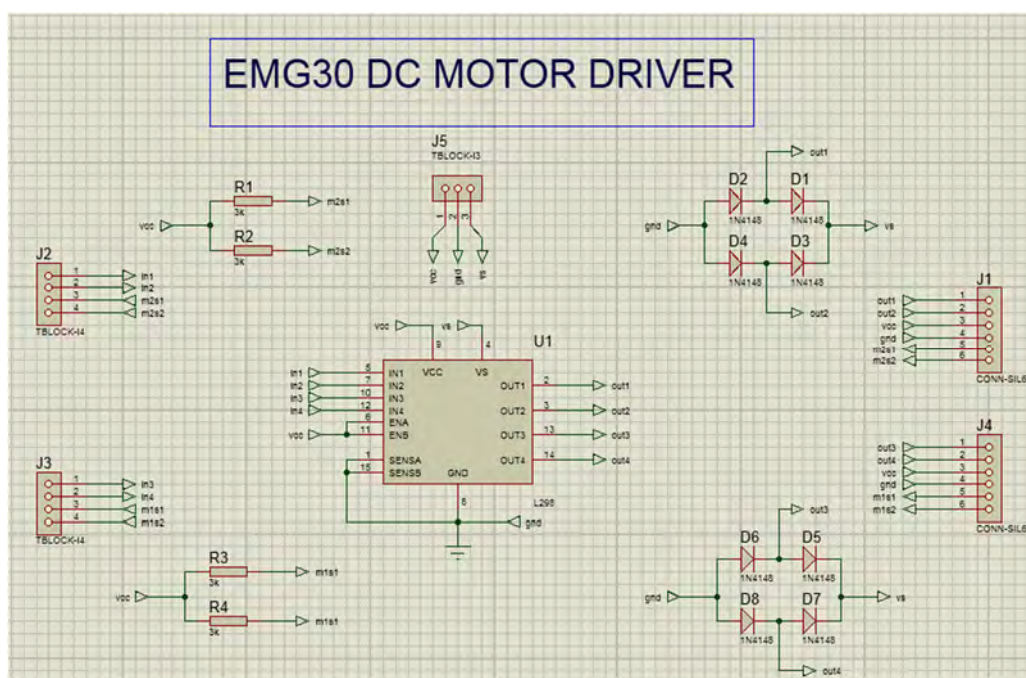


Figure 2.43- Le schéma électronique de l'interface de puissance sous PROTEUS

La figure 2.44 présente le schéma de l'interface de puissance implémenté sous PROTEUS pour le test. Le schéma est utilisé directement pour générer le circuit imprimé à l'aide du logiciel ARES (ISIS),

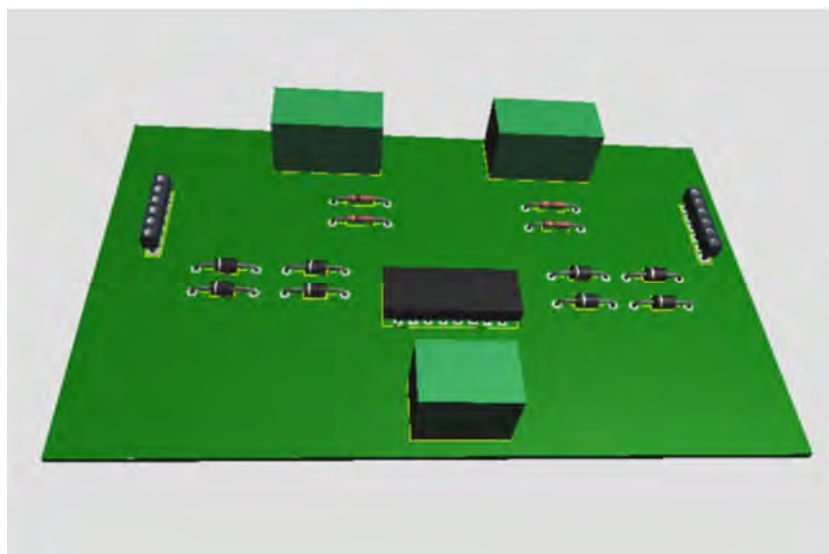
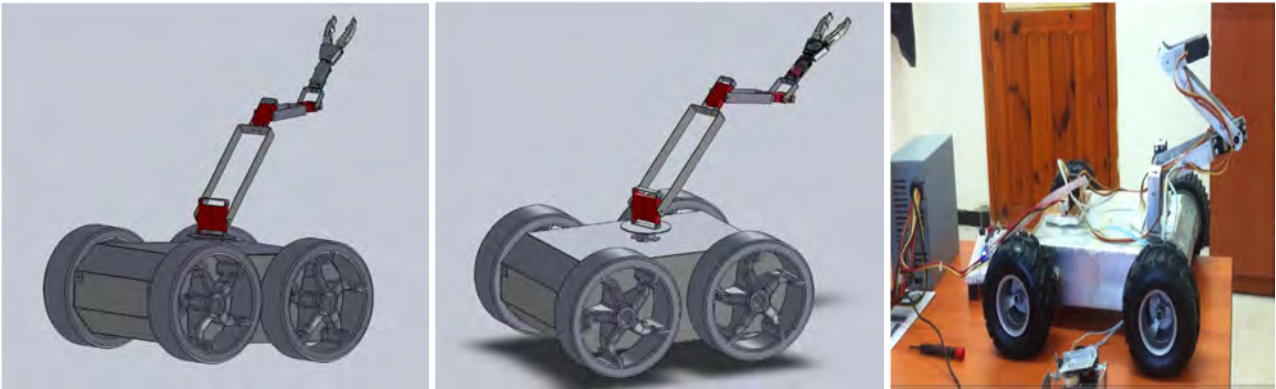


Figure 2.44- Le schéma électronique 3D et le circuit imprimé de l'interface de puissance sous ARES (ISIS)

2.3 Le robot mobile avec bras manipulateur

Dans cette partie, nous présentons l'étude et la réalisation d'un prototype expérimental d'un robot manipulateur porté sur une base mobile à 4 roues motrices « ROVER » (figure 2.45). Le bras manipulateur est de 5ddl. Nous avons divisé le robot en deux sous robots : le bras manipulateur (en haut) et la base mobile (en bas), nous avons commencé par modéliser géométriquement le bras manipulateur. Le modèle géométrique inverse est sélectionné pour commander le robot.



(a) La conception en 3D (CAD) du prototype

(b) Le prototype réel

Figure 2.45- Le prototype expérimental du robot sériel

2.3.1 La réalisation mécanique

Nous présentons la réalisation mécanique du bras manipulateur puis la base mobile (figure 2.46).



Figure 2.46- Le bras manipulateur plus la base mobile

2.3.1.1 Le bras manipulateur

C'est un bras manipulateur articulé à 5 degrés de liberté. La figure 2.47 présente le schéma 3D du bras à réaliser plus le prototype expérimental réalisé avec l'organe terminal qui est une pince prismatique.

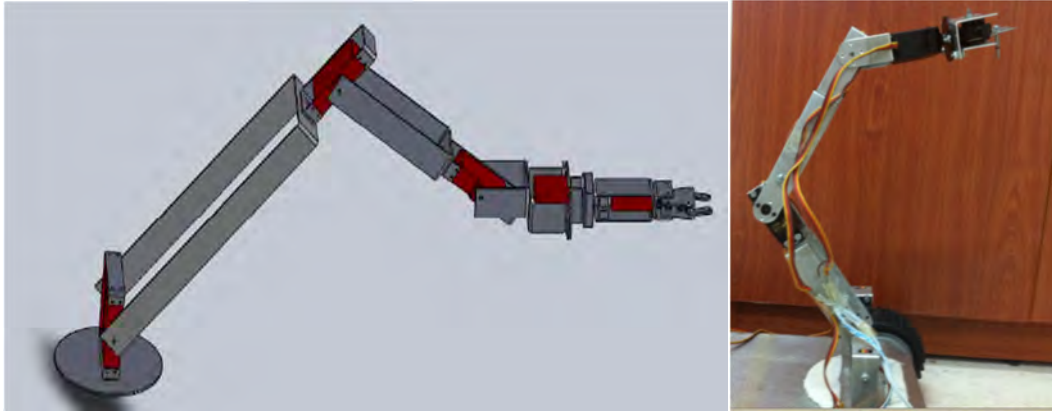


Figure 2.47- Le bras manipulateur avec pince

Ce bras manipulateur est constitué de trois segments en série reliés avec trois articulations et une pince prismatique. Des servomoteurs de type HX5010 de HEXTRONIK (figure 2.48) sont utilisés au niveau des articulations du robot. Ce servomoteur à un couple de 5.5kg/cm sous une alimentation de 4.8V et 6.5kg/cm sous 6V avec un espace de rotation de 180°.



Figure 2.48- Servomoteur de type HX5010

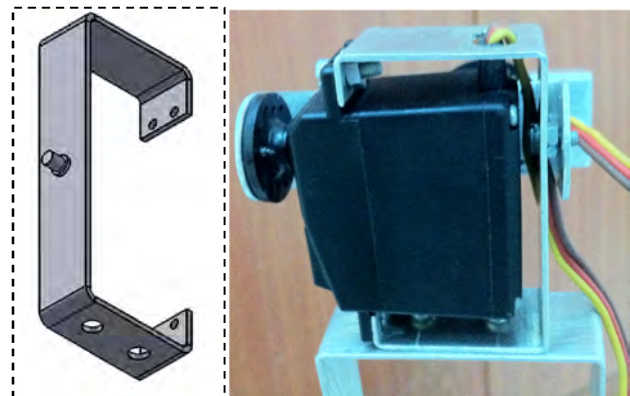


Figure 2.49- La pièce de fixation

Pour pouvoir fixer le servomoteur et connecter le segment précédent avec le suivant, une pièce en aluminium plié montré par la figure 2.49 est réalisée. Comme nous avons un seul axe de rotation sur le servomoteur, nous avons ajouté un autre axe sur la pièce pour le guidage et la robustesse de la structure (figure 2.50).



Figure 2.50- le segment du bras

La figure 2.50 présente la pièce en aluminium pliée qui réalise le segment du bras. Elle permet la fixation des deux axes du servomoteur précédant avec la partie inférieure de la pièce de la figure 2.49, fixée sur le servomoteur suivant. Cette pièce est la même utilisée pour le segment de l'effecteur final (figure 2.51). La seule différence est au niveau des dimensions, elle permet la fixation de la pince prismatique sur le bras.

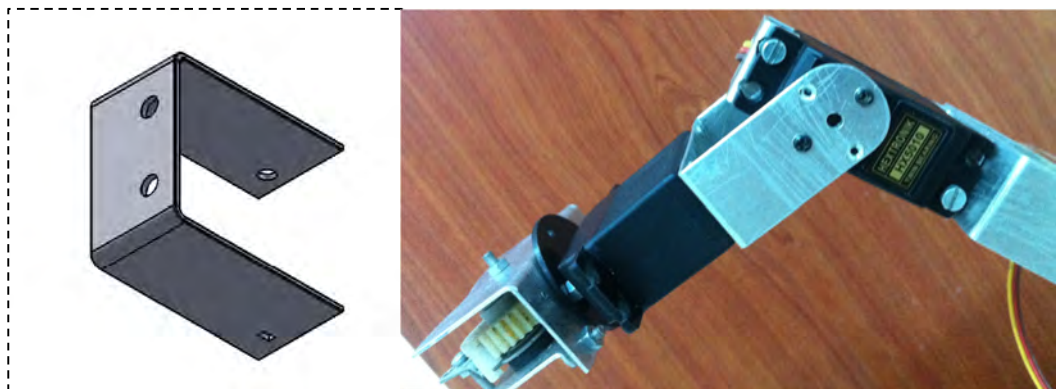


Figure 2.51- La pièce de fixation de la pince prismatique

2.3.1.2 La base mobile

Nous avons réalisé une base mobile de type « ROVER 4x4 » (figure 2.52). Chaque roue possède son propre moteur. Ce type de bases mobiles permet une bonne mobilité sur les terrains accidentés. La symétrie et la forme choisie de cette base nous donne la possibilité de l'utiliser indépendamment du bras manipulateur. Elle peut aussi être utilisée de façon symétrique dans les deux faces. Dans ce qui suit, nous allons présenter les différentes parties essentielles de cette réalisation.



Figure 2.52 La base mobile de type « ROVER »

Le boîtier de cette base mobile est réalisé en utilisant une feuille d'aluminium pliée (figure 2.52) avec une symétrie par rapport à l'axe principal. Deux tiges de stabilisation et d'équilibrage sont ajoutées près de chaque couple moteur (figure 2.53).

Les moteurs utilisés au niveau de chaque roue sont des moteurs de types EMG30 (figure 2.54). Ce moteur fournit un couple de 1.5 kg/cm avec une puissance de 4.22W. Donc pour un rover 4x4 nous pouvons déplacer une masse de 6kg en utilisant les 4 moteurs.

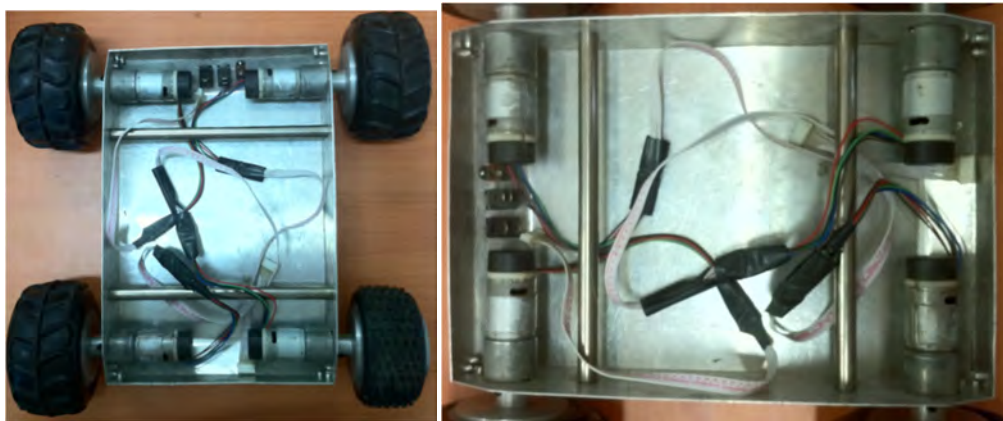


Figure 2.53- Une vue à l'intérieur de la base mobile

Nous avons ajouté une pièce en aluminium pour renforcer le couplage entre la roue et le moteur (figure 2.54).



Figure 2.54- la pièce de renforcement + la fixation de la jante

Pour les roues, nous avons utilisé des roues tout-terrain (figure 2.55).



Figure 2.55- Les roues utilisées

2.3.2 Modélisation géométrique du bras manipulateur

Notre bras manipulateur est de 5 ddl. Il est illustré dans la figure 2.56. Pour notre application, nous avons effectué les simplifications suivantes : l'angle θ_4 est en relation directe avec l'angle précédente θ_3 pour assurer que le segment de l'effecteur final soit horizontal pour une stratégie de prise d'objets. Nous noterons que l'angle θ_5 est commandé manuellement selon les dimensions de l'objet.

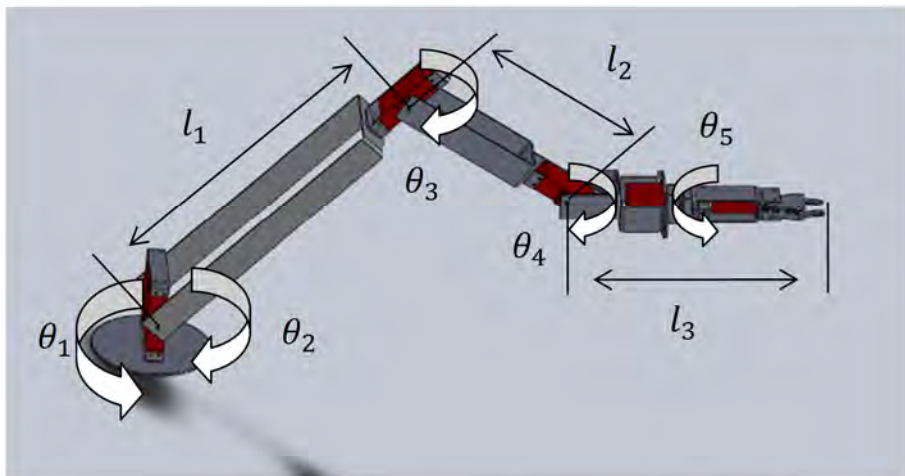


Figure 2.56- Le bras manipulateur du robot

2.3.2.1 Modèle géométrique inverse

Nous avons deux configurations possibles pour accéder au point désiré pour le coude : le coude en haut ou le coude en bas.

Pour une prise d'objets avec coude en haut :

- Calcul de θ_1, θ_2 et θ_3 :

$$\theta_1 = \text{atan2}(y, x)$$

$$\theta_2 = \text{atan2}(s_2, c_2)$$

$$\theta_3 = \text{atan2}(s3, c3) + \pi/2$$

Tel que:

$$\begin{aligned} B &= x\cos(\theta_1) + y\sin(\theta_1); & u &= -2zl_2 \\ v &= -2Bl_2; & w &= l_3^2 - l_2^2 - z^2 - B^2 \\ c2 &= (vw - u(\sqrt{(u^2 + v^2 - w^2)}))/(u^2 + v^2) \\ s2 &= (uw + v(\sqrt{(u^2 + v^2 - w^2)}))/(u^2 + v^2) \\ c3 &= (-Bs2 + zc2)/l_3; & s3 &= (-zs2 - Bc2 + l_2)/l_3 \end{aligned}$$

Pour une prise d'objets avec coud en bas :

o Calcul de θ_1, θ_2 et θ_3 :

$$\begin{aligned} \theta_1 &= \text{atan2}(y, x) \\ \theta_2 &= 2\beta - \text{atan2}(s2, c2) \\ \theta_3 &= \text{atan2}(s3, c3) + \pi/2 \end{aligned}$$

Tel que: $\beta = \text{atan2}(z, (\sqrt{x^2 + y^2}))$

Dans ce qui suit, nous présenterons quelques exemples de commande de position du robot en utilisant la commande joystick :

- **Z = 218, X = 82, Y = 0**



Figure 2.57- Configuration du prototype, simulateur 3D

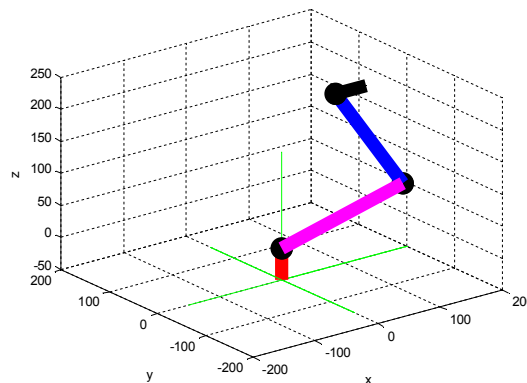


Figure 2.58- Simulation Matlab

- $Z = 218, X = 333, Y = 0$:

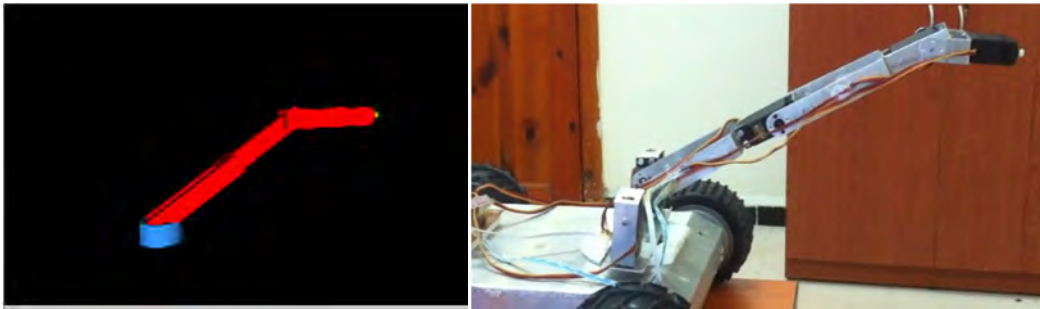


Figure 2.59- Configuration du prototype, simulateur 3D

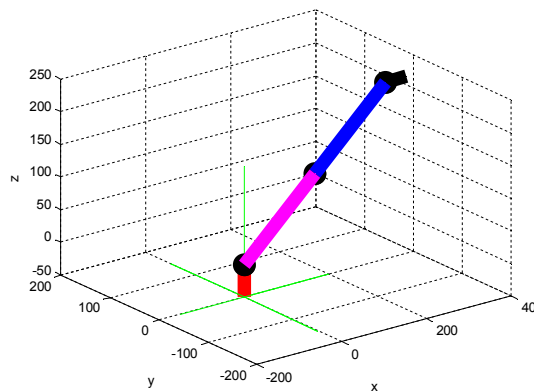


Figure 2.60- Simulation Matlab

2.4 Conclusion

Dans ce chapitre, nous avons présenté la conception, la simulation, la réalisation, ainsi que le contrôle bas niveau de type PD pour les différents prototypes expérimentaux réalisés au niveau de notre laboratoire. Ces prototypes ont servi à analyser le fonctionnement et l'adaptabilité de l'interaction homme-robot pour chaque mode de commande sur ces systèmes.

Dans la première partie, nous avons présenté la réalisation de deux robots parallèles : un robot à 3 câbles et un robot à 4 câbles. Nous avons exploité les spécificités de leurs caractéristiques, à savoir l'avantage de bénéficier d'un grand espace de travail en comparaison avec les manipulateurs parallèles classiques. En plus, les robots à câbles ont une structure légère qui a de bonnes propriétés dynamiques; un rapport masse/ charge utile élevé et un coût de construction réduit; une bonne portabilité; une structure facile à monter, démonter, stocker et transporter. Ces propriétés rendent ce type de robots très intéressants pour envisager des applications originales.

Dans la seconde partie, nous avons présenté la réalisation d'un bras manipulateur à 5ddl porté par une base mobile.

Cette variété de systèmes, nous donne la possibilité d'analyser la dépendance et l'adaptabilité de l'interface multimodale par rapport aux prototypes expérimentaux utilisés.

Chapitre 3

Implémentation des Modes de Contrôle Haut Niveau

Sommaire

3.1	Introduction	82
3.2	La commande vocale.....	82
3.2.1	Détection des limites d'un mot.....	82
3.2.2	La méthode MFCC (Mels Frequency Cepstrum Coefficients).....	86
3.2.2.1	L'échelle des Mels.....	86
3.2.2.2	Création d'un cepstres à partir des coefficients MFCC.....	88
3.2.3	Quantification vectoriel	93
3.2.3.1	Conception de l'algorithme LBG	94
3.2.3.2	Classifieur.....	95
3.2.3.3	Résultats expérimentales	96
3.2.4	L'utilisation de la bibliothèque VOCE.....	97
3.2.4.1	Fichiers de grammaire.....	97
3.2.4.2	Post-traitement du texte reconnu	98
3.2.4.3	Références des classes pour la librairie VOCE.....	99
3.2.4.4	Les problèmes rencontrés.....	100
3.3	La commande gestuelle.....	100
3.2.1	Implémentation de la commande gestuelle.....	100
3.4	La commande par pointage sur image.....	105
3.4.1	Implémentation de la commande de pointage sur image.....	105
3.4.1.1	La procédure de calibration de la camera.....	108
3.4.1.2	Exemples d'applications	110
3.5	La commande joystick.....	113
3.6	Conclusion.....	114

3.1 Introduction

Récemment, certaines recherches dans le domaine de l'interaction homme-machine s'orientent vers l'étude des systèmes supervisés qui se caractérisent par l'intégration de plusieurs modes de commandes (systèmes multimodaux). Ces systèmes sont contrôlés au moyen d'une interface graphique utilisateur appelée interface multimodale. Cette interface est celle qui offre au superviseur plusieurs modes de commande pour lui permettre d'interagir avec un système robotique de façon aisée pour accomplir des tâches et des missions plus ou moins complexes. Des exemples de tels modes de commande sont la commande par la voix, la commande par télémanipulation au moyen d'un joystick, la commande basée sur l'image, la commande gestuelle, la commande physiologique, etc.

Dans le but d'analyser l'interaction multimodale homme-robot, nous présentons dans ce chapitre l'implémentation de quatre techniques de contrôle haut niveau. Chaque mode de contrôle sera illustré par la description d'un test préliminaire. On commencera dans l'ordre par la commande vocale, la commande gestuelle, la commande par pointage sur image et la commande par joystick.

3.2 La commande vocale

Dans cette partie, nous présentons l'implémentation de la commande vocale selon deux méthodes. La première méthode consiste à implémenter une technique de création des coefficients cepstraux selon une échelle biologique modélisant la perception de l'oreille humaine appelée Mel. La seconde, consiste à utiliser une bibliothèque prédéveloppée appelée VOCE exploitant les points forts des logiciels existants : un synthétiseur TTS et Sphinx4 pour la phase de reconnaissance.

3.2.1 Détection des limites d'un mot

Il existe un grand nombre de variantes d'algorithme de détection des limites d'un mot [23, 81, 82]. L'algorithme utilisé ici est décrit par les trois points suivants (figure 3.1):

- 1- Nous avons un début de mot dès que l'amplitude moyenne

$$\bar{a}_m = \frac{1}{K} \sum_{k=1}^K a_m(k) \quad (3.1)$$

Est supérieure à un certain Seuil s : $a_m(k)$ est l'amplitude du signal vocal du k eme canal, échantillonné à l'instant m .

- 2- Nous avons une fin de mot, si on a déjà détecté un début de mot, et si l'amplitude moyenne \bar{a}_m reste inférieure au seuil s pendant une durée supérieure à T_{sil} (typiquement 200 ms).
- 3- La durée d'un mot doit être au moins égale à T_{min} (typiquement 200 ms) et au plus égale à T_{max} (typiquement 2000 ms).

Le paramètre T_{min} permet de rejeter un bruit pour autant qu'il dure moins que T_{min} . Le paramètre T_{sil} permet d'avoir des silences au sein d'un mot, si leurs durées sont inférieures à T_{sil} (par exemple, silence avant les plosives).

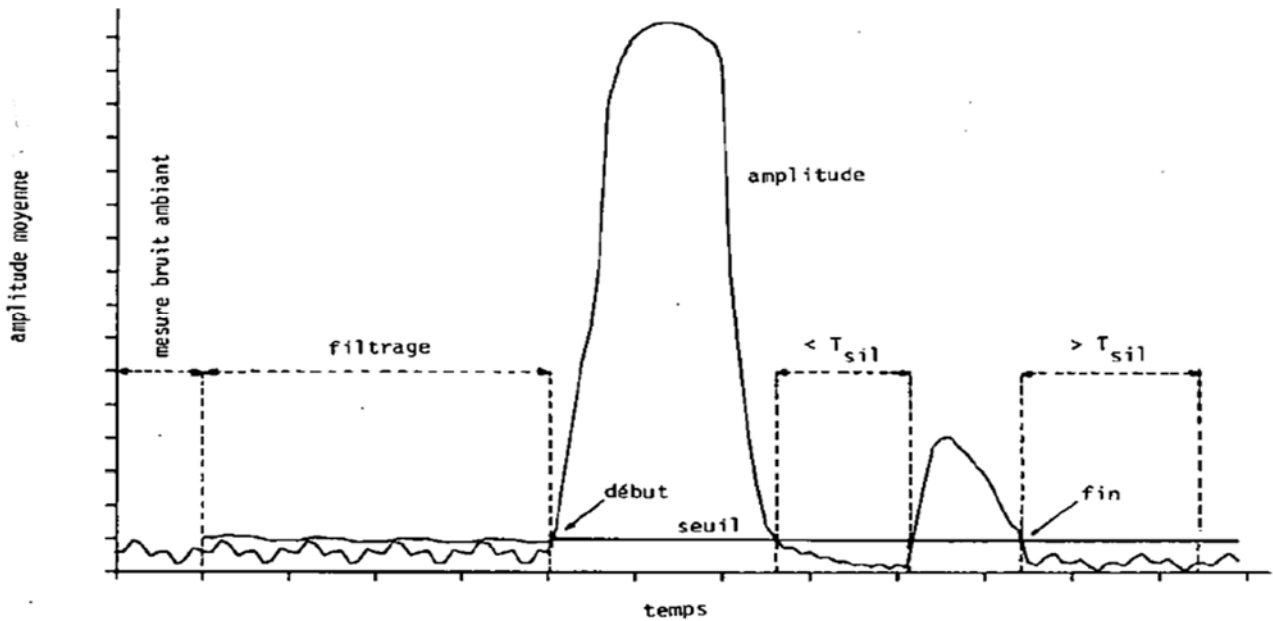


Figure 3.1- Détection de début et de fin de mot

Le seuil s doit être adapté au bruit ambiant. Pendant une certaine durée, on mesure la moyenne du niveau de l'amplitude du signal du bruit ambiant :

$$s_{m_0} = \frac{1}{m_0} \sum_{m=1}^{m_0} \bar{a}_m \quad (3.2)$$

Où m_0 est choisi égal à la durée de la phase de silence, le seuil est ensuite déterminé à l'aide de la relation suivante :

$$s'_m = \alpha s_{m-1} + (1 - \alpha) \bar{a}_m \quad (3.3)$$

Ce qui correspond à un filtre numérique passe-bas. En outre, on fixe la valeur minimale de s_m à une valeur s_{min} :

$$s_m = \text{Max}(s'_m, s_{min}) \quad (3.4)$$

Nous avons alors un début de mot dès que :

$$\bar{a}_m > (1 + \beta) s_m \text{ avec } \beta = 0.5 \quad (3.5)$$

L'algorithme de détection de début et de fin de mot est présenté par la figure 3.2.

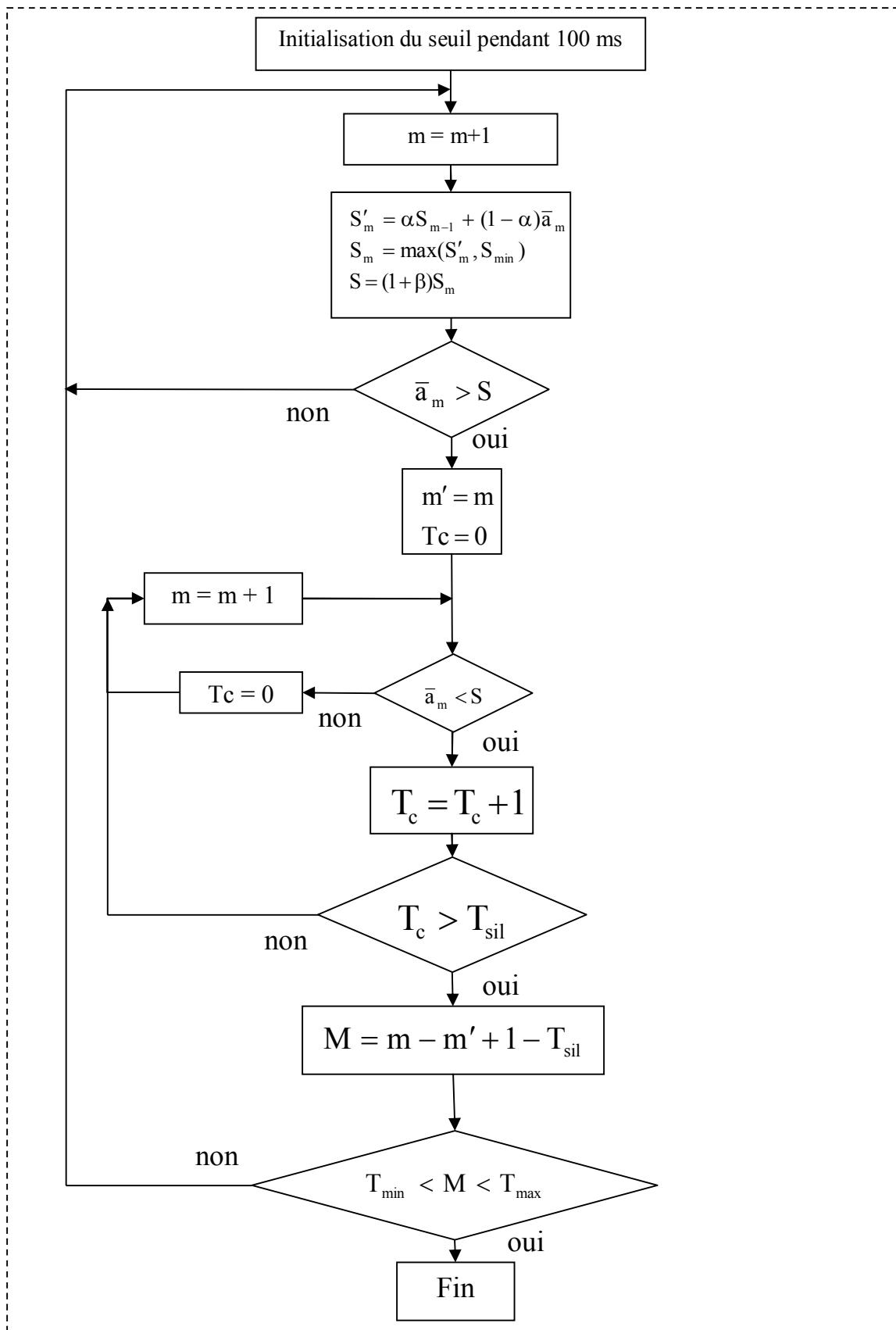


Figure 3.2-L'algorithm de détection de début et de fin de mot

Dans notre cas, nous avons adapté cet algorithme en utilisant deux seuils logiques T_H et T_L dont :

$$T_H = \begin{cases} 1 & \text{si } \bar{a}_m > s_m \\ 0 & \text{si } \bar{a}_m < s_m \end{cases} \quad \text{et} \quad T_L = \begin{cases} 1 & \text{si } \bar{a}_m > s_{min} \\ 0 & \text{si } \bar{a}_m < s_{min} \end{cases}$$

La condition de détection est devenue comme suit : nous avons un début de mot si nous avons déjà détecté que $T_L = 1$, et si $T_H = 1$ pendant une durée T_{mot} tel que $T_{min} < T_{mot} < T_{max}$

Dans ce qui suit, nous allons présenter deux exemples de détection de début et de fin, le premier concernant un exemple de deux lettres prononcées séparément et le deuxième pour un mot complet (plusieurs lettres concaténées).

La figure 3.3 présente la détection de début et de fin des deux lettres « A » et « B » dont : le signal « commande » représente le signal vocale original parasité, le signal « Début Fin » représente le signal logique de la détection, le signal « Mots détectés » représente la multiplication du signal originale avec le signal logique de la détection dont nous avons récupéré seulement les deux lettres.

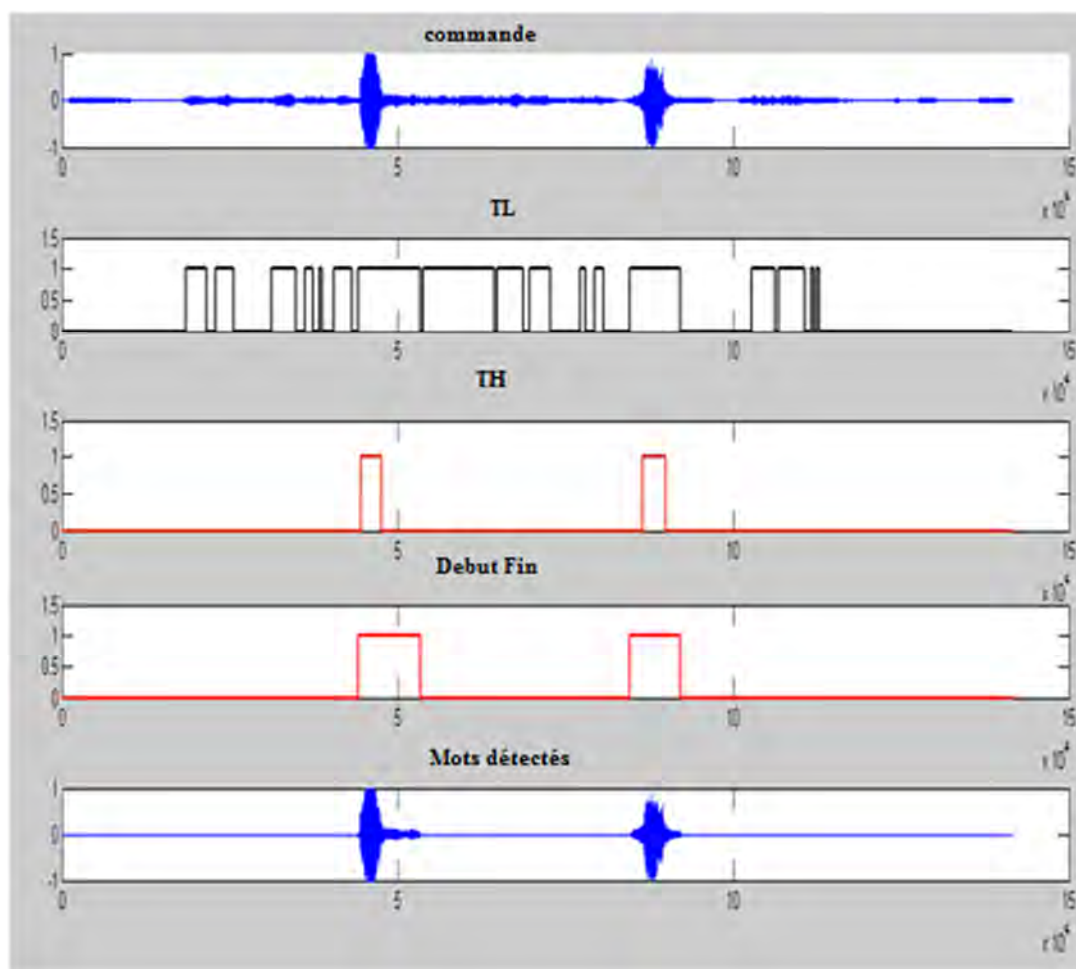


Figure 3.3- Détection des limites des lettres « A » et « B »

Le deuxième exemple est présenté par la figure 3.4 pour la détection de début et de fin du mot « START ».

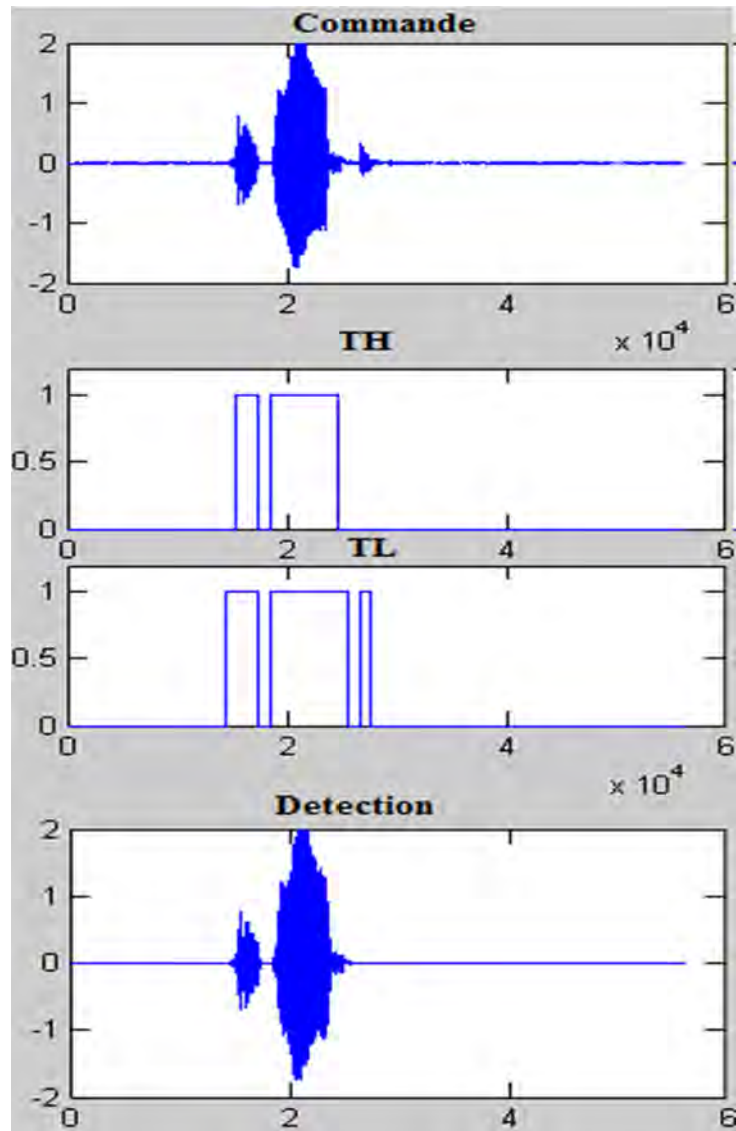


Figure 3.4-Détection de début et de fin du mot « START »

3.2.2 La méthode MFCC (Mels Frequency Cepstrum Coefficients)

3.2.2.1 L'échelle des Mels

L'échelle des Mels est une échelle biologique. C'est une modélisation de l'oreille humaine. A noter que le cerveau effectue en quelque sorte une reconnaissance vocale complexe avec filtrage des sons. Prenons l'exemple suivant où vous êtes à table en compagnie de nombreuses personnes, l'ensemble de ses personnes parle en même temps et vous discutez avec votre voisin. Malgré le bruit, vous arrivez à discerner clairement ce que vous dit votre voisin, vous ignorez de façon naturelle le bruit de fond et vous amplifiez le son qui vous paraît le plus important. Vous pouvez répéter cette expérience avec chacun des convives. Le cerveau ne se contente non pas seulement de filtrer les sons

et de les amplifier mais aussi de prédire. Prenons l'exemple suivant où une personne discute avec vous avec un volume sonore très bas, vous n'avez pas entendue une certaine partie de la phrase mais vous arrivez à la reconstituer et à la comprendre.

A partir de l'étude du cerveau nous pouvons nous faire une idée de la complexité de la reconnaissance vocale et nous pouvons nous rapprocher d'un modèle de plus en plus puissant et parfait.

On considère que l'oreille humaine perçoit linéairement le son jusqu'à 1000 Hz, mais après, elle perçoit moins d'une octave par doublement de fréquence. L'échelle de Mels modélise assez fidèlement la perception de l'oreille : linéairement jusqu'à 1000 Hz, puis logarithmiquement au-dessus [83].

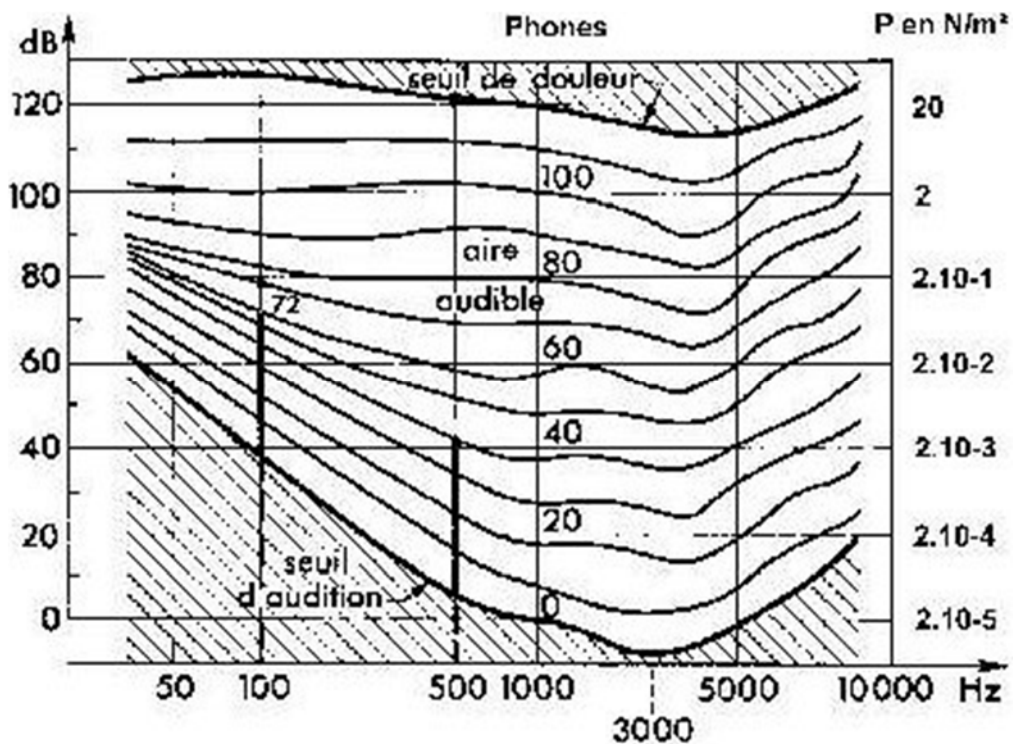


Figure 3.5- Diagramme de Fletcher Décrit les seuils de perception et lignes d'isotonies selon les fréquences

On voit bien que la perception n'est pas linéaire. La formule donnant la fréquence en Mels m à partir de celle en Hz f est :

$$m = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right) \quad (3.6)$$

L'échelle des Mels permet donc de modéliser une perception de l'oreille linéairement (figure 3.6).

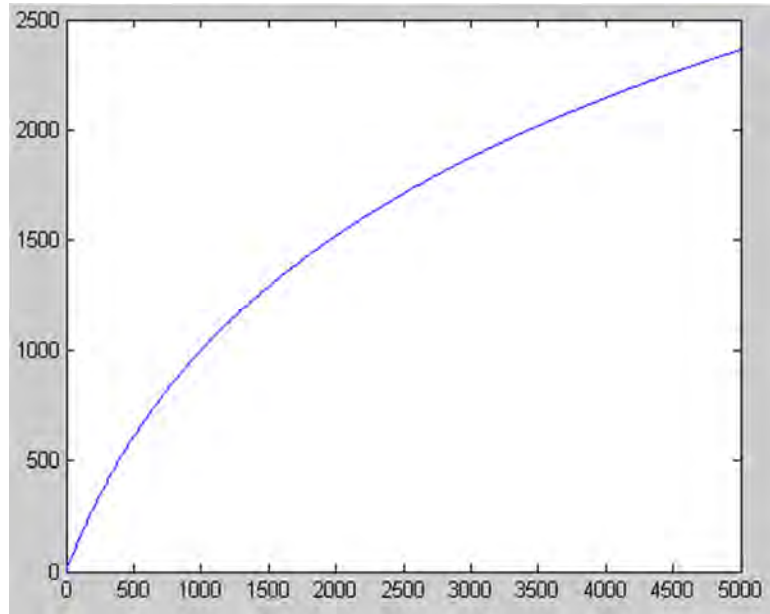


Figure 3.6- La courbe de Mels

3.2.2.2 Création d'un cepstres à partir des coefficients MFCC

Notre reconnaissance vocale fonctionne à partir de comparaison entre cepstres MFCC. Plusieurs étapes sont nécessaires pour transformer un fichier audio en cepstres MFCC (figure 3.7). Nous détaillons dans cette section la construction pas à pas de ces coefficients. Le principal intérêt des MFCC est d'extraire des informations pertinentes et en nombres limités en s'appuyant à la fois sur la production théorique (théorie cepstrale) et à la fois sur la perception de la parole (échelle des Mels) [84].

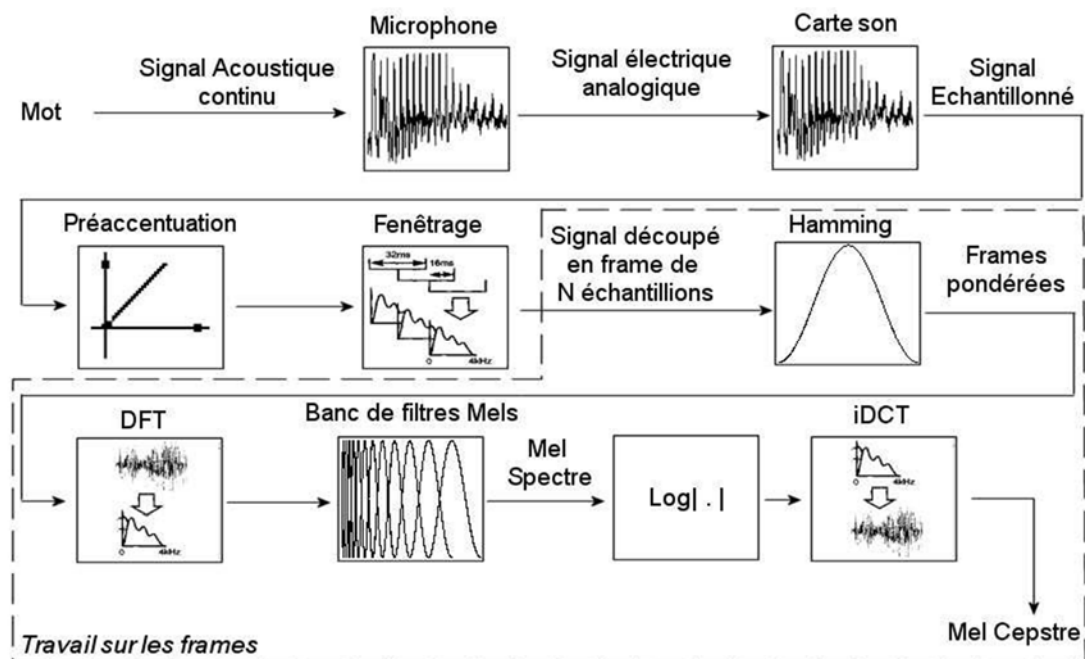


Figure 3.7- Représentation des étapes de création des MFCC

3.2.2.2.1 Prétraitement

On considère à partir de maintenant que le signal d'un mot a été numéroté (ou échantillonné). Il est représenté à partir de maintenant par une famille $(X_n)_{n \in [1, M]}$ où M est le nombre d'échantillons dans le signal. Chaque élément de la famille est un réel compris entre -1 et 1 (en effet, pour un codage sur 16 Bits, les valeurs sont codées sur 15 Bits, plus un bit pour le signe, ce qui donne des valeurs possibles entières entre -32768 et 32767, ramenée à des valeurs réelles entre -1 et 1). Le signal est échantillonné (par la carte son sur un ordinateur) à une fréquence F_s (communément 11 025 ou 22 050 Hz), c'est-à-dire que en une seconde, 11 025 ou 22 050 points ont été créés.

Nous considérons par la suite que le signal de la parole est délimité de manière à bien commencer par le début d'un mot, qu'il se termine sans trop de blanc par la fin de ce mot et qu'il n'y a pas de parasite (enregistrement idéal sans bruit en laboratoire).

3.2.2.2.2 Fenêtrage

Nous découpons le signal en fenêtres. Le signal est découpé en tranches de 2^n échantillons appelées frames ou encore fenêtres qui ont la particularité de se recouvrir de moitié dans l'objectif d'avoir un meilleur traitement pour FFT (Fast Fourier Transform). On utilise typiquement une fenêtre de $N = 512$ échantillons ou un nombre qui est une puissance de 2, cela vient du fait que l'algorithme FFT que nous utilisons est bien plus rapide pour ces nombres. Dans notre programme principal nous utilisons des fenêtres de 512 échantillons.

3.2.2.2.3 Application d'une fenêtre de pondération

Une fenêtre de pondération est appliquée à chaque frame (figure 3.8), ceci est dans l'objectif d'harmoniser les échantillons pour permettre un meilleur traitement pour l'algorithme FFT. En effet FFT ne donne pas de bons résultats quand une pente trop importante est détectée dans une partie du signal. La fenêtre de pondération a pour objectif de minimiser les erreurs produites par FFT.

Fenêtre de pondération Hamming : $0.54 - 0.46 \times \cos\left(\frac{2\pi n}{N-1}\right), n \in [0, N - 1]$

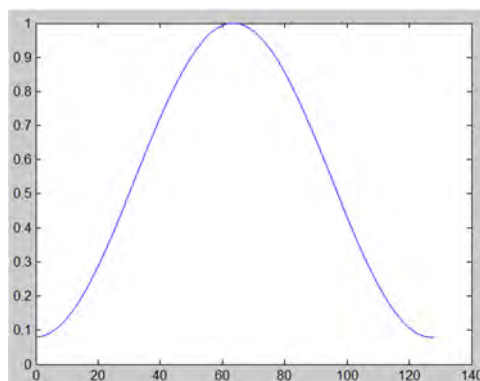


Figure 3.8- Courbe de la fonction de pondération de Hamming sur 128 points

3.2.2.2.4 La FFT (Fast Fourier Transform)

La transformée de Fourier rapide (Fast Fourier Transform - FFT) a été introduite en 1960. C'est un algorithme très puissant mais qui possède la contrainte de ne pouvoir utiliser que des séries de 2^n nombres.

Cette étape consiste à prendre chaque fenêtre et à appliquer la transformée de Fourier, on convertit ainsi chaque frame du domaine temporel en domaine fréquentiel. La DFT (Discrete Fourier Transform) sur le $i^{\text{ème}}$ frame est définie ainsi :

$$Y_{i,n} = \sum_{k=1}^N Y_{i,k} e^{\left(\frac{-2\pi(n-1)(k-1)}{N}\right)}, \text{ avec } n \in [1, N] \text{ et } j^2 = -1 \quad (3.7)$$

Avant de transformer les nombres complexes en amplitudes et en fréquences compréhensibles, nous devons faire une mise à l'échelle, pour cela on applique la formule suivante en rapport à l'image ci-contre :

$$r = \sqrt{a^2 + b^2} \quad \text{avec } a \text{ les valeurs réelles et } b \text{ les valeurs complexes}$$

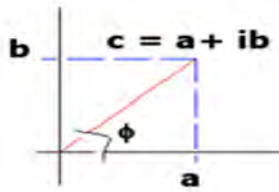


Figure 3.9- Mise à l'échelle du résultat de l' FFT

Ensuite nous remarquons que l'FFT donne deux fois les mêmes informations, en effet sur 512 points on retrouve de façon inversée la même chose sur les 512 points suivants. Nous allons donc supprimer les 512 points qui ne nous intéressent pas (figure 3.10).

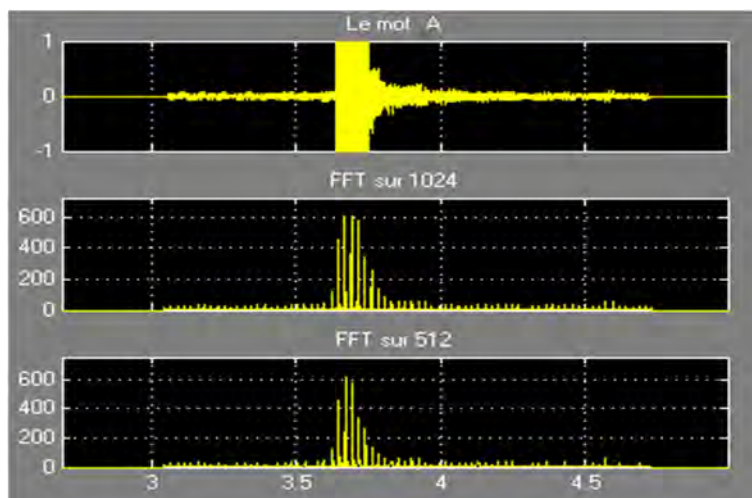


Figure 3.10- La FFT du mot A

Un spectre en 3D se présente de cette façon, l'axe X représentant l'axe des temps, l'axe Y, l'axe des fréquences et enfin l'intensité des couleurs représentant l'amplitude des spectres. Nous avons prendre l'exemple d'un son court parasité qui représente le mot « A » (figure 3.11), le spectre suivant nous permet de bien traiter l'ensemble des informations généré par FFT :

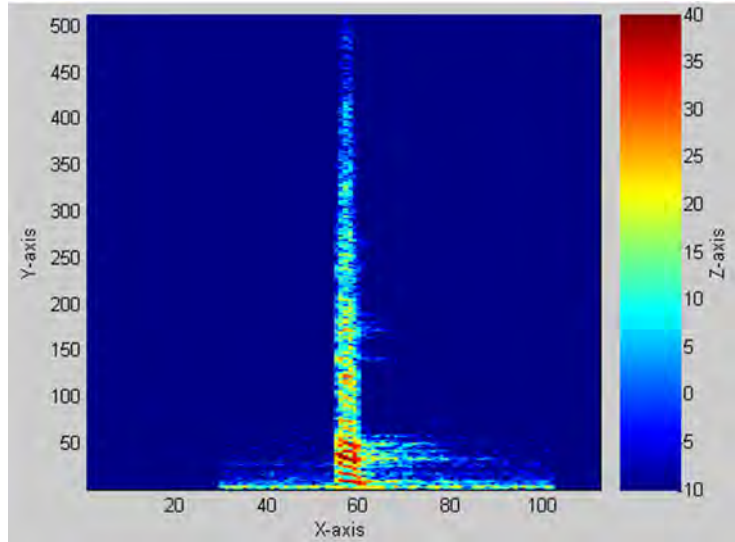


Figure 3.11- Spectre 3D du mot « A »

3.2.2.2.5 Banc de filtres Mels

Comme l'étendue des fréquences présentes dans le spectre est encore très large, nous avons donc beaucoup de données à traiter, on a recours au banc de filtres dans l'échelle de Mels. On relie ainsi le système de reconnaissance vocale au fonctionnement de l'oreille humaine. Il s'agit de filtres passes bandes triangulaire centrés linéairement dans le domaine fréquentiel des Mels et de largeur telle qu'ils divisent l'espace des fréquences de manière égale dans le domaine des Mels et qu'ils se recouvrent chacun par moitié. Les filtres sont donc disposés logarithmiquement dans l'échelle des fréquences usuelles (Hz) (figure 3.12), ainsi nous avons beaucoup de filtres pour les basses fréquences alors que les hautes fréquences sont disposées plus largement.

Nous rappelons la formule donnant la fréquence en Mels à partir de la fréquence en Hz :

$$m = 2595 \times \log_{10} \left(1 + \frac{f}{700} \right) \text{ avec } f \text{ la fréquence en Hertz} \quad (3.8)$$

Chaque filtre va donner un coefficient cepstrale :

$$S_{1,k} = \sum_{n=0}^{N/2} Y_{i,n} M_{n,k}, \quad k = 0, \dots, K \quad (3.9)$$

Avec : $Y_{i,n}$ le $n^{\text{ième}}$ coefficient de la transformée du $I^{\text{ième}}$ frame.

$M_{n,k}$ Le $n^{\text{ième}}$ coefficient du $k^{\text{ième}}$ filtre.

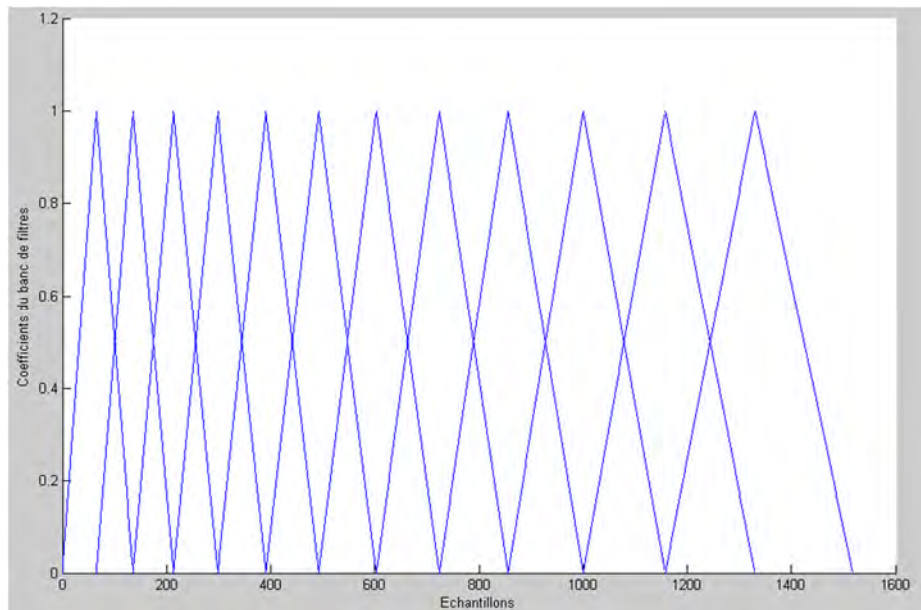


Figure 3.12- Représentation d'un banc de filtres dans l'échelle des Mels

3.2.2.2.6 Les coefficients Cepstraux

C'est l'étape finale, nous transformons les données dans l'échelle des Mels vers l'échelle des temps. Le résultat de cette étape sera les MFCC proprement dit (figure 3.13). Il suffit d'effectuer l'inverse de la transformée de Fourier. Dans la pratique, on effectue une Transformée en Cosinus Discrète Inverse (TCDI), ce qui revient au même puisque la transformée en Cosinus inverse donne la partie réelle de la transformée de Fourier ; or ici on a que des réels. Il faut noter que la transformée en sinus donnera la partie imaginaire de la transformée de Fourier.

$$C_{i,n} = \sum_{k=1}^K \left(\log_{10}(S_{i,k}) \right) \cdot \cos\left(n \left(k - \frac{1}{2}\right) \cdot \frac{\pi}{K}\right) \quad (3.10)$$

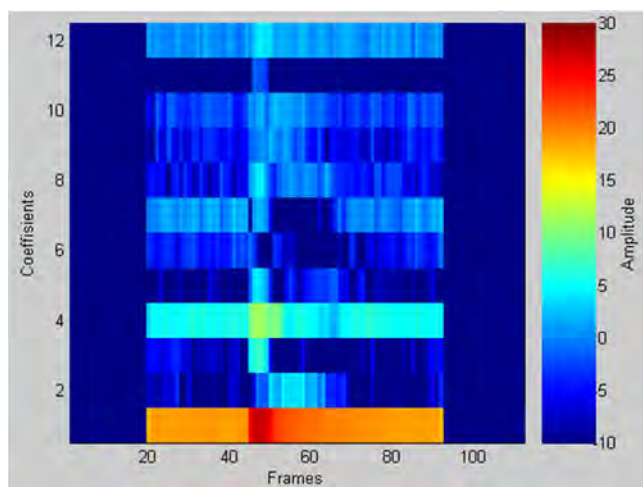


Figure 3.13- La représentation 3D des coefficients Cepstraux du mot « A »

3.2.3 Quantification vectoriel

La quantification vectorielle est un moyen efficace qui permet de compresser de grands nombre de vecteurs d'entraînement à l'aide d'un dictionnaire. Les dictionnaires contiennent la représentation numérique des caractéristiques qui sont propres du locuteur spécifié. Le dictionnaire est généré dans la phase d'apprentissage, en regroupant les vecteurs caractéristiques de chaque locuteur (comme indiqué dans la figure 1.5. Dans la phase de reconnaissance, les paroles d'entrée sont vecteur quantifié et la distorsion VQ (Victor Quantization) qui est calculée sur tout l'énoncé est utilisée pour déterminer l'identité du locuteur.

Il existe de nombreux types d'algorithme de génération de dictionnaire, mais le plus connu et le plus utilisé est l'algorithme K-means [85, 86] et le Linde, Buzo et Gray (LBG) algorithme [87].

La quantification vectorielle (QV) est une méthode de compression de données avec perte basée sur le principe de codage par blocs. Il s'agit d'un algorithme de longueur fixe. Au début, la conception d'un quantificateur vectoriel (QV) était considérée comme un problème difficile en raison de la nécessité d'une intégration multidimensionnelle. En 1980, Linde, Buzo, et Gray (LBG) a proposé la conception d'un algorithme de QV basée sur une séquence d'apprentissage. L'utilisation d'une séquence d'apprentissage contourne la nécessité d'une intégration multidimensionnelle. Un QV qui est conçu à l'aide de cet algorithme sont mentionnés dans la littérature comme une LBG-VQ.

Une QV est une approximation dont l'idée est similaire à celle du « arrondi » (disant à l'entier le plus proche). Un exemple de QV 1-dimensionnel est indiqué ci-dessous (figure 3.14):

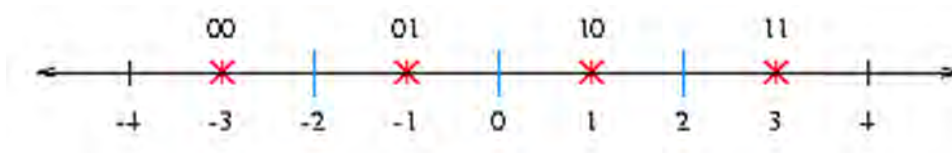


Figure 3.14- QV 1-dimensionnel

Ici, tous les nombres inférieurs à -2 sont estimés par -3. Chaque nombre entre -2 et 0 est estimé par -1. Chaque nombre entre 0 et 2 est estimés par +1. Chaque nombre supérieur à 2 est estimé de +3. Notez que les valeurs approximatives sont uniquement représentées par 2 bits. Il s'agit d'un 1-dimensionnelle, la QV a un taux de 2 bits / dimension.

Un exemple de QV de 2-dimensions est indiqué ci-dessous:

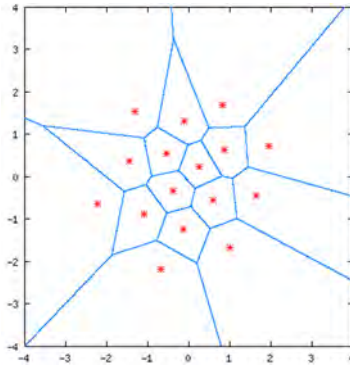


Figure 3.15- Un exemple de QV de 2-dimensions

Ici, chaque paire de nombres tombent dans une région donnée est approché par une étoile rouge associé à cette région. Notez qu'il ya 16 régions et 16 étoiles rouges - dont chacun peut être uniquement représentés par 4 bits. Ainsi, il s'agit d'un 2-dimensionnelle, 4-bit QV. Son taux est également de 2 bits / dimension.

Dans les deux exemples ci-dessus, les étoiles rouges sont appelés vecteurs de codes et les régions définies par les bordures bleues sont appelés les régions codées. L'ensemble de tous les vecteurs de code est appelé le dictionnaire de codes et l'ensemble de toutes les régions de codage est appelé l'espace de la partition.

3.2.3.1 Conception de l'algorithme LBG

L'algorithme LBG est un algorithme itératif qui résout alternativement ces deux critères d'optimalité, l'algorithme nécessite un dictionnaire de codes initial $\mathcal{C}^{(0)}$. Ce dictionnaire de codes initial est obtenu selon la méthode de fractionnement (splitting). Dans cette méthode, un vecteur de code initial est défini comme la moyenne de la séquence d'apprentissage entière. Ce vecteur de code est ensuite divisé en deux. L'algorithme itératif est exécuté avec ces deux vecteurs en tant que dictionnaire initial. Les deux derniers vecteurs de code sont divisés en quatre, et le processus est répété jusqu'à ce que le nombre souhaité de vecteurs de codes soit obtenue. L'algorithme est résumé ci-dessous.

$$\mathcal{T} = \{x_1, x_2, \dots, x_M\} \quad (3.11)$$

1. \mathcal{T} fixe $\varepsilon > 0$ très petit nombre
2. affecter $N = 1$ et $c_1^* = \frac{1}{M} \sum_{m=1}^M x_m$ calculer $D_{ave}^* = \frac{1}{M} \sum_{m=1}^M \|x_m - c_1^*\|^2$
3. Fractionnement (splitting): pour $i = 1, 2, \dots, N$,

$$c_i^{(0)} = (1 + \varepsilon)c_i^* \quad (3.12)$$

$$c_{N+i}^{(0)} = (1 - \varepsilon)c_i^* \quad (3.13)$$

Met $N = 2N$

4. Itération: $D_{ave}^{(0)} = D_{ave}^*$ affecter la valeur 0 a $i = 0$.

i. pour $m = 1, 2, \dots, M$ trouver la valeur minimale de : $\|x_m - c_n^{(i)}\|^2$

Pour $n = 1, 2, \dots, N \cdot n^*$

$$Q(x_m) = c_{n^*}^{(i)} \quad (3.14)$$

ii. pour $n = 1, 2, \dots, N$, mise a jour du vecteur de code

$$c_n^{(i+1)} = \frac{\sum Q(x_m)=c_n^{(i)} x_m}{\sum Q(x_m)=c_n^{(i)}} \quad (3.15)$$

iii. Affecter $i = i + 1$

$$\text{Calculer } D_{ave}^{(i)} = \frac{1}{Mk} \sum_{m=1}^M \|x_m - Q(x_m)\|^2 \quad (3.16)$$

v. Si $\left(\frac{D_{ave}^{(i-1)} - D_{ave}^{(i)}}{D_{ave}^{(i-1)}}\right) > \varepsilon$, aller a l'étape (i).

vi. Affecter $D_{ave}^* = D_{ave}^{(i)}$. pour $n = 1, 2, \dots, N$,

Affecter $c_n^* = c_n^{(i)}$ comme étant le vecteur de code final.

5. Répéter les l'étape 3 et 4 jusqu'a l'obtention du nombre désiré du vecteur de code.

3.2.3.2 Classifieur

Pour pouvoir comparer le mot de commande avec le mot de référence pendant la phase de reconnaissance, nous avons implémenté un Classifieur basé sur la distance euclidien. La figure 3.16 représente le schéma global de notre système de reconnaissance vocale en deux phases: la phase d'apprentissage et la phase de reconnaissance.

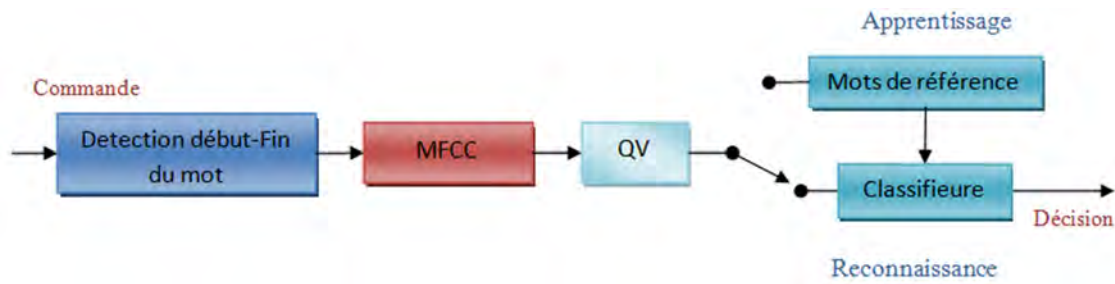


Figure 3.16- Le Classifieur

3.2.3.3 Résultats expérimentaux

Pour notre système de reconnaissance vocale, le traitement est appliqué avec une résolution pour l'FFT à 1024 points. Le résultat est une matrice des coefficients MFCC de dimension $(12 \times j)$, où la valeur de j dépend de la durée du mot prononcé, de la fréquence d'échantillonnage de la carte son et de la résolution de l'FFT. Le système est testé sur un dictionnaire de 4 commandes (START, STOP, UP, DOWN). La matrice MFCC est compressée en une matrice de (12×32) . Pour un opérateur donné, nous avons les dimensions suivantes:

Tableau 3.1– Dimensions des matrices MFCC et QV

Commande	MFCC	QV
START	12×128	12×32
STOP	12×128	12×32
UP	12×128	12×32
DOWN	12×128	12×32

La figure 3.17 représente les coefficients MFCC et leurs Centroides pour le mot "START".

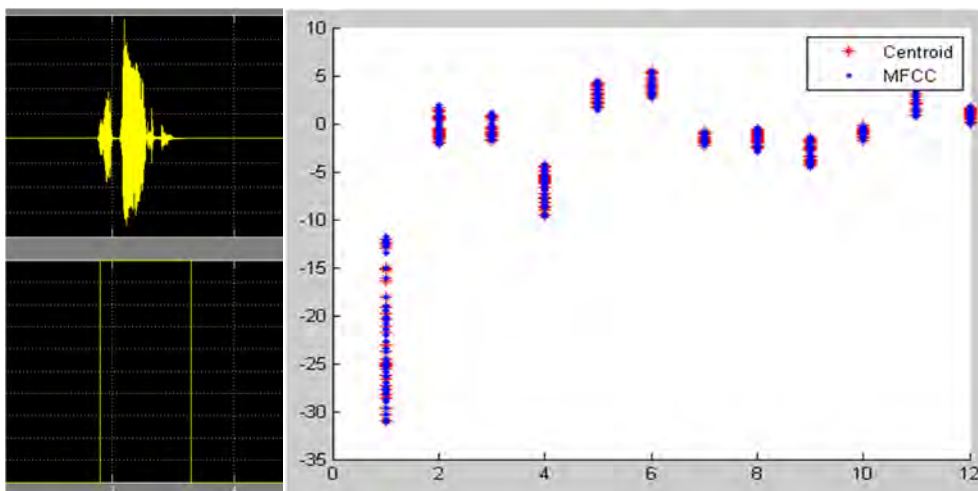


Figure 3.17- Le mot START détecté et compressé

Dans le tableau 3.2 nous présentons le taux de succès de notre système de reconnaissance pour 20 essais en utilisant 4 mots de commande.

Tableau 3.2– Le taux de succès

Commande	True	False	
START	18	2	90%
STOP	12	8	60%
UP	14	6	70%
DOWN	15	5	75%

3.2.4 L'utilisation de la bibliothèque VOCE

Cette bibliothèque exploite les points forts des logiciels existants. Voce fournit une couche mince entre les bibliothèques sous-jacentes d'interaction vocale et les applications.

L'architecture globale de la bibliothèque Voce est conceptuellement simple (figure 3.18). Pour la synthèse vocale, Voce prend des chaînes de textes à partir des applications et les transmet à un synthétiseur TTS qui les convertit en une sortie audio. Pour la reconnaissance vocale, elle alloue un microphone Sphinx4 qui écoute en permanence les données audio entrantes du matériel audio de l'utilisateur. Un reconnaiseur Sphinx4 traite en permanence ces données. La synthèse et la reconnaissance sont exécutées dans des threads séparés du thread principal de l'application pour éviter que l'application attende la fin de la synthèse ou de la reconnaissance.

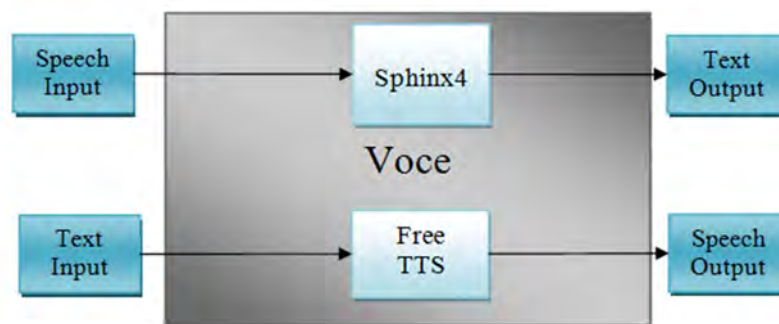


Figure 3.18- L'architecture de la bibliothèque VOCE

3.2.4.1 Fichiers de grammaire

Le fichier de grammaire définit ce qui est identifié au cours de la reconnaissance vocale. Bien que l'ensemble du dictionnaire de mots possibles est de 120.000 mots, la plupart des applications, utilisent uniquement un petit sous-ensemble défini par les fichiers de grammaire. Ces fichiers sont spécifiques à l'application de sorte qu'ils doivent être écrits par le développeur de l'application. Ils

doivent être conformes au Java Speech Grammar File (JSGF) format. Ce qui suit est un exemple d'un JSGF:

```
grammar robot;

public<command> = (forward | stop | <turn>);

<turn> = turn <direction>;

<direction> = (left | right);
```

Dans cet exemple, un ensemble de commandes simples est utilisé pour guider un robot. La règle de "command" grammairale est satisfaite si les mots "forward" ou "stop" sont prononcés ou si la règle "turn" est respectée. La règle "turn" est satisfaite si le mot «turn» suivi de «left» ou «right» est parlé.

3.2.4.2 Post-traitement du texte reconnu

Une fois les données audio sont reconnues et converti en texte, c'est à l'application de décider comment les traiter. Voce ne fait aucune analyse supplémentaire sur le traitement du langage naturel ou sur les chaînes reconnues. L'exemple suivant illustre les façons communes pour traiter les chaînes de caractères reconnues.

Une méthode simple consiste à vérifier si les chaînes reconnues contiennent certains mots-clés. Par exemple, une application peut vérifier si l'utilisateur veut quitter en recherchant toutes les chaînes reconnues pour le mot "quit" (en supposant que "quit" a été spécifié dans la grammaire). Si l'application dispose d'une liste de commandes (par exemple, "left", "right", "up", "down", "slow" et "fast"), il sera nécessaire de rechercher les chaînes reconnues pour l'ensemble de ces mots. L'exemple suivant illustre cette méthode:

```
while (voce::getRecognizerQueueSize() > 0)
{
std::string s = voce::popRecognizedString();
// Check if the string contains 'quit'.
if (std::string::npos != s.rfind("quit"))
{
quitApp = true;
}
// Check for movement direction.
if (std::string::npos != s.rfind("left"))
{
moveDirection = MOVE_LEFT;
}
else if (std::string::npos !=
s.rfind("right"))
{
moveDirection = MOVE_RIGHT;
```

```

}
else if (std::string::npos !=
s.rfind("up"))
{
moveDirection = MOVE_UP;
}
else if (std::string::npos !=
s.rfind("down"))
{
moveDirection = MOVE_DOWN;
}
else if (std::string::npos !=
s.rfind("slow"))
{ moveSpeed = SPEED_SLOW; }
else if (std::string::npos !=
s.rfind("fast"))
{
moveSpeed = SPEED_FAST;
}}

```

3.2.4.3 Références des classes pour la librairie VOCE

Les fonctions publiques et statiques:

Tableau3.3– Références des classes pour la librairie VOCE

staticvoid	<u>init</u> (String vocePath, booleaninitSynthesis, booleaninitRecognition, String grammarPath, String grammarName)
staticvoid	<u>destroy</u> ()
staticvoid	<u>synthesize</u> (String message)
staticvoid	<u>stopSynthesizing</u> ()
staticint	<u>getRecognizerQueueSize</u> ()
static String	<u>popRecognizedString</u> ()
staticvoid	<u>setRecognizerEnabled</u> (boolean e)
staticboolean	<u>isRecognizerEnabled</u> ()

Les attributs statiques privés :

Tableau3.4– Les attributs statiques privés

static <u>SpeechSynthesizer</u>	<u>mSynthesizer</u> = null
static <u>SpeechRecognizer</u>	<u>mRecognizer</u> = null

3.2.4.4 Les problèmes rencontrés

Deux problèmes importants se posent pendant le développement concernant le threading:

- 1) synthèse de la parole prend une quantité importante de temps pour produire sa première sortie audio lors de l'exécution d'une application.
- 2) La fonction de reconnaissance CMU Sphinx4 primaire n'obéit pas à la règle de file d'attente. parce qu'elle doit terminer prioritairement la reconnaissance.

3.3 La commande gestuelle

Dans cette partie, nous présentons l'implémentation de deux extensions la commande gestuelle (la commande avec contact avec l'environnement et sans contact) en utilisant une technique de détection et de poursuite des éléments colorés.

3.3.1 Implémentation de la commande gestuelle

Dans cette section, nous présentons le principe et la mise en œuvre de la commande gestuelle qui est une technique de contrôle de robot de haut niveau qui exploite le **geste graphique**. Le système peut être décomposé en deux sites. Le premier est le site opérateur et le second est le site robot (figure 3.20).

Pour le site robot, nous avons deux configurations possibles, la configuration pour le mode « point and click » et la configuration pour la commande gestuelle (figure 3.19). Dans notre cas, nous avons deux changements à faire pour l'espace du travail du robot (le plan carré) et pour l'effecteur final. Une feuille de papier glacée est placée sur le plan carré du robot à 4 câbles. L'effecteur final est équipé par un stylo à encre effaçable. Cette configuration permettra l'initialisation de l'espace de travail après chaque tâche et la vérification de l'exécution de la trajectoire demandée.

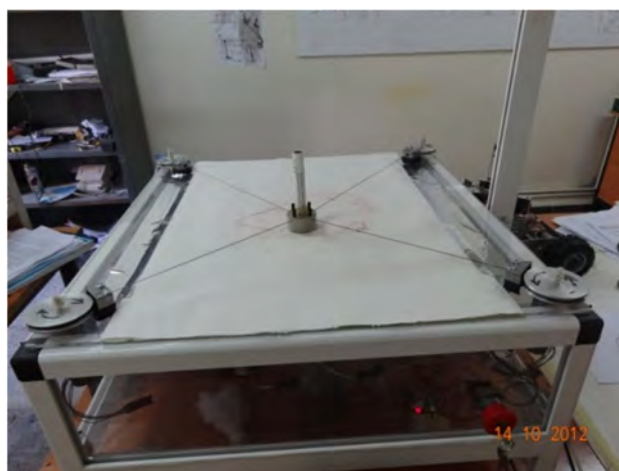


Figure 3.19-La configuration du robot parallèle pour la commande gestuelle

La configuration du system global est comme suit. Dans le site operateur, l'utilisateur interagit avec le prototype expérimental par l'intermédiaire d'une interface graphique, à l'aide des gestes spécifiques sous l'observation d'une camera CCD (figure 3.20). Les mouvements sont détectés et convertis en commandes pour contrôler le robot. Dans le site du robot, une camera CCD est placée pour superviser l'espace de travail et permettre la connaissance de l'état actuel du robot (pour ce mode est utilisée comme élément de feedback du site distant).

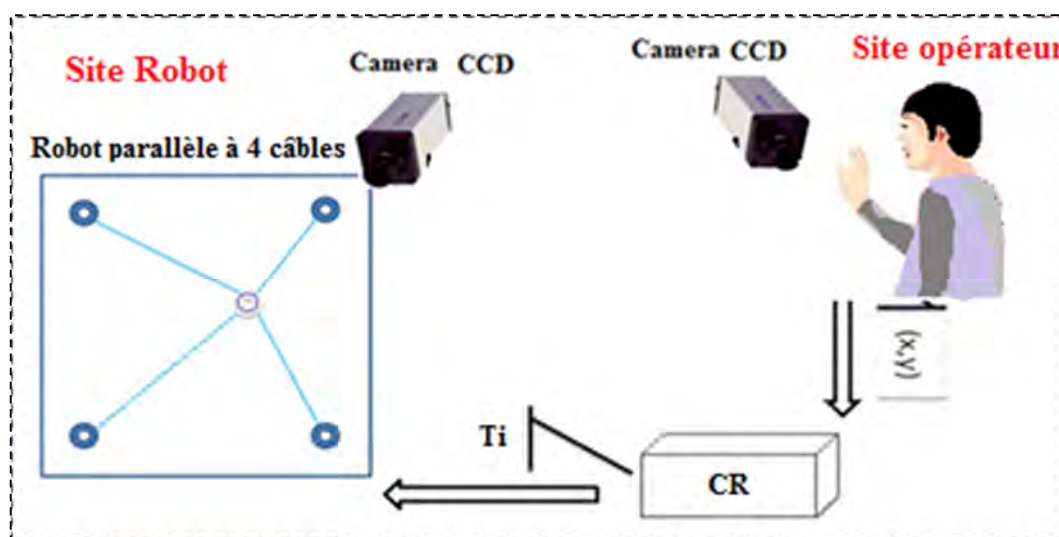


Figure 3.20-Le système de contrôle gestuel pour le robot à 4 câbles

La technique de ce mode de contrôle est mise en œuvre en se basant sur la détection d'un objet coloré à partir de l'image fournie par la camera CCD du site opérateur. L'algorithme de détection se compose de deux parties ; la première partie concerne le processus d'initialisation qui commence par sélectionner l'objet coloré à suivre. Cela se fait au moyen d'un simple clic de souris sur les éléments tel qu'un doigt coloré de la main ou un stylo coloré.

La seconde partie concerne le processus de poursuite qui est basé sur l'analyse des trames d'images. Chaque trame est complètement analysée en fonction de l'algorithme présenté à la figure 3.21, afin de détecter l'élément de couleur et déterminer la direction du déplacement.

Les éléments qui apparaissent dans la figure 3.21 sont définis comme suit:

TC: la couleur en surveillance (Tracked color)

CP: la Couleur du Pixel

i: la ligne de la matrice de pixels

j: la colonne de la matrice de pixels

Dist: la fonction de la distance euclidienne

Nous avons utilisé la fonction de la distance euclidienne pour comparer la couleur de l'objet à la couleur du pixel. Cette distance est comparée à une valeur de seuil.

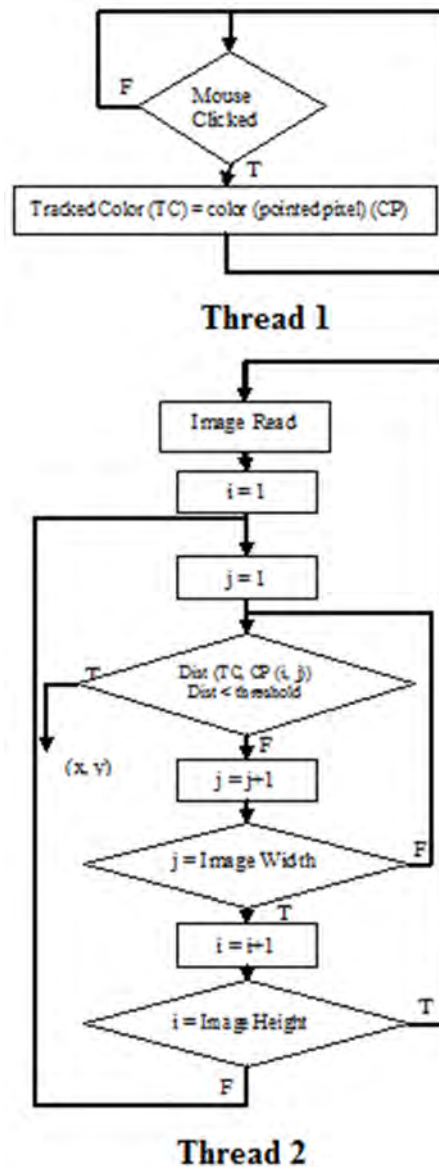


Figure 3.21- L’algorithme de détection et poursuite des objets colorés

Les étapes permettant d’exécuter une tâche en utilisant ce mode s’effectue comme suit. L’utilisateur dans le site opérateur sélectionne le stylo coloré placé sur l’espace de travail opérateur comme objet à poursuivre. La localisation de l’objet à poursuivre s’effectue par l’opérateur au moyen d’un clic de la souris sur l’image reçue à travers la camera CCD du site (figure 3.22). Ce clic sur l’emplacement image du stylo engage le système de vision à pointer virtuellement l’objet à poursuivre sur l’image reçue. Le verrouillage sur l’objet en question est assuré par le tracé d’un cercle virtuel de même couleur que l’objet (figure 3.22). Ce point est transmis comme commande de position à l’effecteur final du robot. A partir de cet instant, ce premier point constitue l’emplacement initial de notre trajectoire. Par la suite, chaque déplacement du stylo dans l’espace opérateur est considérée commettent une commande à exécuter en temps réel par le robot.

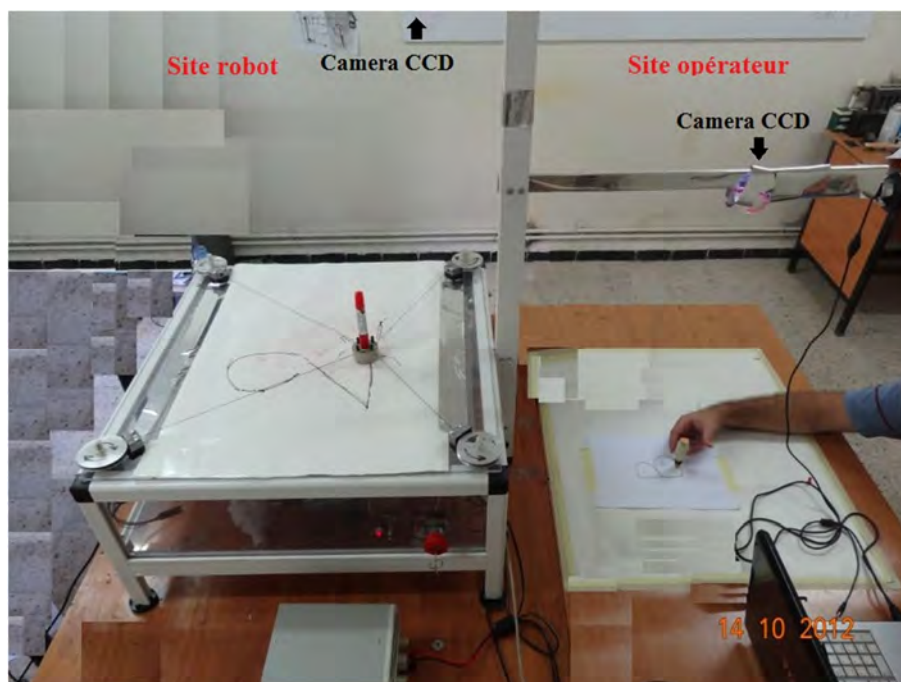


Figure 3.22- Le système interactif en mode gestuel

Dans ce qui suit, nous allons présenter quelques expériences d'illustration de ce mode de contrôle haut niveau. Ces expériences nous ont permis de diviser ce mode de commande en deux sous modes, le geste graphique avec contact avec l'environnement et sans contact avec environnement.

La figure 3.23 présente une première expérience en utilisant le mode gestuel avec contact. Dans ce cas, l'objet sélectionné est un stylo coloré en orange qui est en contact direct avec le plan de l'espace de travail. Après le clic sur le stylo, un cercle en orange est affiché sur le sommet du stylo (figure 3.23.a) pour indiquer que **cet objet** est en état de poursuite par le system de vision. La figure 3.23.b présente la trajectoire demandée (consigne) en bleu qui est un cercle et la trajectoire réalisée par l'effecteur final en rouge. Les deux trajectoires sont superposées sur une zone graphique pour pouvoir les comparer. La figure 3.23.c présente le cercle réalisé par le robot.

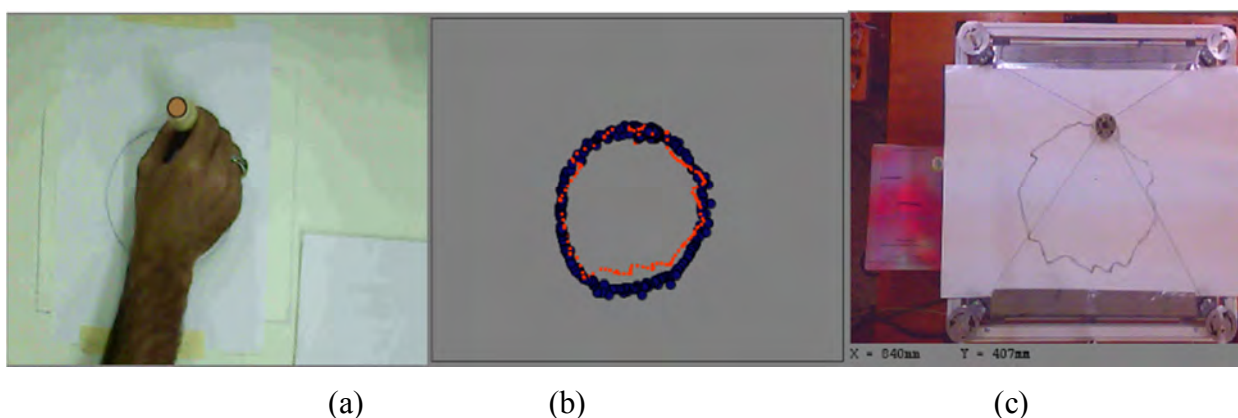


Figure 3.23-L'acquisition d'un cercle en mode gestuel avec contacte avec l'environnement

Nous pouvons remarquer que la trajectoire réalisée par l'effecteur final du robot à 4 câbles, est la même que celle récupérée par le système d'acquisition et qui est affichée sur l'interface graphique.

La seconde expérience est la réalisation de la lettre « B » en utilisant le mode de contrôle gestuel avec contact (figure 3.24). La qualité de la trajectoire en termes de précision sera discutée dans le chapitre de l'analyse de l'interaction.

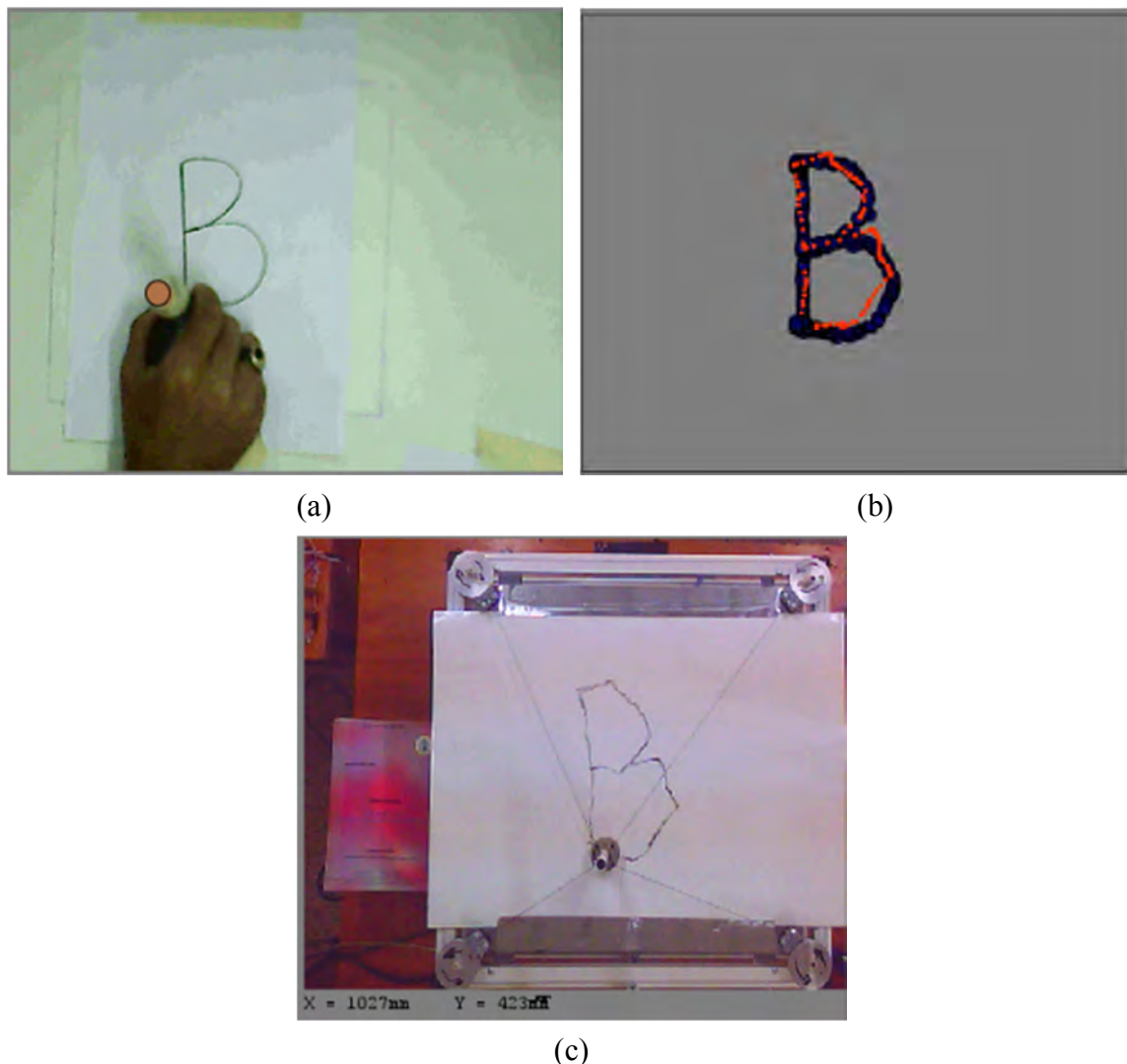


Figure 3.24-L'acquisition de la lettre « B » en mode gestuel avec contacte avec l'environnement

Le deuxième exemple d'application concernant la commande basée sur le geste graphique est présenté par la figure 3.25 en utilisant le mouvement gestuel de la main de l'opérateur sans contact avec l'environnement. Cette technique peut être utilisée pour contrôler notre robot en suivant le mouvement de la main, mais aussi en traduisant cette action en des tâches préprogrammées. Par exemple, un geste spécifique de la main peut être préprogrammé pour engager le robot à initialiser l'espace de travail. La figure 3.25.a, montre un mouvement du doigt de l'opérateur sans contact. La figure 3.25.b montre la superposition de la trajectoire du mouvement de la main avec celui de l'organe terminal. Nous remarquons la différence entre les deux trajectoires qui est causée par de

nombreux éléments tels que le manque de coordination entre les vitesses du doigt par rapport à la réponse du robot; l'alignement de l'objet à traquer avec la caméra (doigt mobile dans l'espace 3D).

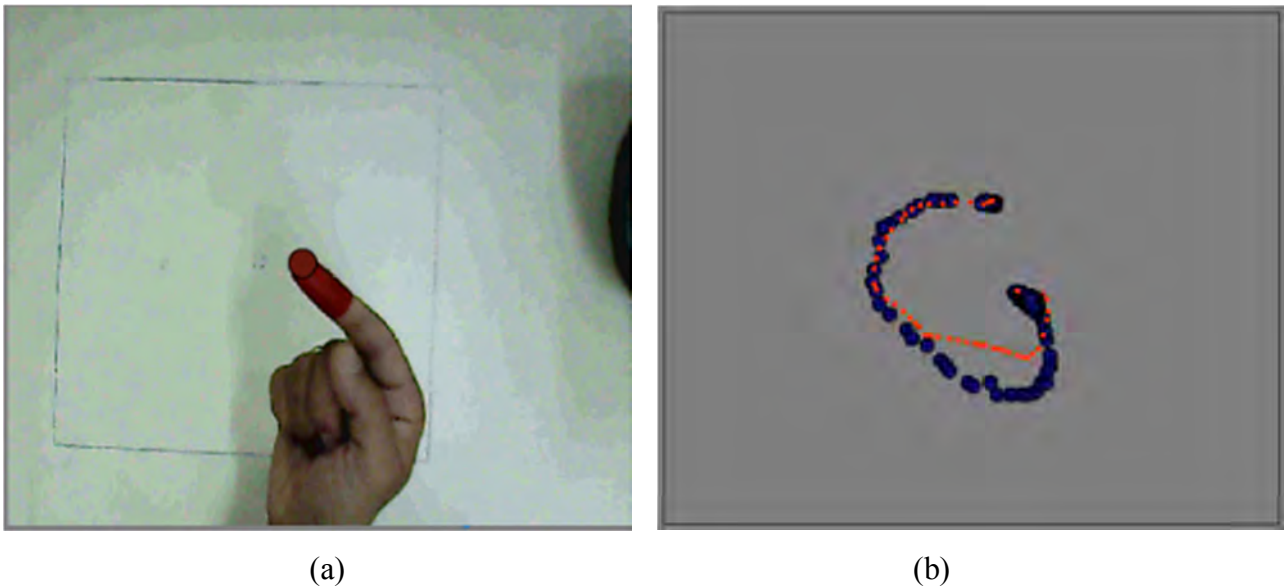


Figure 3.25- L'acquisition d'une commande en mode gestuel sans contact avec l'environnement

3.4 La commande par pointage sur image

Dans cette section, nous présentons le développement et l'implémentation du mode de pointage sur image appliqué à un robot parallèle à 4 câbles.

3.4.1 Implémentation de la commande de pointage sur image

Nous présentons dans cette section, le développement et l'implémentation de la commande de haut niveau qui est le pointage sur image appliqué à notre robot à 4 câbles. Dans ce mode de commande, l'opérateur dirige l'effecteur final vers des endroits dans l'espace 2D au moyen d'un clic d'une souris. L'espace de travail est divisé en deux sites (figure 3.26), le site robot où un robot planaire à 4 câbles est placé sous l'observation d'une camera CCD, et le site operateur où se situe l'operateur muni d'un écran et d'un pointeur (souris). Ce dernier par de simples clics peut sélectionner des objets de l'espace de travail qui s'affichent sur son écran. Ces points sont envoyés en tant que commandes au robot pour l'exécution dans l'espace réel.

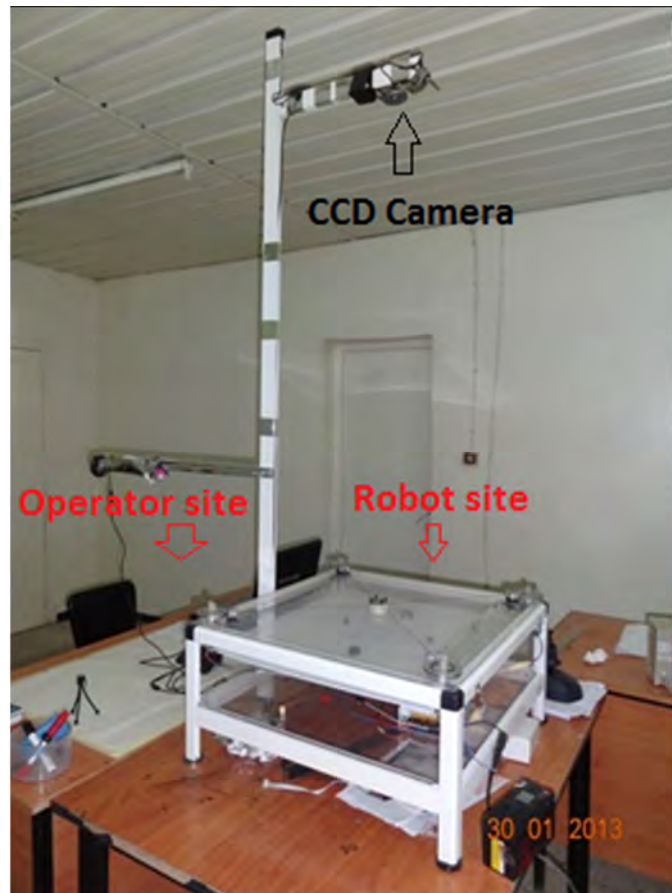


Figure 3.26- L'espace expérimental du robot à 4 câbles

Lorsque l'opérateur clique sur un point choisi dans l'image, le système détermine les coordonnées image (u, v) , et les convertit en leur emplacement correspondant dans l'espace réel en 2D (x, y) (figure 3.27). Par la suite, les coordonnées (x, y) sont utilisés comme entrées de consigne pour le contrôleur bas-niveau (figure 3.27). Cette conversion de coordonnées est effectuée au moyen de la matrice de projection de perspective P de la camera CCD utilisée (figure 3.28).

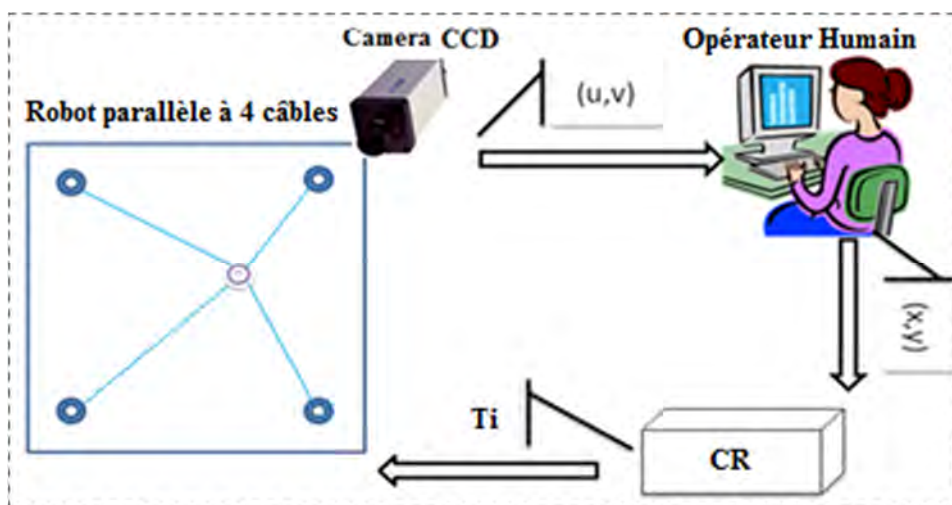


Figure 3.27-Le système de pointage sur image pour le robot parallèle à câbles

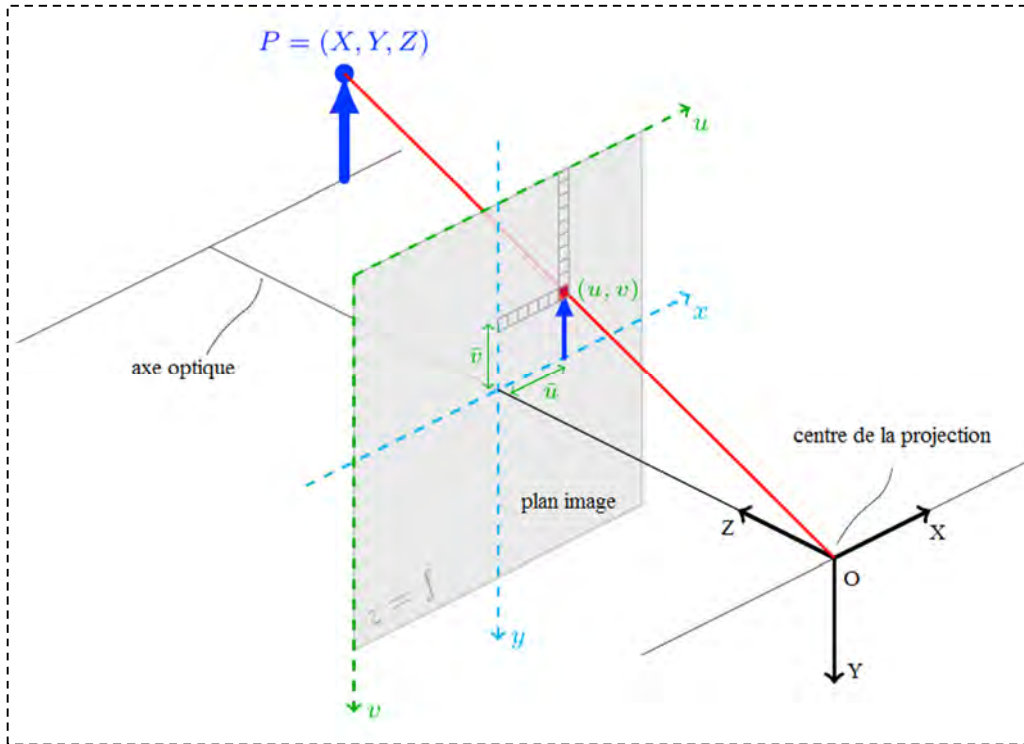


Figure 3.28- Projection perspective

Une caméra est caractérisée par sa matrice de projection perspective P (figure 3.28). Cette matrice détermine la projection d'un point de l'espace 2D réel en un point de l'image selon les transformations suivantes [88]:

$$\begin{cases} x = x_0 + h \frac{f_u \cos(\alpha) + (u_i - u_0) \sin(\alpha)}{(u_i - u_0) \cos(\alpha) - f_u \sin(\alpha)} \\ y = y_0 + h \frac{f_u (v - v_0)}{f_v ((u - u_0) \cos(\alpha) - f_u \sin(\alpha))} \\ z = h \end{cases} \quad (3.17)$$

Où les éléments des deux expressions sont définis comme suit :

(f_u, f_v, u_0, v_0) Les paramètres intrinsèques de la camera estimés dans la phase de calibration.

u_0, v_0 : Les coordonnées des axes dans le plan d'image.

f_u, f_v : Sont données par la relation

$$f_u = \frac{n_x}{l_x} \text{foc} \quad \text{Et} \quad f_v = \frac{n_y}{l_y} \text{foc}$$

n_x, n_y : Le nombre de pixels dans la direction dans le plan image.

l_x, l_y : La longueur et la largeur d'un pixel.

h : La hauteur de la camera par rapport au terrain.

α : L'angle entre l'axe horizontal et l'axe central de la camera.

Le modèle inverse de la camera qui donne les coordonnées u, v dans une image a partir des coordonnées de terrain 2D est:

$$\begin{cases} u = u_0 + \frac{f_u \cos(\alpha) + (x-x_0) \sin(\alpha)}{(x-x_0) \cos(\alpha) - h \sin(\alpha)} \\ v = v_0 + f_v \frac{y-y_0}{(x-x_0) \cos(\alpha) - h \sin(\alpha)} \end{cases} \quad (3.18)$$

3.4.1.1 La procédure de calibration de la camera

L'opération de calibration d'une caméra revient à modéliser le processus de formation des images, c'est-à-dire de trouver la relation entre les coordonnées spatiales d'un point de l'espace avec le point associé dans l'image correspondante. La modélisation de l'optique de la camera qui est basée sur des considérations de la lentille et de sa distorsion revient à calculer les paramètres intrinsèques. La détermination de la transformation entre le repère absolu et le repère de la camera revient à calculer les paramètres extrinsèques.

Le calcul de la matrice de transformation P peut s'effectuer en utilisant un objet plan dont la géométrie est connue. Les paramètres sont calculés à partir de l'observation de plusieurs vues de l'objet sans besoin de connaître leur position [89]. Pour notre application, le plan du modèle utilisé pour la procédure de calibration contient un motif constitué de carrés 3x3 ou grille de calibration, la taille de chaque motif est de 70 mm x 70 mm (figure 3.29). Le motif est placé dans l'espace de travail du robot à 4 câbles comme présenté dans la figure 3.30.

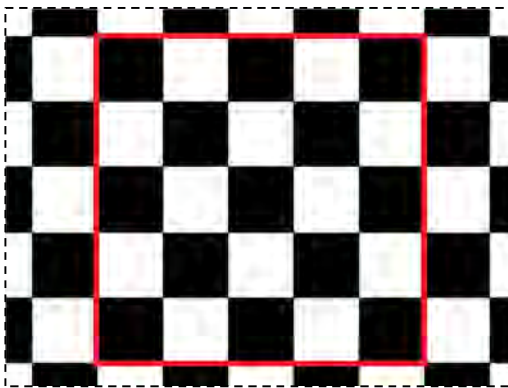


Figure 3.29- Grille de calibration



Figure 3.30-Montage de calibration

Nous avons effectué l'acquisition de 13 images de la grille de calibration pour différentes positions et orientations (figure 3.31).



Figure 3.31- Différentes positions et orientations de la grille sur l'espace de travail

Nous avons choisi d'utiliser le Toolbox « Camera Calibration Toolbox » « CCT » qui est implémenté sous Matlab. Cet outil utilise la méthode de Zhang pour le calibrage. Nos recherches avaient préalablement indiqué que cette dernière, ainsi que la méthode de Tsai [90], étaient les plus utilisées et qu'elles donnaient d'excellents résultats. La figure 3.32 présente la projection des points de l'image sur les points de la grille.

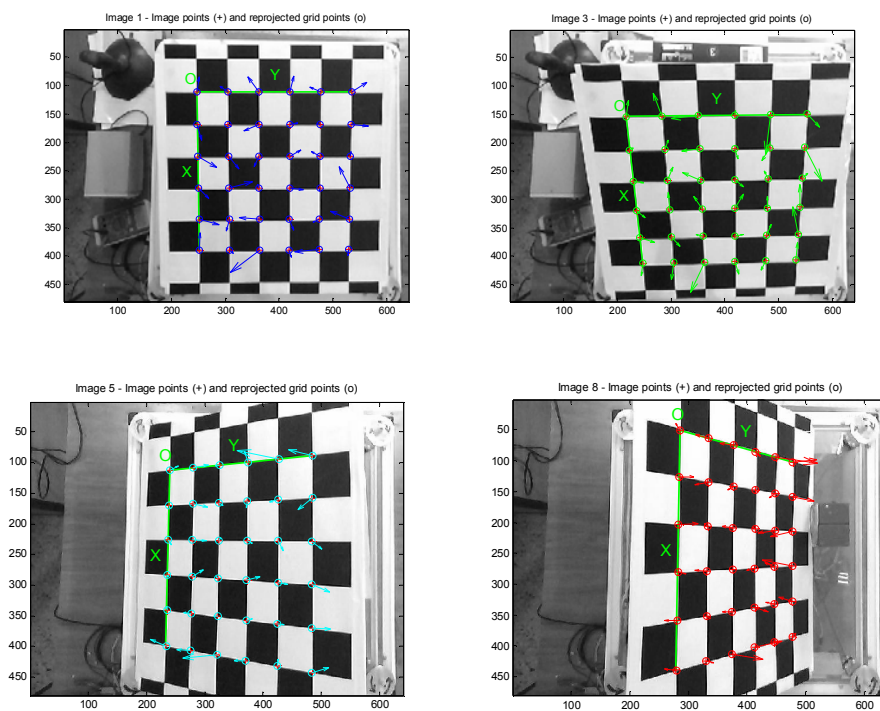


Figure 3.32- La projection des points de l'image sur les points de la grille

Les paramètres intrinsèques sont calculés selon le modèle de Zhang et sont :

La distance focale: $f_c = [859.18830 \ 860.98151] \pm [6.22887 \ 6.34436]$

Le point principal: $cc = [411.96062 \ 171.57415] \pm [8.11019 \ 9.07356]$

Skew: $\alpha_c = [0.00000] \pm [0.00000] \Rightarrow \text{angle of pixel axes} = 90.00000 \pm 0.00000 \text{ degrees}$

Distorsion: $k_c = [-0.05870 \ 0.63338 \ -0.00798 \ 0.00418 \ 0.00000] \pm [0.03896 \ 0.19827 \ 0.00435 \ 0.00371 \ 0.00000]$

Pixel error: $err = [0.39969 \ 0.33098]$

Les paramètres extrinsèques dépendent de chaque prise de vue. La figure 3.33 reconstitue chaque prise de vue selon une numérotation spécifique donnée par le Toolbox CCT de Matlab.

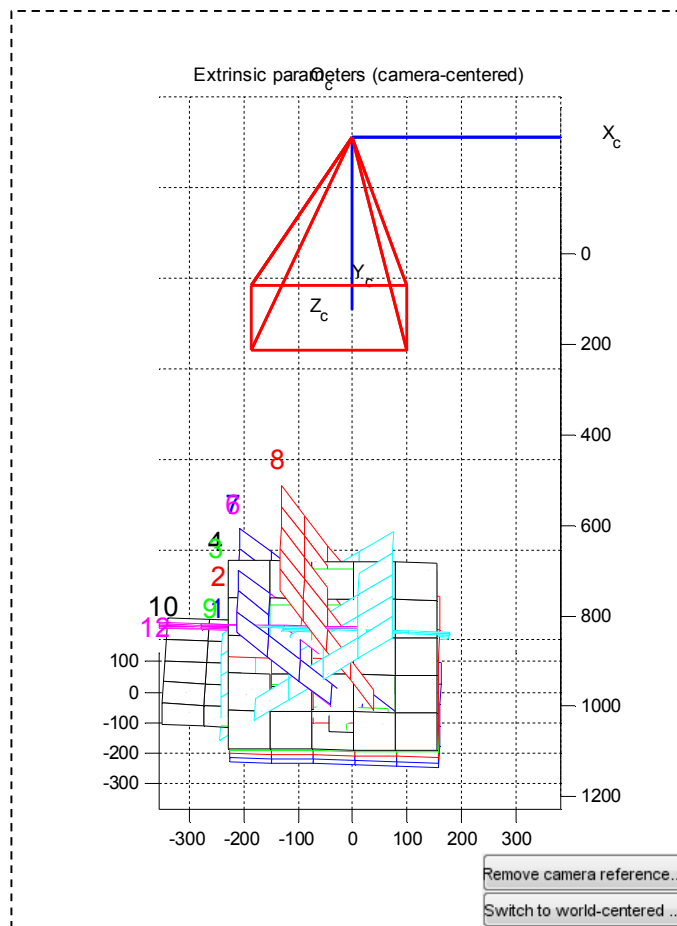


Figure 3.33-La reconstruction des prises de vue en utilisant le Toolbox CCT

3.4.1.2 Exemples d'applications

Nous illustrons ci-après deux essais expérimentaux en utilisant la commande par pointage sur image à l'aide d'une interface graphique. Dans le premier essai, cette interface est constituée de deux parties (figure 3.34): la partie à gauche montre l'image transmise par la camera de l'espace de travail incluant le robot à câbles. La partie à droite montre une zone graphique pour l'affichage des deux

points (désirée et actuelle). Les cibles visées sont des objets quelconques placés arbitrairement dans l'espace de travail (figure 3.34). Le principe d'utilisation est le suivant : l'opérateur désigne à partir de l'interface graphique la cible à l'aide d'un clic de souris sur son image, l'effecteur final se déplace jusqu'à ce qu'il se superpose sur la cible. La figure 3.34 représente le robot dans une configuration initiale au centre de l'espace de travail. Dans l'espace de travail apparaît un objet considéré comme cible. L'opérateur désigne cette cible par un simple clic sur l'image (figure 3.35), le robot à câble exécute cette commande et transfère l'effecteur final vers ce point (figure 3.36).

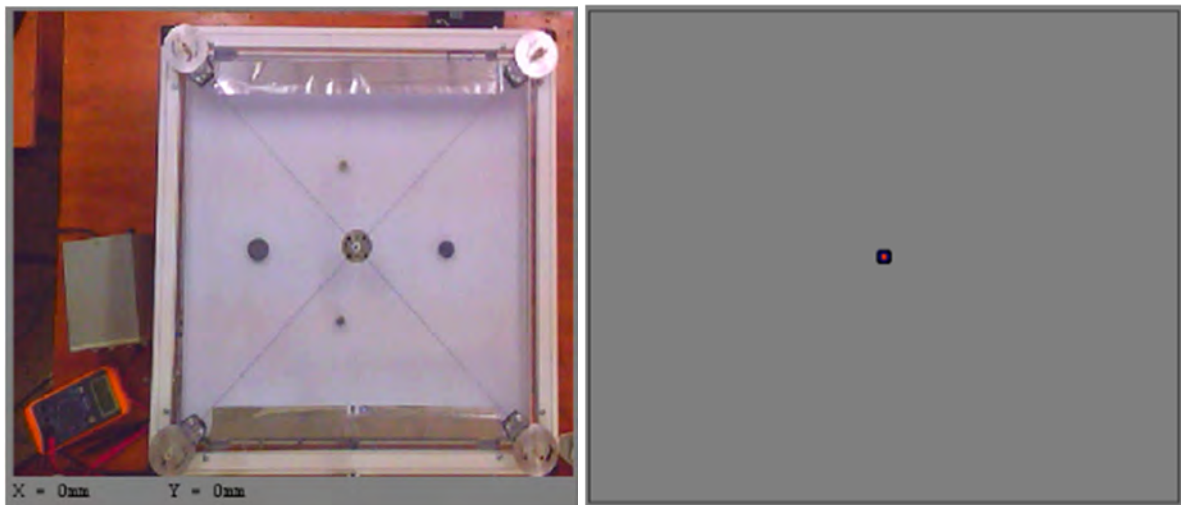


Figure 3.34- L'état initiale du robot parallèle en mode pointage par image

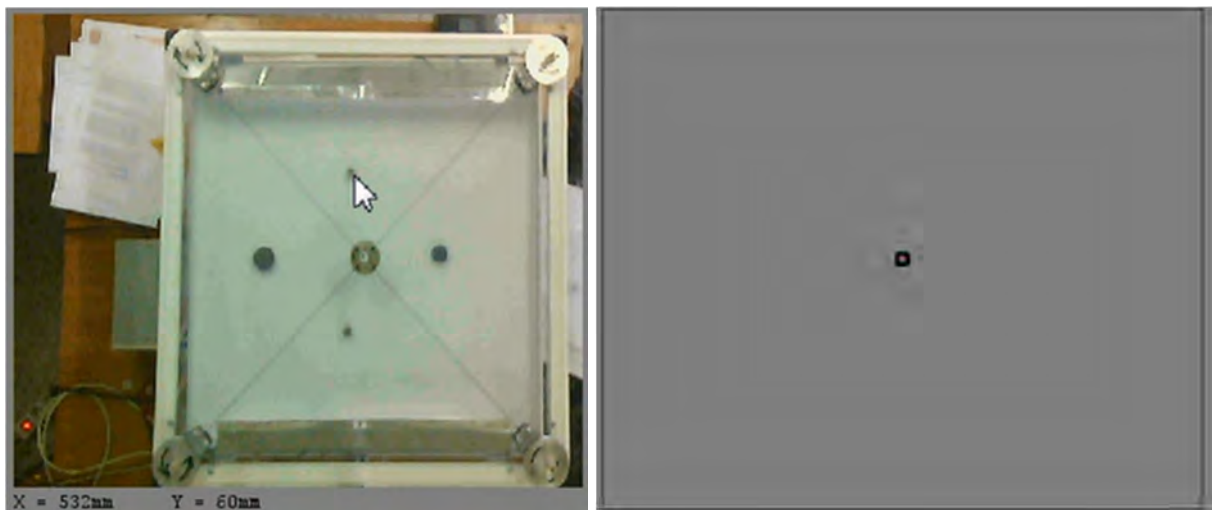


Figure 3.35-La sélection d'un objet par le mode de pointage par image

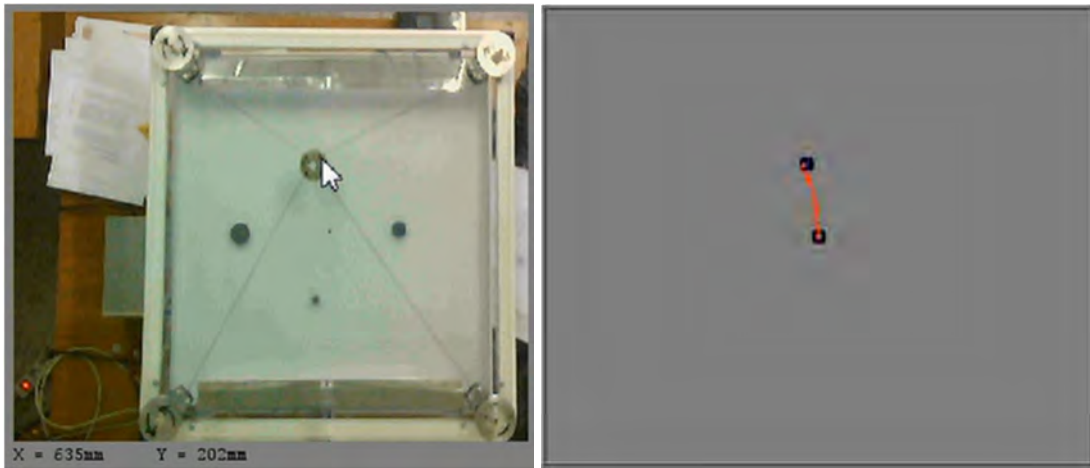


Figure 3.36-La superposition de l'effecteur final sur l'objet sélectionné

Dans le deuxième essai, nous présentons un test pour le pointage de 3 objets successivement. Pour ce cas l'interface est constituée de deux parties : la partie supérieure montre l'image transmise par la camera de l'espace de travail et de robot à câbles, la partie inférieure montre le simulateur du robot à câbles. Si l'opérateur sectionne un objet sur l'image, le simulateur exécute cette commande et transfère l'effecteur final vers ce point, en dernier lieu cette commande est exécutée par le robot à câble dans l'espace réel (figure 3.37).

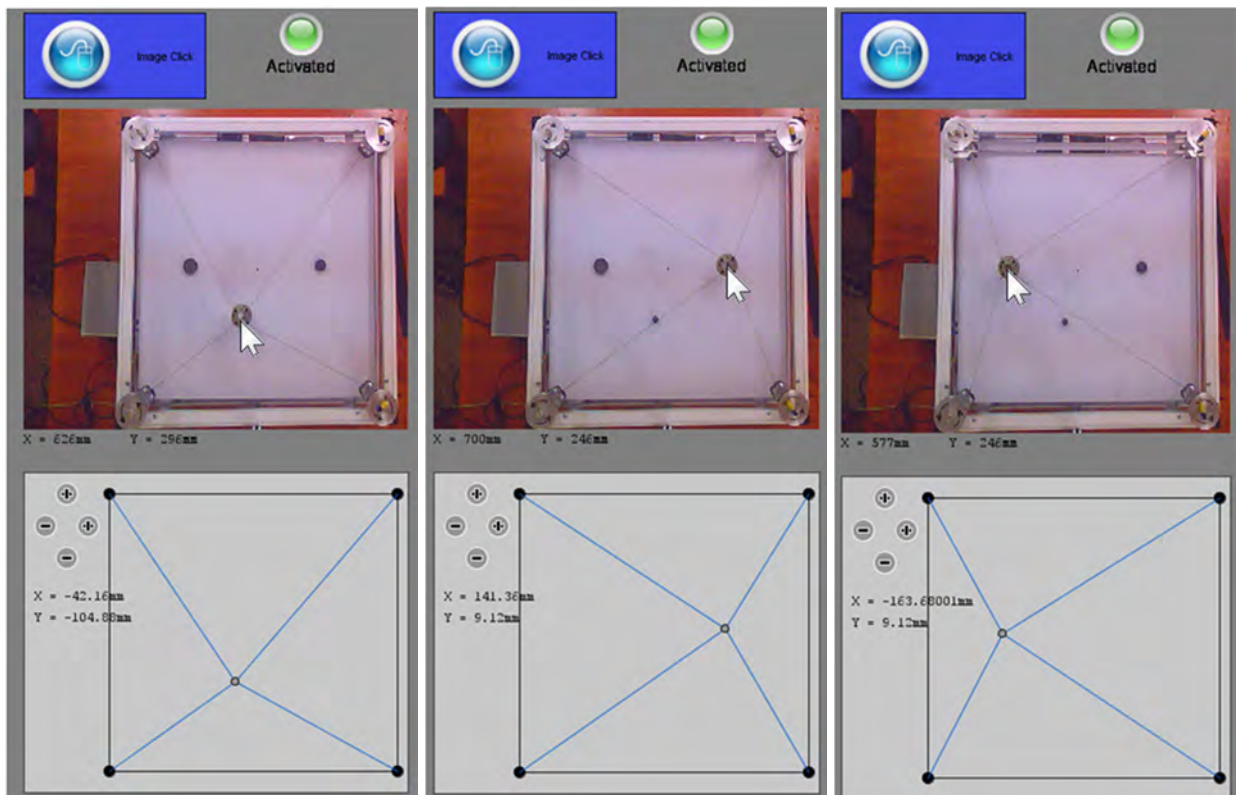


Figure 3.37-Une sélection successive de 3 objets avec pointage par image

3.5 La commande joystick

Pour illustrer la commande par joystick, nous utilisons la manette de jeu montrée à la figure 3.38. Ce type de manette dispose de deux modes de fonctionnement: le mode numérique et le mode analogique. Dans le cas du mode numérique, les boutons d'actions ont trois positions (1, 0, -1); contrairement en mode analogique où les commandes peuvent prendre plusieurs valeurs allons de 1 jusqu'a -1 selon la résolution.



Figure 3.38-Manette de jeu utilisée comme joystick

Nous avons exploité ces deux modes de fonctionnement dans notre interface multimodale pour enrichir les possibilités de contrôle. Le mode numérique a été adapté pour commander le déplacement de l'effecteur final pas à pas (figure 3.39). Le mode analogique a été adapté pour une commande continue (figure 3.40).

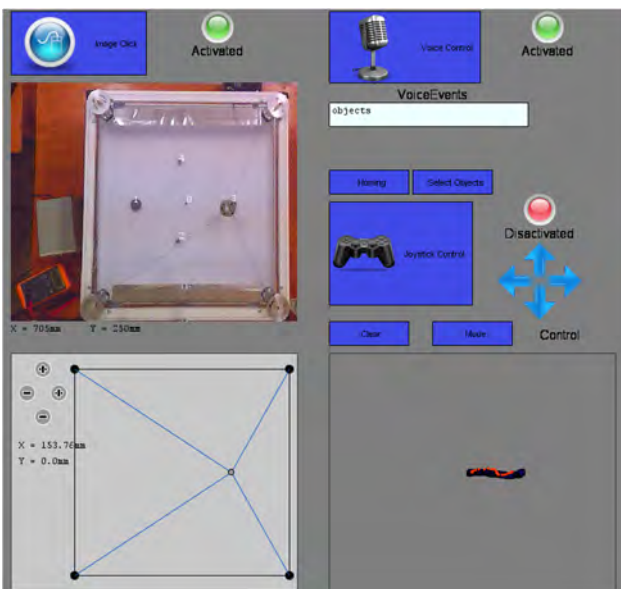


Figure 3.39- Le mode continue

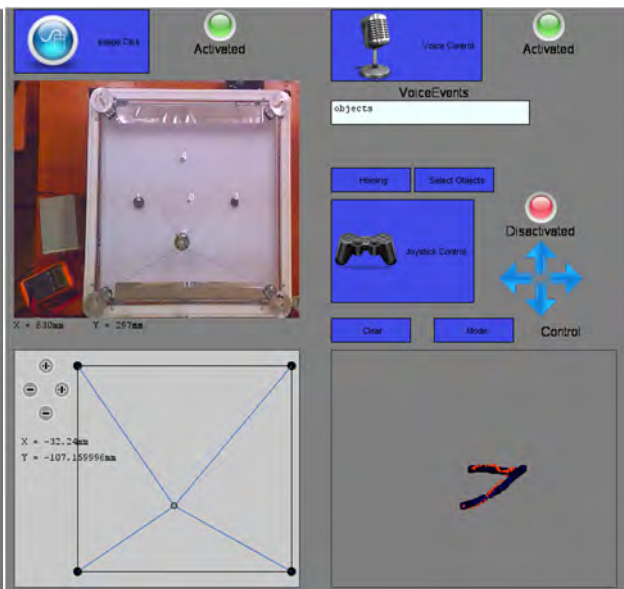


Figure 3.40- Le mode pas à pas

3.6 Conclusion

Ce chapitre a été consacré à l'implémentation de quatre mode de commande haut niveau pour l'interaction homme-robot. Nous avons commencé par l'implémentation de la commande vocale en utilisant deux méthodes. La première méthode consiste dans l'implémentation de la technique MFCC sous l'environnement Matlab. Cette technique est basée sur la détermination des coefficients cepstraux selon une échelle modélisant l'oreille humaine appelée échelle de Mel. La seconde méthode, consiste en l'utilisation de la bibliothèque VOCE sous l'environnement Processing en exploitant ces deux fonctionnalités, la reconnaissance et la synthèse de la parole.

Ensuite, nous avons présenté l'implémentation de la commande gestuelle en utilisant une technique de poursuite des éléments colorés. Nous avons distingué deux extensions de ce mode: le mode gestuel avec contact avec l'environnement et le mode sans contact avec l'environnement. Par la suite, nous avons présenté l'implémentation de la commande de pointage sur image. Nous avons également exposé la procédure de calibration pour extraire les paramètres intrinsèques et extrinsèques de la caméra, ainsi que l'implémentation de l'algorithme de conversion entre le plan image et le plan réel. A la fin de cette partie, nous avons présenté l'implémentation de la commande joystick.

Nous y avons également présenté des résultats de tests préliminaires permettant de vérifier l'exploitation effective pour chaque mode.

Chapitre 4

La Conception de l'Interface Multimodale Homme-Robot

Sommaire

4.1	Introduction	116
4.2	Le système interactif global	116
4.3	La conception centrée utilisateur appliqué au SMM	117
4.3.1	Quatre principes fondamentaux.....	118
4.3.2	Etapes du processus de conception centrée utilisateur.....	119
4.4	L'implémentation de l'interface graphique multimodale	119
4.4.1	L'environnement de développement Processing.....	120
4.4.2	L'implémentation des modes de commande dans l'interface graphique multimodale.....	121
4.4.2.1	L'implémentation de la commande vocale.....	121
4.4.2.2	L'implémentation de la commande joystick.....	122
4.4.2.3	L'implémentation de la commande pointage sur image.....	124
4.4.2.4	L'implémentation de la commande gestuelle.....	125
4.4.3	Les routines supplémentaires implémentées.....	126
4.4.3.1	L'initialisation automatique du robot.....	126
4.4.3.2	Le mode d'acquisition.....	128
4.4.3.3	Les zones graphiques.....	129
4.4.3.4	Le simulateur virtuel.....	131
4.4.4	L'interface multimodale globale.....	132
4.4.5	Développement de l'interface graphique pour le robot manipulateur sous Labview.....	133
4.5	Conclusion	137

4.1 Introduction

Dans le domaine de l'Interaction Homme-Machine « IHM » il y a un ensemble des phénomènes physiques et cognitifs qui interviennent dans la réalisation des tâches. Cette interaction est généralement effectuée à travers une Interface Homme-Machine qui désigne un assemblage de composants logiciels et matériels. Dans ce chapitre nous allons concentrer notre attention sur la conception et l'implémentation de l'interface graphique multimodale en se basant sur des méthodes de conception connues notamment la méthode de conception centré utilisateur "CCU".

Le but principal est de concevoir une interface graphique multimodale intuitive, flexible et généralisable pour superviser divers systèmes robotiques. La méthode CCU nécessite une analyse expérimentale et une évaluation cyclique du développement de cette interface [91]. Pour cette raison, nous avons réalisé une version initiale d'une interface multimodale expérimentale construite autour des modes développés individuellement et présentés au chapitre 3. Cette interface a été conçue pour nous permettre d'analyser les différents modes de commande séparément, combinés ou enchainés. Notre choix de l'environnement de développement de cette interface est motivé principalement par le développement des interfaces graphiques offrant un maximum de flexibilité, tout en assurant la capacité de gérer les multiples entrées/sorties hétérogènes telles que le son, la vidéo, le joystick et les ports d'interfaçages.

4.2 Le système interactif global

Notre système interactif est composé de trois parties (figure 4.1), le site opérateur, le site robot et l'interface graphique d'utilisateur multimodale. Au niveau du site opérateur des éléments matériels sont mis en œuvre pour la capture des entrées provenant de l'utilisateur comme le geste et la voix et autres informations. Ainsi, des éléments logiciels sont mis en œuvre dans l'interface multimodale qui travaille en coordination avec des éléments matériels placés dans le site robot pour assurer le contrôle supervisé à l'aide d'autres commandes de haut niveau comme le pointage sur image.

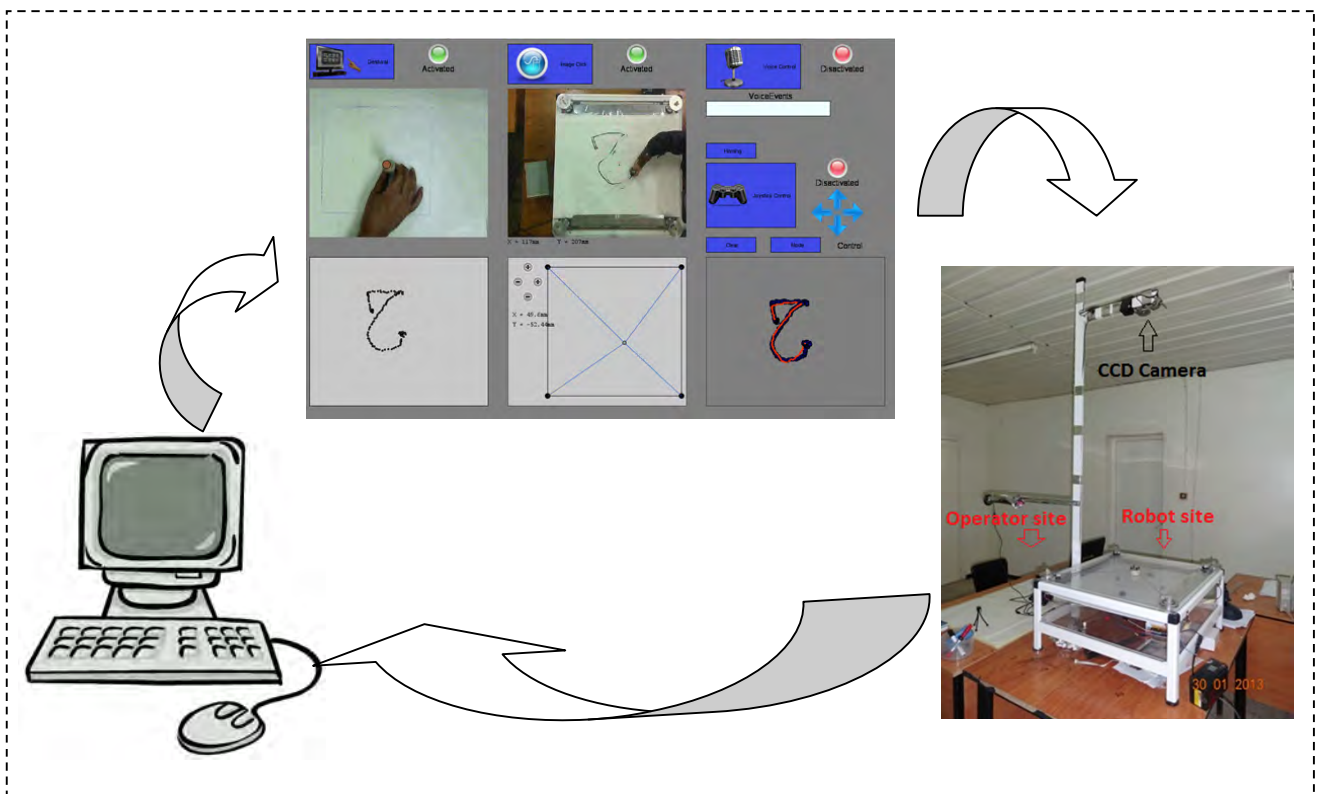


Figure 4.1- Le système interactif multimodal global

Dans le cas de notre application, quatre modes de commande de haut niveau sont implémentés (la commande vocale, la commande gestuelle, la commande par pointage sur image et la commande joystick). Dans ce qui suit, nous allons présenter la méthode sélectionnée pour la conception et l'implémentation du système interactif ainsi que l'interface graphique multimodale d'utilisateur, en se basant sur les principes de conception centré sur l'utilisateur.

4.3 La conception centrée utilisateur appliquée au SMM

Les interactions homme-machine (IHM) définissent les moyens et outils mis en œuvre afin qu'un humain puisse contrôler et communiquer avec une machine. Les ingénieurs dans ce domaine étudient la façon dont les humains interagissent avec les ordinateurs ou entre eux à l'aide d'ordinateurs, ainsi que la façon de concevoir des systèmes qui soient ergonomiques, efficaces, faciles à utiliser ou plus généralement adaptés à leur contexte d'utilisation. La conception centrée sur l'opérateur humain est basée sur le principe que les utilisateurs finaux sont les mieux placés pour évaluer et influencer le développement d'un produit. Elle est décrite dans la norme internationale ISO 9241-210 (2010) (anciennement ISO 13407) comme « *une manière de concevoir les systèmes interactifs, ayant pour objet spécifique de rendre les systèmes utilisables* » [92].

4.3.1 Quatre principes fondamentaux

Une conception centrée utilisateur est basée sur 4 principes qui sont nécessaires à la satisfaction de cette norme :

- Une préoccupation amont des utilisateurs, de leurs tâches et de leur environnement :
Dans notre cas nous avons deux types de plateformes d'études en termes de groupe (utilisateurs, tâches et environnement). La première plateforme ou application concerne les centres de réhabilitation pour les personnes ayant des besoins spéciaux. La deuxième est destinée pour une étude plus générale dans le but de généraliser la conception de l'interface graphique multimodale, qui tient compte d'un groupe plus large en termes d'utilisateurs, tâches et environnement.
- La participation active de ces utilisateurs, ainsi que la compréhension claire de leurs besoins et des exigences liées à leurs tâches :
Nous avons travaillé avec des utilisateurs finaux réels et des utilisateurs finaux potentiels (des handicapés avec différents niveaux d'handicap et des utilisateurs pour usage professionnel ou général). Ces personnes ont participé réellement à la conception du système interactif et l'interface multimodale.
- Une répartition appropriée des fonctions entre les utilisateurs et la technologie :
Ce principe a été pris dans son sens plein par l'objectif d'utiliser plusieurs modes de commande haut niveau, commençons par une version standard de l'interface multimodale et finissons par un système interactif adaptable tout en travaillant dans un niveau d'abstraction élevé. Ce niveau d'abstraction tient compte de la répartition des fonctions et même de la dépendance par rapport à l'utilisateur, à la technologie utilisée et à l'environnement.
- L'intervention d'une équipe de conception multidisciplinaire. La CCU représente en effet plus que de simples considérations sur l'utilisabilité. Cette notion d'expérience utilisateur est au carrefour de disciplines différentes (facteurs humains, architecture de l'information, design, ..., etc) : Dans notre environnement universitaire nous avons fait recours à l'intervention de plusieurs spécialistes de plusieurs domaines comme l'électronique, la mécanique, l'informatique, la psychologie, l'industrie...etc. Nous avons essayé de couvrir les différents domaines de l'analyse de l'interaction homme-robot pour analyser le système interactif et améliorer la conception au fur et à mesure. Il y a lieu de noter que la CCU n'est pas encore appliquée pour les systèmes multimodaux.

Le concept d'utilisateur final réfère ici à deux types de référents :

- L'utilisateur final réel, c'est à dire qui utilisera l'application de façon personnelle ou professionnelle après son lancement (et éventuellement qui utilise déjà une version précédente du produit).
- L'utilisateur final potentiel, qui présente les mêmes caractéristiques que celles de la cible prévue. On fait donc intervenir des participants représentatifs d'un type spécifique de cible (en termes d'âge, de

culture, d'expérience avec l'outil informatique, d'expertise dans un domaine de connaissance donné, d'environnement technologique, etc.).

4.3.2 Etapes du processus de conception centrée utilisateur

Parallèlement aux 4 principes décrits précédemment, ils existent quatre activités interdépendantes de l'CCU (figure 4.2):

- Comprendre et spécifier le contexte d'utilisation.
- Spécifier les exigences liées à l'utilisateur et à l'organisation.
- Proposer des solutions de conception.
- Evaluer les conceptions par rapport aux exigences.

Les deux premières activités constituent la phase d'analyse, suivie de la phase de conception et de la phase d'évaluation. Ces activités forment une boucle itérative dans le but d'améliorer le processus d'une façon continue jusqu'à atteindre les objectifs.

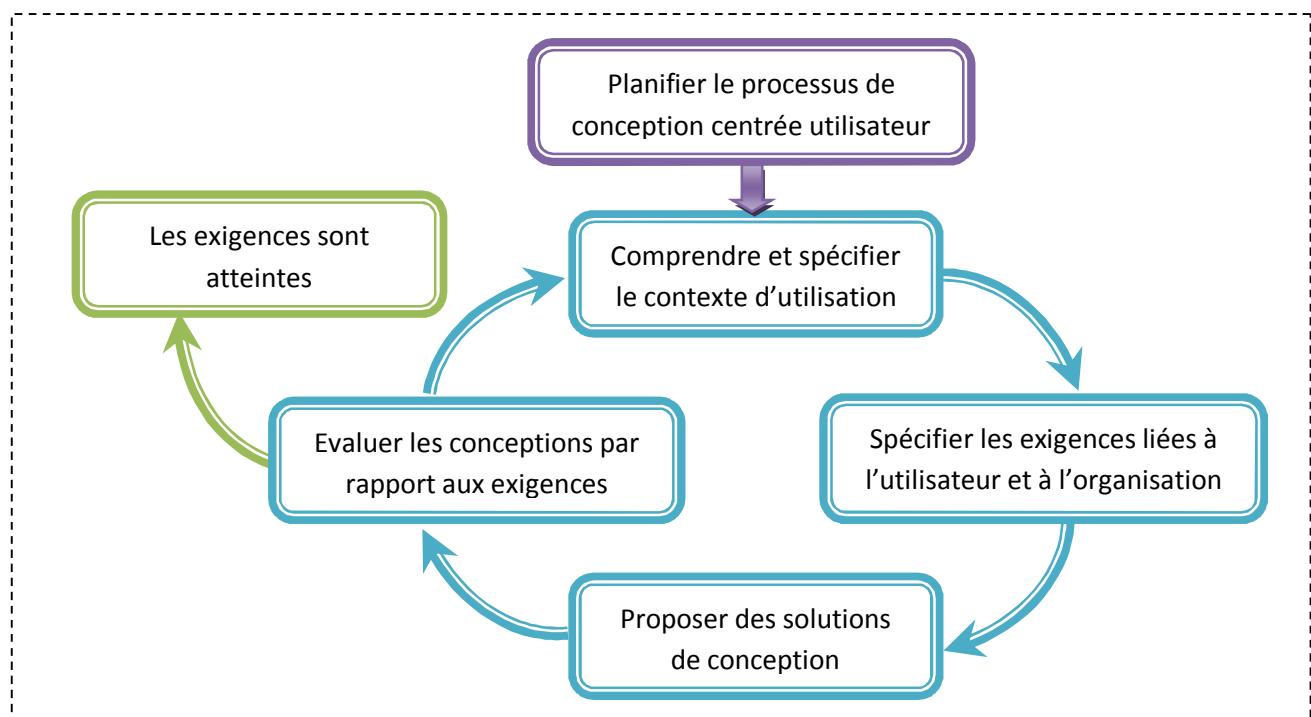


Figure 4.2- Les activités de conception centrée sur l'opérateur humain (source : norme ISO 13407)

4.4 L'implémentation de l'interface graphique multimodale

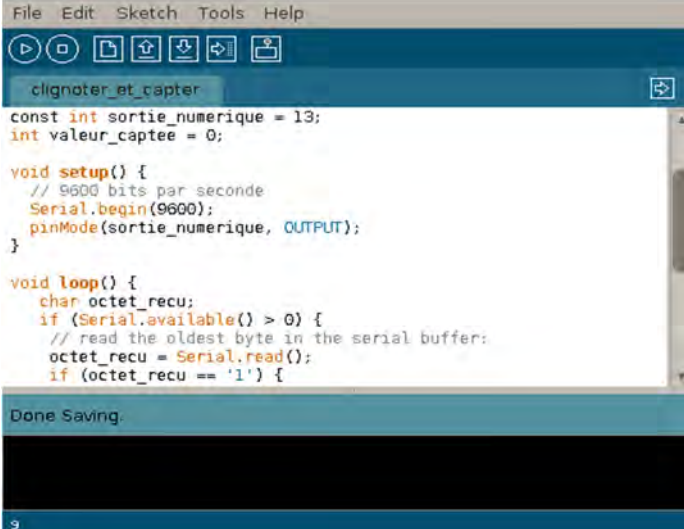
L'existence d'une variété de modes de commande haut niveau dans notre travail, nous a conduit à choisir de commencer par une première version de l'interface graphique multimodale, qui nous donne la possibilité d'intégrer tous les modes de commande haut niveau. Ces modes doivent être présents en même temps, et peuvent être activé ou désactivé par l'utilisateur selon le besoin. Cette

présence simultanée des modes de commande donne la possibilité d'analyser l'interaction homme-robot en commençant par l'analyse de chaque mode de commande séparément, jusqu'à l'analyse des possibilités de combinaison et d'enchaînement de ces modes. Le choix de l'environnement de développement est basé principalement sur la caractéristique de créer et développer des interfaces graphiques avec plus de flexibilité, tout en assurant la capacité de gérer les entrées/sorties telle que le son, le vidéo, joystick et les ports d'interfaçages en temps réel. Pour cela nous avons choisi d'utiliser l'environnement de développement qui s'appel « Processing », qui est principalement orienté ver le développement des interfaces graphiques interactives.

4.4.1 L'environnement de développement Processing

Processing est un logiciel open-source (figure 4.3). Il fonctionne sur les plateformes Windows, Linux, Mac (et sur toute autre plateforme pouvant faire fonctionner des logiciels conçus en Java). Processing est le prolongement « multimédia » de « Design By Numbers », l'environnement de programmation graphique développé par John Maeda au Media Lab du « Massachusetts Institute of Technology ». Le principe majeur de Processing est la simplicité, dans la mise en œuvre des programmes comme dans la syntaxe du langage. Adapté à la création graphique, Processing demande moins d'efforts que Java pour effectuer des tâches telles que la modification ou la création d'une animation. Ses fonctionnalités sont limitées aux besoins des créateurs des interfaces graphique et des animations 2D et 3D mais peuvent être étendues, par le biais de modules externes, pour la capture d'un flux vidéo, la génération et la manipulation de son, et pour l'interfaçage des ports d'entrées-sorties, etc.

Les programmes réalisés avec Processing peuvent être lus par les navigateurs internet équipés du plug-in java, mais aussi sous forme d'applications indépendantes pour Windows, Linux ou Mac OS X (en réalité n'importe quelle machine disposant d'une Machine virtuelle Java).



```
File Edit Sketch Tools Help
clignoter_et_capter
const int sortie_numerique = 13;
int valeur_captée = 0;

void setup() {
  // 9600 bits par seconde
  Serial.begin(9600);
  pinMode(sortie_numerique, OUTPUT);
}

void loop() {
  char octet_recu;
  if (Serial.available() > 0) {
    // read the oldest byte in the serial buffer:
    octet_recu = Serial.read();
    if (octet_recu == '1') {
```

Done Saving.

Figure 4.3- L'environnement de développement Processing

4.4.2 L'implémentation des modes de commande dans l'interface graphique multimodale

Pour l'implémentation des modes de contrôles dans l'interface graphique multimodale d'utilisateur dans sa première version; nous nous sommes basés sur plusieurs propriétés d'apparence et d'interaction pour la rendre efficace pendant la phase d'analyse initiale des modes et de leurs combinaisons:

- Tous les modes de commande haut niveau sont présents en même temps dans l'interface graphique.
- L'opérateur a la possibilité d'activer ou de désactiver manuellement les modes de commande à n'importe quel moment (pour effectuer des combinaisons de modes ou enchainement).
- Assurer le maximum possible de feedback pour chaque mode en tenant compte de la lisibilité de l'interface.
- L'utilisation des icônes symboliques et des couleurs significatives pour rendre l'interface interactive, intuitive et simple à utiliser pour différents opérateurs.
- Enrichir cette interface avec des routines supplémentaires pour assister l'opérateur pour des raisons de sécurité, et pour assurer le bon fonctionnement du système (un simulateur virtuel, les indications sur les limites de l'espace de travail, initialisation automatique du système,... etc.).

4.4.2.1 L'implémentation de la commande vocale

La partie graphique de la commande vocale dans l'interface multimodale est présentée par la figure 4.4. Dans cette zone nous avons trois éléments principaux, un bouton avec une icône symbolique d'un microphone plus un nom du mode « Voice Control ». Ce bouton donne la grande priorité à l'utilisateur pour activer ou désactiver manuellement le mode pour des raisons de sélection, combinaison ou enchainement. Les deux autres éléments concernent le feedback, une LED pour l'état actuel du mode (rouge pour désactivé et vert pour activé). Et une zone de texte qui affiche le mot reconnu par le système, qui donne la possibilité pour l'utilisateur de vérifier ou corriger la commande.

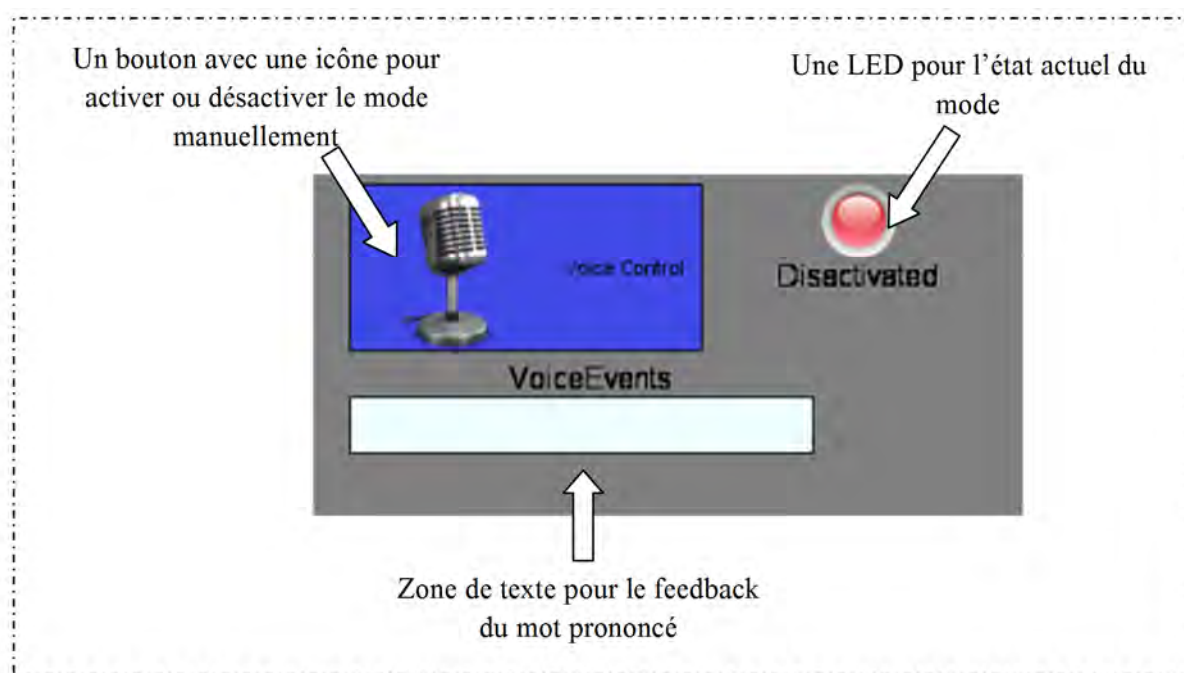


Figure 4.4- L'implémentation de la commande vocale sur l'IHR

La figure 4.4 présente la zone graphique de la commande vocale sur l'interface multimodale dans son état actif avec l'affichage du mot « seven » après détection.

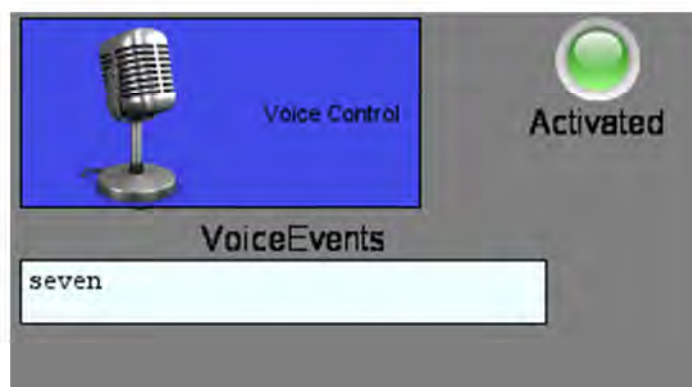


Figure 4.5- L'état reconnaissance du mot « seven »

4.4.2.2 L'implémentation de la commande joystick

La figure 4.6 présente l'implémentation de la commande joystick dans l'interface multimodale. Elle comporte trois éléments, un bouton pour activer ou désactiver le mode « Joystick Control », une LED pour connaître l'état actuel du mode. Et 4 flèches d'indication pour le feedback de la direction du mouvement demandée, pour les différentes directions (x+, y+, x- et y-).

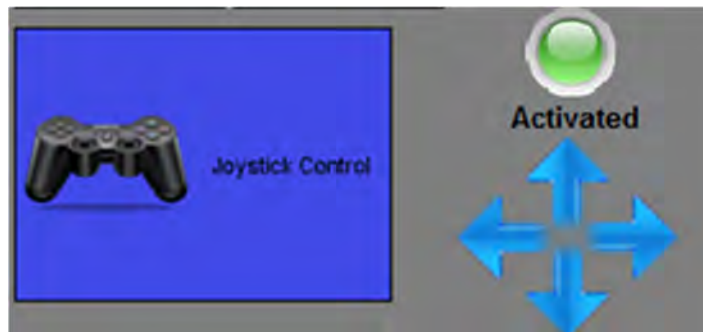


Figure 4.6- L'implémentation de la commande joystick sur l'IHR

Si le mode « Joystick Control » est activé figure 4.7, l'opérateur peut voir sur cette zone graphique la direction demandée sur les 4 flèches. Dans l'état inactif, la flèche est allumée en bleu et si le sens correspondant est demandé, la flèche sera allumée en rouge (figure 4.7.a pour x+, figure 4.7.b pour y+).



Figure 4.7- Les flèches d'indication de la commande joystick

Nous pouvons aussi avoir une combinaison de direction entre les flèches comme pour la configuration présentée par la figure 4.8, un déplacement sur le plan (-x, +y) est demandé.

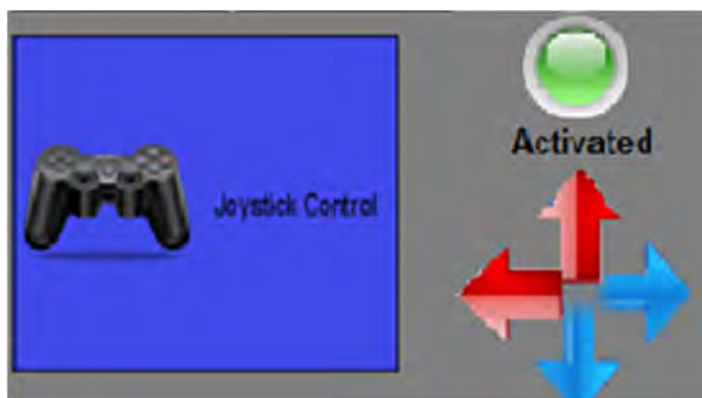


Figure 4.8- Combinaison de direction par la commande joystick

4.4.2.3 L'implémentation de la commande pointage sur image

Pour l'implémentation du mode de commande basé sur l'image, nous avons besoin d'afficher sur l'interface multimodale l'image reçue du site robot. La figure 4.9 présente un état de l'espace de travail du robot à 4 câbles, avec la présence de 4 objets qui peuvent être sélectionnés par l'opérateur. Une zone de texte au-dessous de l'image du site robot est ajoutée pour afficher l'emplacement actuel et en temps réel de l'effecteur final.

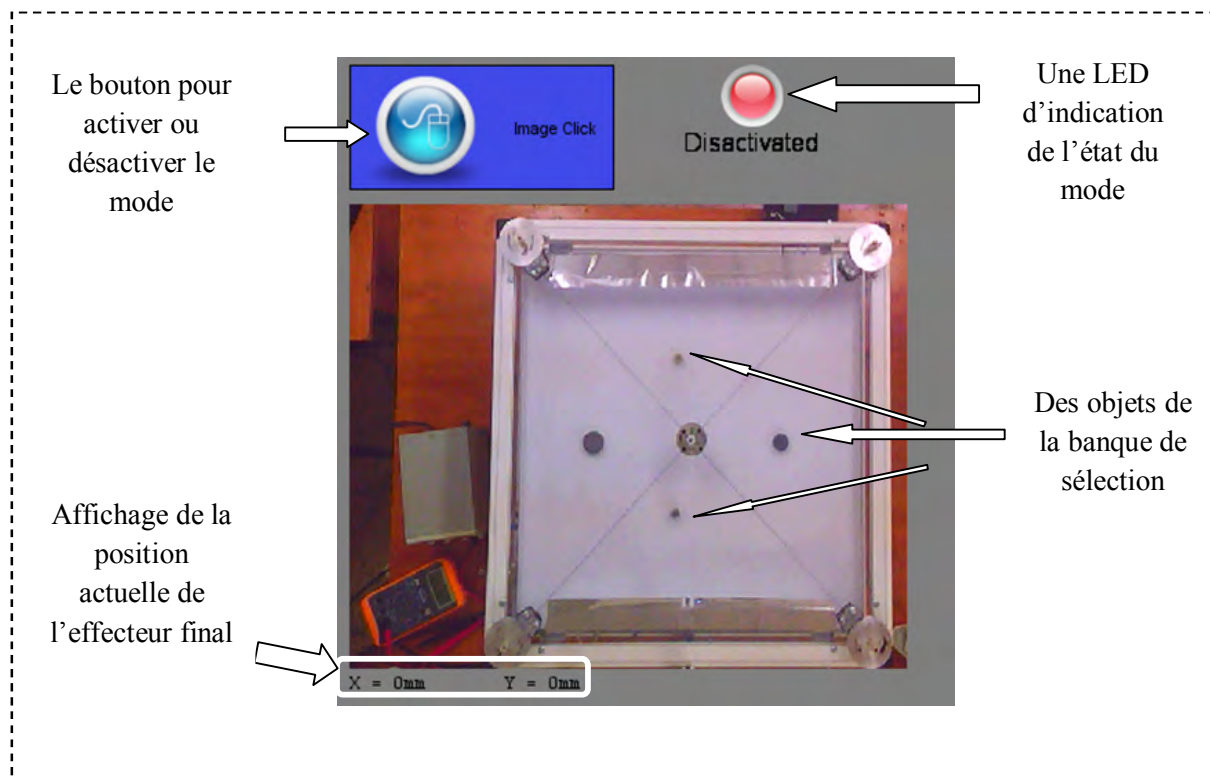


Figure 4.9- La zone de la commande de pointage sur image dans l'interface multimodale

Pour diriger l'effecteur final du robot à 4 câbles vers un objet donné sur l'espace du travail. L'opérateur active le mode de commande en cliquant sur le bouton « Image click ». Ensuite, il pointe en utilisant la flèche de la souris sur l'image de l'objet et click pour engager le robot de se déplacer vers la position réel de l'objet demandé.

La figure 4.10 présente le mode de commande « image-based control » dans son état actif. Nous remarquons que l'effecteur final du robot est superposé sur l'objet sélectionné par l'opérateur pour la position suivante ($x = 210 \text{ mm}$, $y = -20 \text{ mm}$).

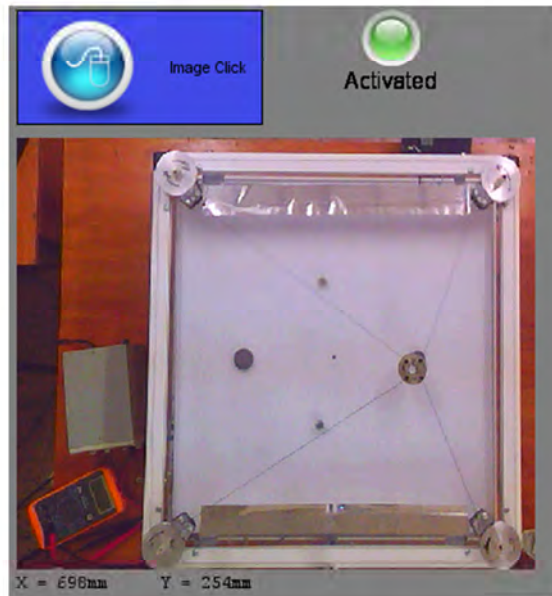


Figure 4.10- Objet sélectionné en utilisant la commande par pointage sur image

4.4.2.4 L'implémentation de la commande gestuelle

La figure 4.11 présente la zone graphique de la commande gestuelle « gestural-based control ». Nous avons un bouton pour activer ou désactiver le mode, une LED d'indication pour l'état actuel du mode et un objet coloré dans l'image reçue de l'espace opérateur, qui est utilisée pour interpréter le geste.

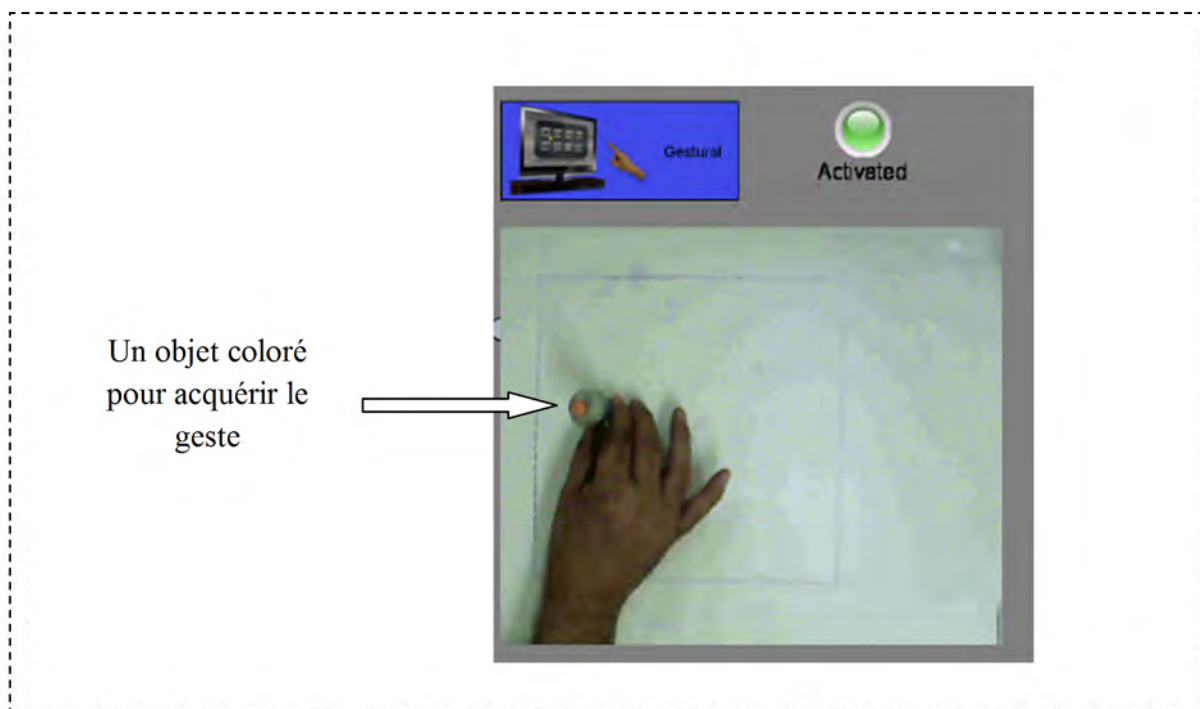


Figure 4.11- La zone graphique pour le mode gestuel

Après la sélection de l'objet coloré par l'opérateur en utilisant un click de la souris sur l'emplacement image de ce dernier, un cercle porte la même couleur de cet objet s'affiche en temps réel sur leur emplacement (figure 4.12). Le cercle indique que le système de vision est en état de poursuite de l'objet.

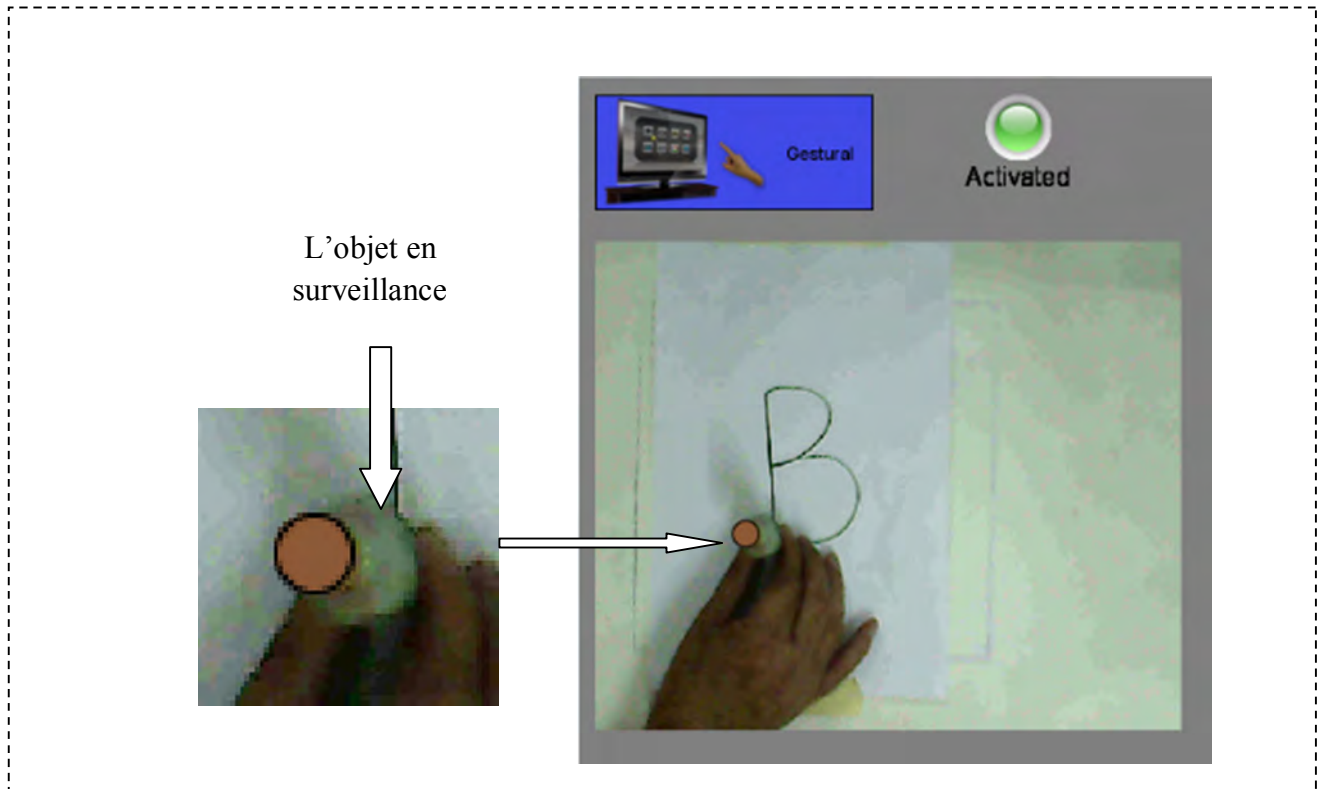


Figure 4.12- Un objet coloré sélectionné en mode gestuel

4.4.3 Les routines supplémentaires implémentées

Pour des raisons de sécurité et pour assurer le bon fonctionnement du système; nous avons ajouté des routines supplémentaires à notre interface multimodale. Parmi ces routines, nous avons : une routine d'initialisation automatique du robot à câbles, un simulateur virtuel du robot à 4 câbles et deux zones de dessin, la première pour afficher le dessin capturé par le mode gestuel, et la deuxième pour afficher les deux trajectoires du système (désiré et actuel).

4.4.3.1 L'initialisation automatique du robot à câbles

Parmi les problèmes que nous avons rencontrés pour contrôler ce nouveau type de robots parallèle qui est le robot à câbles, nous avons le problème d'initialisation du système. Ce problème est dû à la dé-configuration du système mécanique avant le démarrage du système global (la partie électronique plus le programme de control). Toute manipulation sur le robot qui provoque un changement de sa position par rapport à la configuration initiale, ne sera pas détecté. Comme par exemple :

- Une manipulation qui provoque un décalage de l'effecteur final par rapport au point de référence (0,0).
- Une manipulation qui provoque un relâchement des câbles au niveau des poulies ou au niveau de l'effecteur final.
- L'exécution d'une série d'expériences qui nécessitent le retour au point (0,0) suite à un cumul d'erreurs.

Pour assurer un bon fonctionnement, il faut initialiser le système soit après le redémarrage du système global (après un arrêt total), ou en cours de l'exécution de plusieurs actions (une série d'expériences qui nécessitent le retour au point (0,0)). Dans ce but, un sous-programme est ajouté qui est décomposée en deux phases:

- Au cours de la première phase d'exécution, on applique la tension minimale sur les câbles jusqu'à la rectification des relâchements tout en assurant l'équilibre dynamique de l'effecteur final à une position donnée.
- La deuxième phase a pour but de localiser en temps réel l'emplacement de l'effecteur final sur l'espace de travail, dont la technique est basée sur la détection du couleur de l'effecteur final par la camera CCD du site robot (figure 4.13). Une conversion à l'aide de la matrice de projection de la camera est appliquée pour calculer la position réel de l'effecteur final. Dans ce cas l'opérateur a la possibilité de diriger automatiquement l'effecteur vers le point initial (0,0) qui est donné comme consigne avec un simple clic sur le bouton « Homing » (figure 4.14).



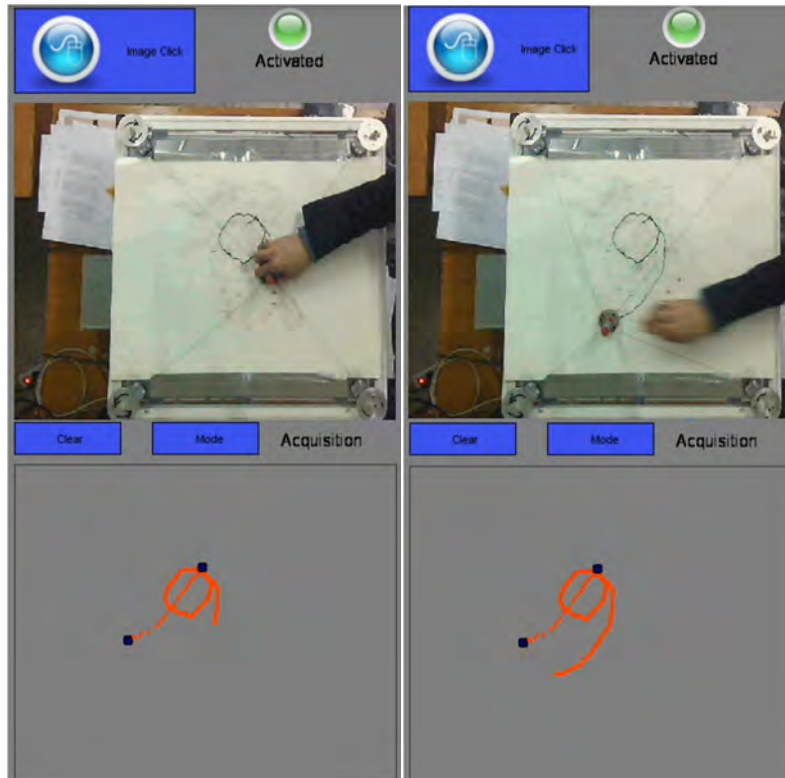
Figure 4.13- La localisation automatique de l'effecteur final sur l'espace du travail



Figure 4.14- Initialisation automatique de l'effecteur final

4.4.3.2 Le mode d'acquisition

Afin d'intégrer des capacités d'apprentissage à notre prototype expérimental, un mode d'acquisition est ajouté à notre interface multimodale. Dans ce cas, le système expérimental dispose de deux modes de fonctionnement: le mode control et le mode acquisition. L'opérateur sélectionne le mode désiré en utilisant le bouton « Mode », et une zone de texte affiche le mode actuellement sélectionné.



(a) .En cours d'acquisition (b) .Fin de l'acquisition

Figure 4.15- L'acquisition de la lettre « و » avec le mode acquisition

Si le mode acquisition est sélectionné, dans ce cas le système est commandé pour assurer l'équilibre dynamique seulement. Dans ce cas, une tension égale à la tension minimale est appliquée sur chaque câble afin d'éliminer les relâchements au niveau des câbles provoquée par la force exercé par l'opérateur pour déplacer l'effecteur final (figure 4.15).

4.4.3.3 Les zones graphiques

Pour assurer le feedback du mode gestuel, une zone graphique a été ajoutée à notre interface multimodale. La figure 4.16 présente l'acquisition de la lettre « ح » en utilisant le mode gestuel. Dans cette figure, nous remarquons l'existence de deux éléments de feedback pour l'utilisateur. Un cercle portant la couleur de l'objet poursuivi est affiché en temps réel sur l'image du site opérateur en haut plus l'affichage en temps réel des points calculés par la commande gestuel «**points gestuels**» sur la zone graphique en bas.

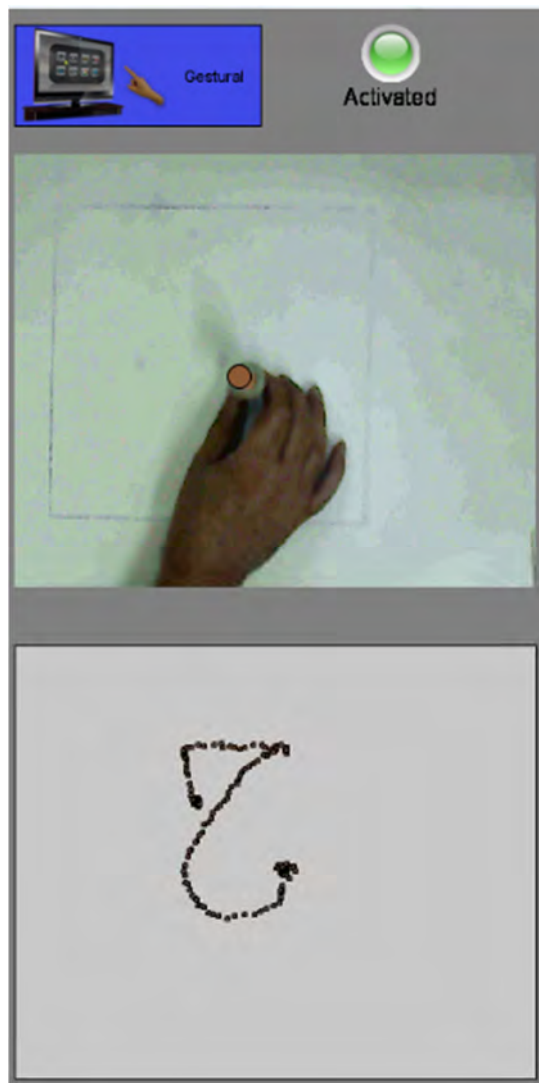


Figure 4.16- L'acquisition de la lettre « ح » en mode gestuel

L'implémentation de plusieurs modes de commande qui sont basés sur l'image (gestuel ou pointage sur image), rend l'attention de l'opérateur toujours dirigée vers l'interface non pas vers sa main. Dans ce cas, l'affichage des points gestuels sur cette zone graphique aide l'opérateur de corriger son geste en temps réel par lui-même.

Comme à chaque instant donné, nous avons un contrôle exclusif du prototype expérimental par des différents modes de commande haut niveau. Donc nous avons toujours une commande demandée et une commande réalisée. Pour cela nous avons utilisé une deuxième zone graphique pour afficher les deux trajectoires (demandée et réalisée). La figure 4.17 présente l'état de cette zone graphique après la réalisation de la lettre « ح » en utilisant les modes hauts niveau. La trajectoire en bleu représente la trajectoire désirée et celle en rouge représente la trajectoire actuelle. Le bouton « Clear » est utilisé pour initialiser cette zone.

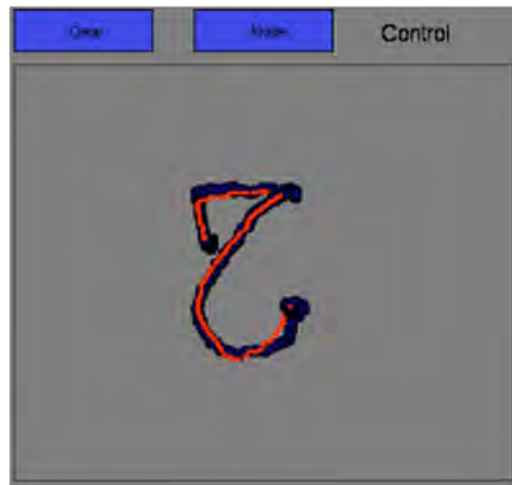


Figure 4.17- La zone graphique en mode contrôle

La figure 4.18 présente l'état de cette zone pendant l'acquisition de la lettre « و » utilisant le mode « acquisition ».

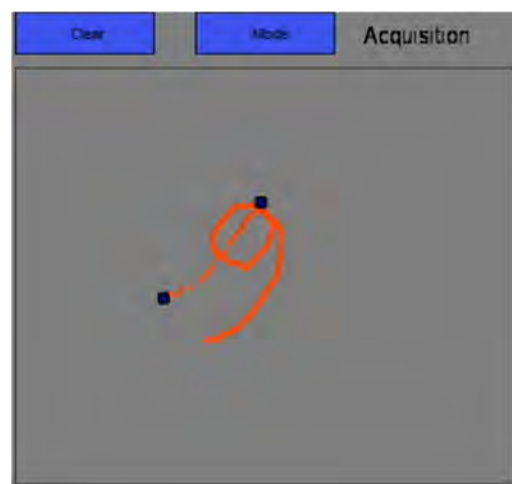


Figure 4.18- La zone graphique en mode acquisition

4.4.3.4 Le simulateur virtuel

La modélisation virtuelle est maintenant un outil important pour la conception, la planification, l'analyse et la prise de décision dans différents domaines de recherche et de développement. En particulier, elle est devenue un outil stratégique en robotique. Dans ce but, l'intégration d'un simulateur virtuel qui travaille soit en mode réalité virtuelle (figure 4.19.a) ou augmentée (figure 4.19.b), dans notre interface multimodale donne plus d'avantages à notre étude en terme de sécurité et d'efficacité en apprentissage.

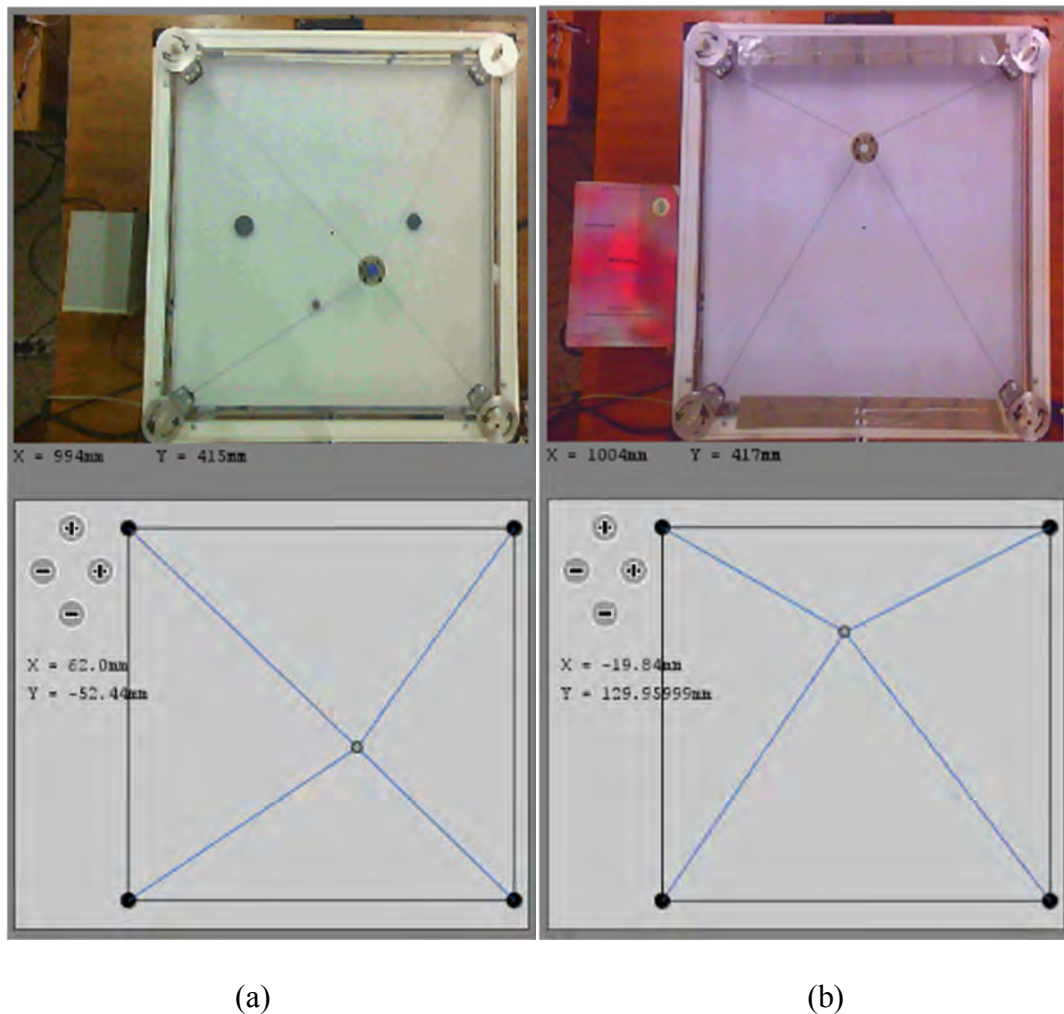


Figure 4.19- Le simulateur virtuel 2D

La figure 4.19.a présente l'état du simulateur virtuel du robot à 4 câbles en mode réalité virtuelle. Dans ce cas, le simulateur est configuré selon les données reçues des différents modes de command haut niveau avant de les transmettre vers le contrôleur bas niveau. Pour le mode réalité augmenté (figure 4.19.b), le simulateur est configuré selon les données réelles reçues à travers le prototype expérimental. Dans ce mode, l'opérateur peut, par exemple, continuer l'exécution de la tâche si l'image du site robot est perdu pendant une télé-opération.

4.4.4 L'interface multimodale globale

Conformément aux propriétés d'apparence et d'interaction citées précédemment, pour rendre l'interface graphique multimodale efficace pendant la phase d'analyse initiale des modes et de leurs combinaisons, nous avons choisi une configuration et placement matricielle des modes dans notre interface, qui nous donne la possibilité d'analyser les différentes possibilités de combiner ou d'enchaîner les modes de commande.

La figure 4.20 présente l'interface multimodale globale où on remarque l'intégration de tous les modes de commande haut niveau ainsi que les routines d'assistance. En utilisant cette configuration, l'opérateur a la possibilité d'activer ou de désactiver chaque mode manuellement pour une éventuelle combinaison ou enchaînement.

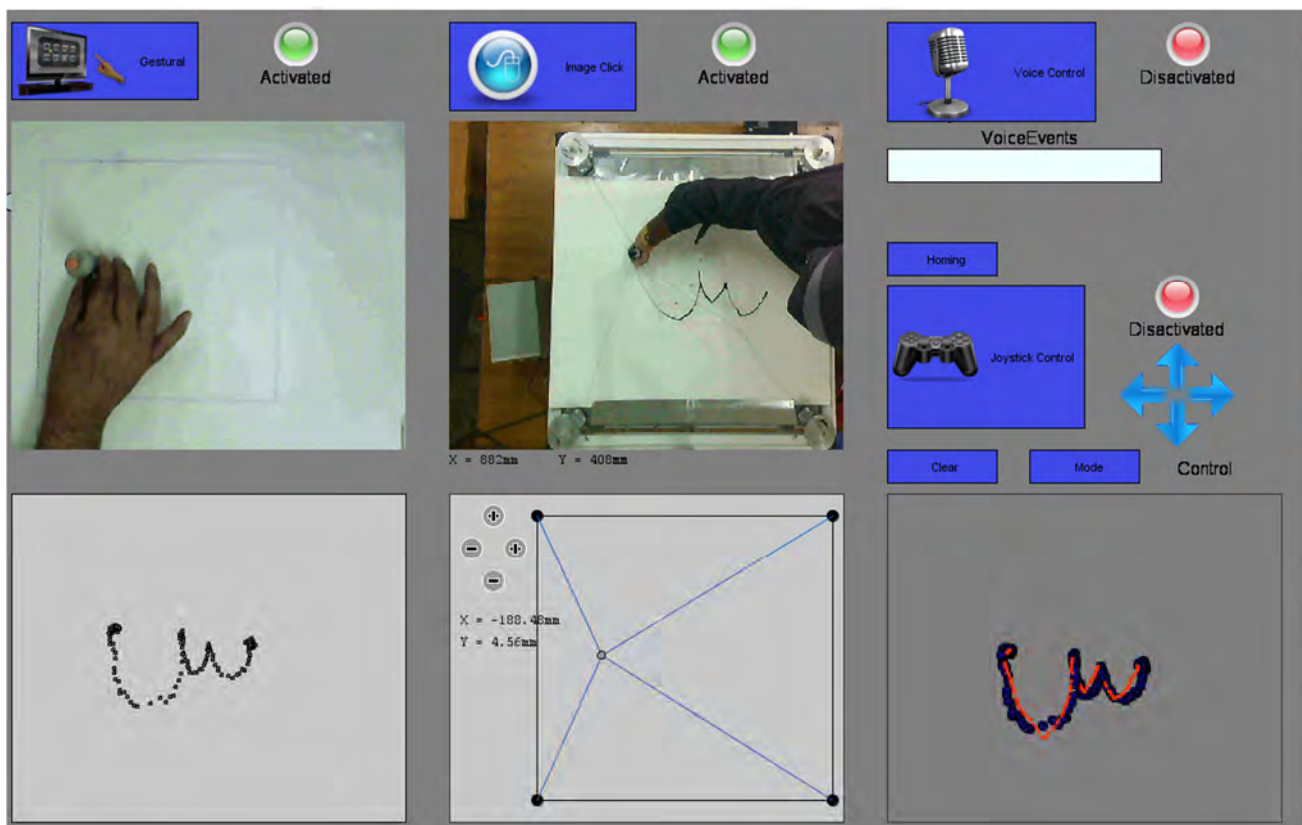


Figure 4.20- L'interface graphique multimodale globale

Nous remarquons dans la figure 4.20 que l'image du site robot est située au centre de l'interface multimodale.

4.4.5 Développement de l'interface graphique pour le robot manipulateur sous Labview

Dans le but de généraliser notre analyse préliminaire sur l'interaction homme-robot pour concevoir une interface multimodale généralisable, nous avons besoin d'analyser la dépendance de cette dernière par rapport aux modes de commande, au système expérimental utilisé et même à l'environnement de programmation. Afin de travailler dans un niveau d'abstraction élevé (orienté mode de command, system, tâche). Pour cet objectif, nous avons conçu une autre interface multimodale (figure 4.21) qui sert à contrôler un autre prototype expérimental qui est un bras manipulateur à 5 degrés de libertés porté sur une base mobile de type ROVER. Implémentée en utilisant un autre environnement de programmation qui est le logiciel LabVIEW.

Le logiciel LabVIEW « **L**aboratory **V**irtual **I**nstrument **E**ngineering **W**orkbench » est un environnement de programmation graphique « le langage G », développé par la société américaine NI (National instruments) pour l'acquisition, l'analyse et la visualisation des données. L'interface graphique d'utilisateur est appelée face-avant, et la face-arrière est appelée diagramme qui contient le code à exécuter. La notion de VI (Virtuel instrument) et sous VI permet de créer des applications modulaires et extensibles.

Notre interface graphique d'utilisateur implémentée sous l'environnement Labview est présentée par la figure 4.21. Cette interface est divisée en trois parties graphiques : la partie du simulateur virtuel 3D plus l'affichage de l'image de l'espace de travail reçu par la camera CCD, la zone d'acquisition du geste graphique et la troisième partie est la commande manuelle en utilisant les sliders. Ce qui concerne le feedback, des indicateurs textuels et des LED d'indication sont ajoutés : une LED pour les positions non atteignables, une camera CCD pour superviser l'espace de travail.

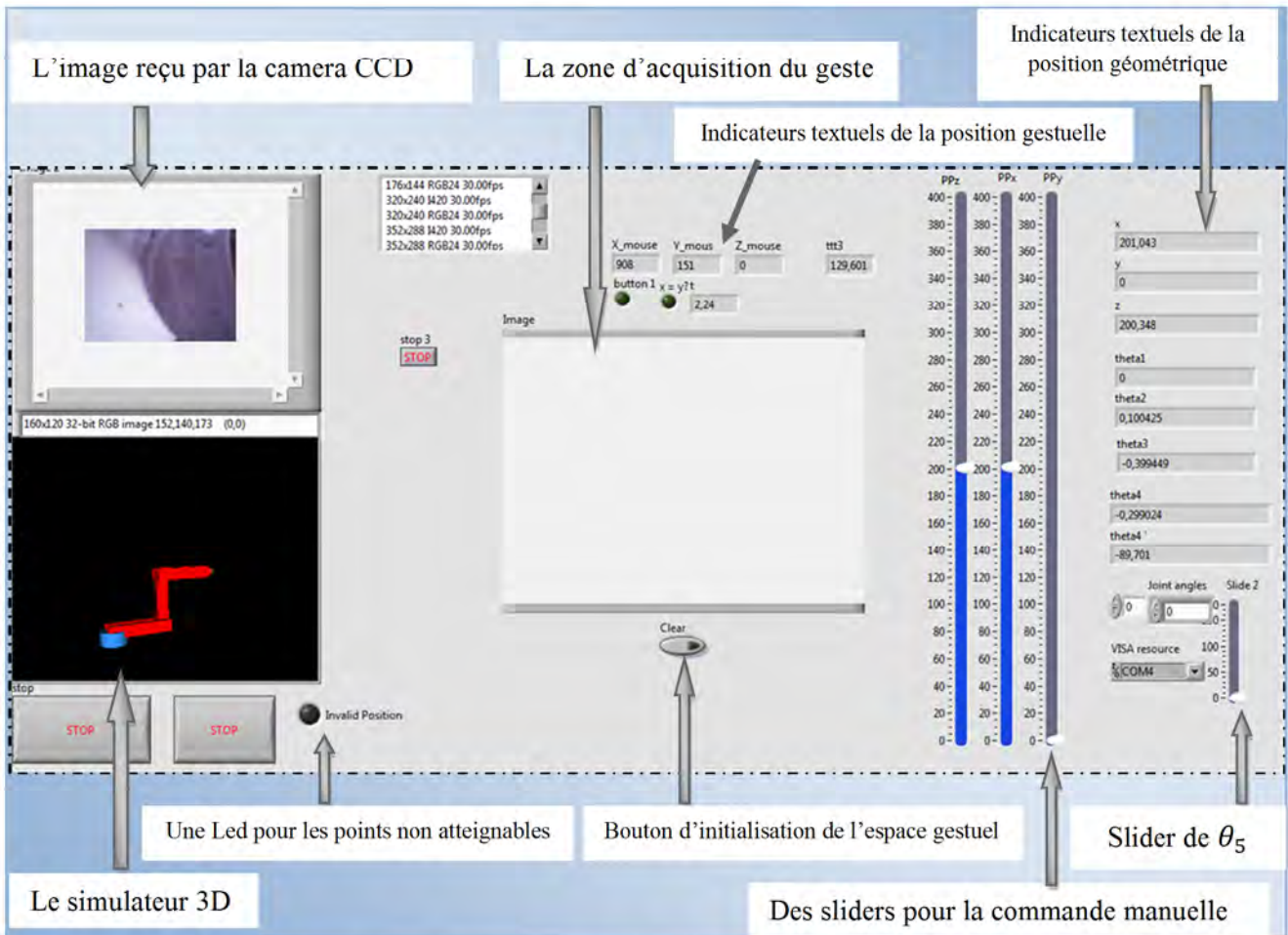
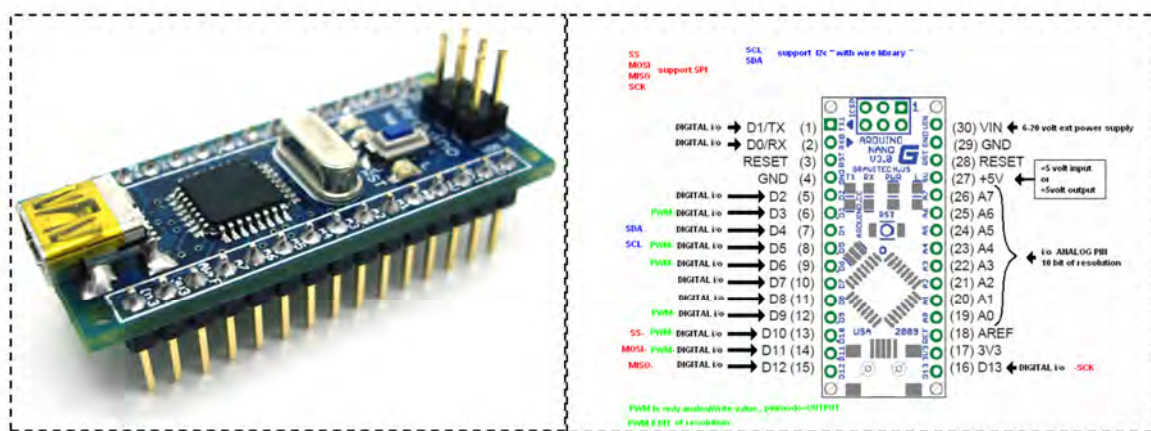


Figure 4.21- L'interface graphique d'utilisateur pour commander le robot

Le diagramme du code pour contrôler les servomoteurs est présenté par la figure 4.23, le signal PWM (Pulse Wide Modulation) ou MLI (Modulation du Largeur d'Impulsion) est généré par une carte open-source de type Arduino NANO (figure 4.22.a). Dans ce cas, nous avons configurés les sorties numériques en mode PWM comme il est indiqué sur la figure 4.22.b.



(a)

(b)

Figure 4.22- La carte Arduino nano

Pour l'affectation des angles pour les différents servomoteurs, nous avons utilisé une bibliothèque pour les cartes de type Arduino sous l'environnement Labview. Le diagramme de ce sous program est présenté dans la figure 4.23.

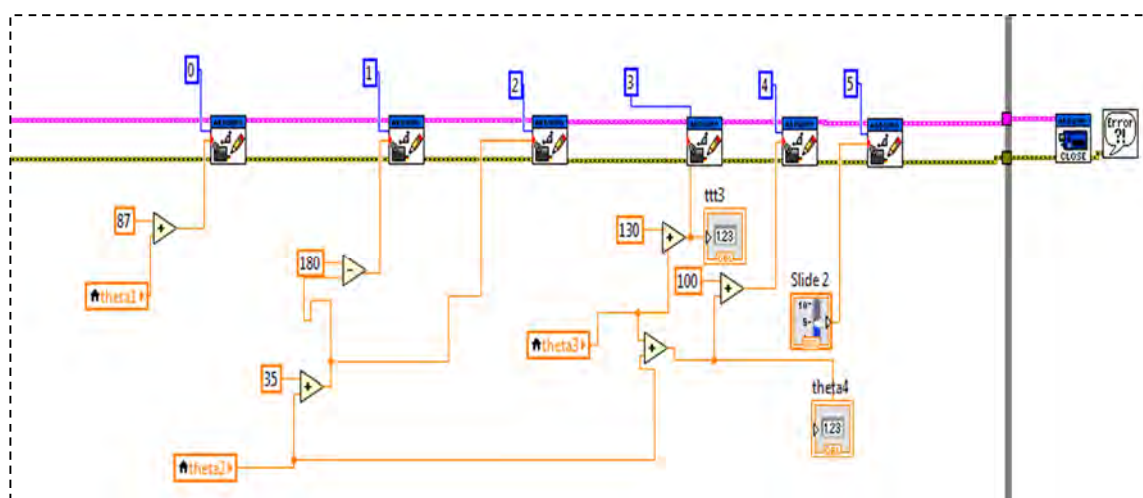


Figure 4.23- Affectation des angles et espace de travail pour les servomoteurs

Le diagramme pour la visualisation 3D du simulateur virtuel est présenté par la figure 4.24, dans ce cas, le simulateur virtuel est contrôlé en mode réalité virtuelle non pas en réalité augmentée c'est-à-dire, le simulateur virtuel reçoit les valeurs des articulations à travers le modèle géométrique inverse calculé par le sous-programme implémenté sur Labview.

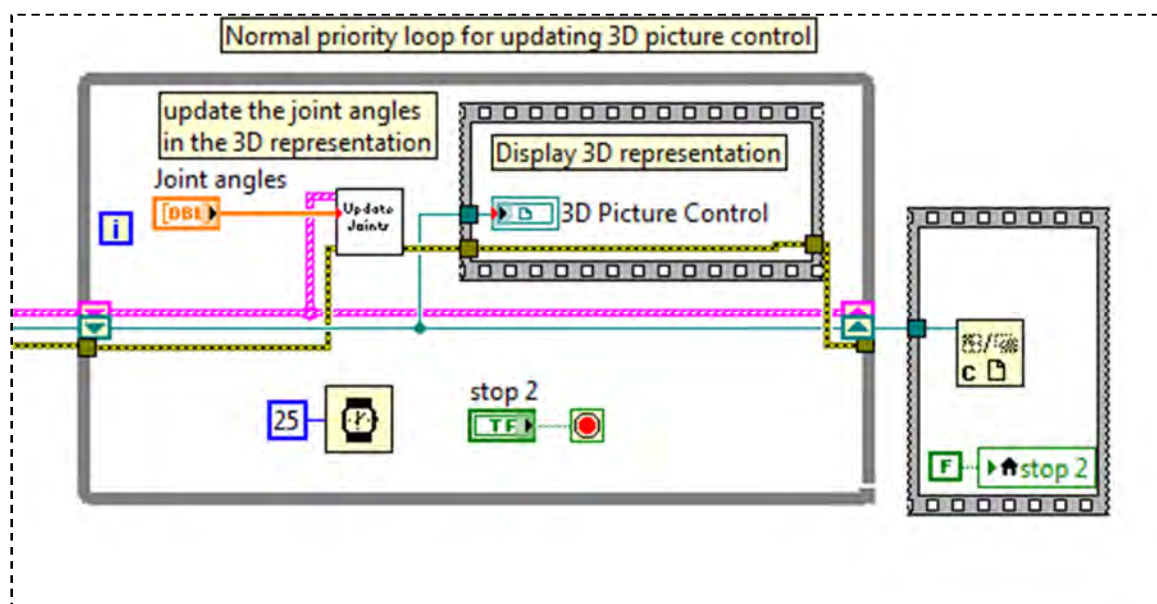


Figure 4.24- Implémentation de la scène 3D pour le simulateur virtuel

Nous avons implémenté le modèle géométrique en utilisant le block « MATLAB script », ce block nous donne la possibilité de le programmer directement en code Matlab sous l'environnement Labview figure 4.25.

```

MATLAB script
%PPx
Px0 = 200;
%PPy
Py0 = 0;
%PPz
Pz0 = 200;
A = 50;

Px= 150;
Py = A*cos(2*pi*t);
Pz = 300 + A*sin(2*pi*t);

dx = PPx - Px0;
dy = PPy - Py0;
dz = PPz - Pz0;
Px = Px0 + dx;
Py = Py0 + dy;
Pz = Pz0 + dz;

eps1 = 1;
eps2 = 1;
D3 = 200;
RL4 = 200;
L2 = D3;
L3 = RL4;
%
theta1 = atan2(Py,Px) ;

B1 = Px*cos(theta1) + Py*sin(theta1);
X = -2*Pz*D3;
Y = -2*B1*D3;
Z = (RL4)^2 - (D3)^2 - (Pz)^2 - (B1)^2;

C2 = (Y*Z - (eps1 * X * sqrt(X^2+Y^2-Z^2)))/(X^2+Y^2);

S2 = (X*Z + (eps2 * Y * sqrt(X^2+Y^2-Z^2)))/(X^2+Y^2);

theta2 = atan2(S2,C2) ;
    
```

Figure 4.25- Implémentation du modèle géométrique inverse avec le block « Matlab script »

L'interface graphique d'utilisateur globale (face-avant) est présentée dans la figure 4.26 après la réalisation d'une tâche donnée qui est dans ce cas l'écriture de la lettre (ζ).

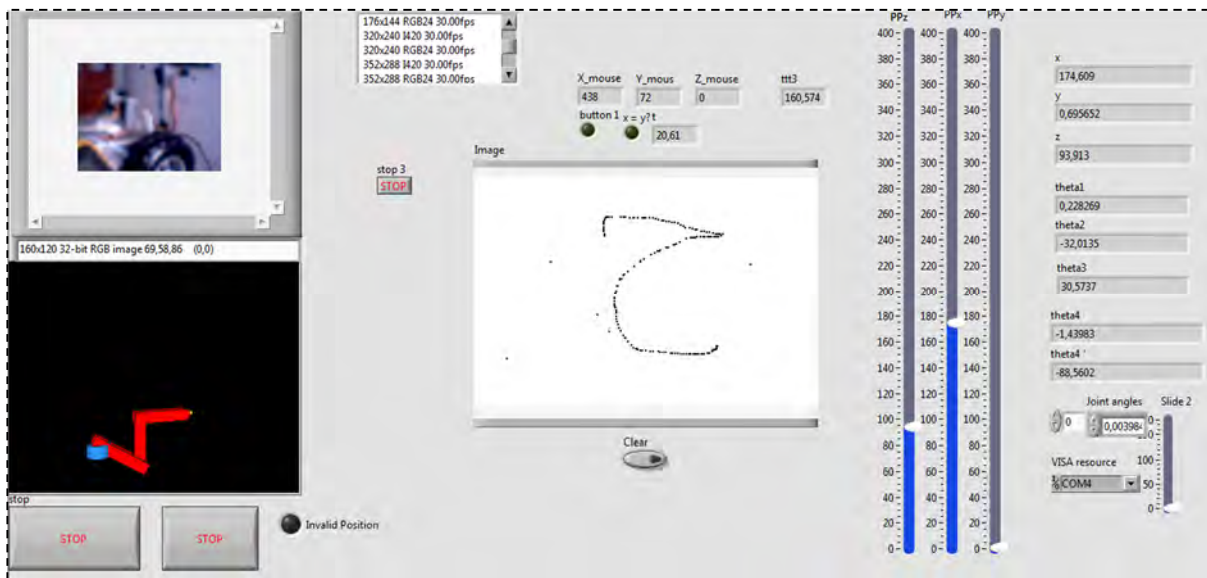


Figure 4.26- L'interface graphique après l'acquisition d'un geste graphique

Le sous-programme pour l'acquisition du geste graphique est présenté dans la figure 4.27. Dans cet exemple nous avons utilisé le geste de la souris qui est un geste graphique plan pour des applications d'écritures.

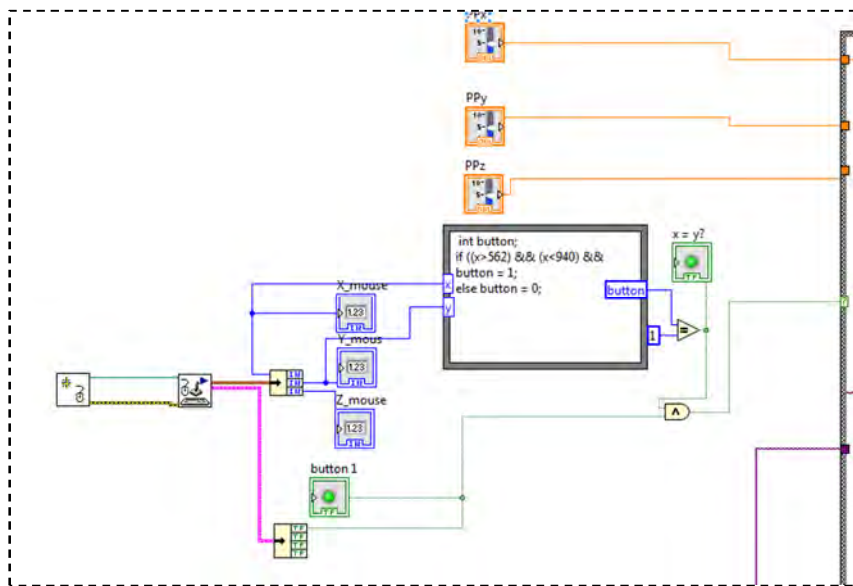


Figure 4.27- Diagramme de sous-programme d'acquisition du geste graphique

4.5 Conclusion

Dans ce chapitre, nous avons présenté la conception et une implémentation de l'interface multimodale graphique utilisateur basée sur la technique de conception centré utilisateur. Cette technique a permis de concevoir des interfaces qui soient ergonomiques, efficaces, faciles à utiliser ou plus généralement adaptées à leur contexte d'utilisation. Elle est basée sur le principe que les utilisateurs finaux sont les mieux placés pour évaluer sa qualité. Cette méthode a été homologuée par les normes internationales pour le développement des interfaces graphiques. Dans ce but, nous avons pendant la phase d'analyse, implémenter plusieurs propriétés d'apparence et d'interaction pour rendre l'interface multimodale efficace et intuitive. Nous avons également ajouté des routines supplémentaires pour assister l'opérateur dans la gestion du système global pendant la réalisation de la tâche. Parmi ces routines, nous noterons : un simulateur virtuel, des indications sur les limites de l'espace du travail, routine d'initialisation automatique du système, ..., etc.

Les exigences du développement d'une interface multimodale basée sur de nombreux modes et basée sur la CCU, nous a conduit à concevoir et à implémenter une version initiale de notre interface graphique multimodale pour nous permettre d'analyser chaque mode de commande séparément, puis combinés ou enchainés.

Nous avons implémenté l'interface graphique multimodale sous deux environnement différents afin de tester deux systèmes robotiques supervisés différents. Le premier environnement est le Processing qui a servi à commander les robots à câbles; et l'environnement de programmation Labview qui a servi à superviser un bras manipulateur.

Chapitre 5

Analyse des Modes et leurs Combinaisons

Sommaire

5.1	Introduction	139
5.2	L'analyse de chaque mode de command séparément	139
5.2.1	L'analyse du mode vocal.....	139
5.2.2	L'analyse du mode de pointage sur image	143
5.2.3	L'analyse du mode gestuel	147
5.2.4	L'analyse de mode joystick	152
5.3	L'analyse de combinaisons des modes	154
5.3.1	La combinaison du mode vocal avec le mode de pointage sur image	154
5.3.2	La combinaison du mode gestuel avec le mode basé sur image.....	157
5.4	L'analyse de la dépendance par rapport à l'environnement de programmation et le système expérimental utilisé	161
5.5	Conclusion	165

5.1 Introduction

Après l'implémentation de notre interface multimodale globale selon les principes de conception centrée sur l'utilisateur, nous abordons dans ce chapitre l'analyse des différents modes et leurs combinaisons. Pour cela, nous avons effectué une étude préliminaire sur l'interaction homme-robot sur la base de nos analyses et observations expérimentales.

Dans la première partie de ce chapitre, nous commençons par présenter une analyse de chaque mode de commande séparément : le mode vocal, le mode basé sur le pointage sur image, le mode gestuel et le mode joystick. Nous discutons certains avantages et inconvénients de chaque mode ainsi que les différents facteurs intervenant au cours de la réalisation de la tâche demandée. Ensuite, nous examinons la possibilité de combiner les modes deux à deux que nous illustrons par deux applications générales et spécifiques. Dans la première combinaison, nous présentons l'utilisation du mode vocal avec le mode de pointage sur image afin de créer une banque d'objets images. La deuxième combinaison utilise le mode de pointage sur image avec le mode gestuel pour des applications de réhabilitation concernant les personnes ayant des exigences particulières. Nous avons appliqué et testé l'interface multimodale pour superviser des robots à câbles sous l'environnement Matlab.

Dans le but de généraliser notre analyse préliminaire sur l'interaction homme-robot; nous présentons dans la deuxième partie de ce chapitre, une application de la multimodalité sur un autre système robotique expérimental qui est un bras manipulateur en utilisant un autre environnement de programmation LabView.

Nous présentons à la fin de ce chapitre, des observations et des analyses sur la dépendance de l'interface multimodale par rapport aux modes de commande, au système expérimental et même à l'environnement de programmation. On discutera de l'influence de la nature et le nombre de modes disponibles, les facteurs humains, les performances du système opérationnel à conduire, le degré de complexité de la tâche ou de la mission.

5.2 L'analyse de chaque mode de command séparément

5.2.1 L'analyse du mode vocal

La commande vocale a été implémentée en utilisant la librairie «VOCE», bien que cette bibliothèque exploite les points forts de logiciels existants. Nous avons remarqué plusieurs difficultés d'implémentation et d'utilisation de ce mode de commande qui nécessite une configuration pseudo spéciale soit pour l'environnement ou pour le matériel d'implémentation. Ces exigences et difficultés influent directement sur la qualité de l'interaction ; et sur les possibilités d'utilisation de ce mode de commande.

Comme VOCE prend des chaînes de texte à partir des applications et les transmet à un synthétiseur TTS qui les convertit en une sortie audio, et alloue un microphone Sphinx4 qui écoute en permanence aux données audio entrantes du matériel audio de l'utilisateur pour la reconnaissance vocale. L'utilisation en parallèle de ces deux fonctions de ce mode (la synthèse et la reconnaissance) qui sont exécutés dans des threads séparés du thread principal de l'application, plus la présence de plusieurs modes de command haut niveau qui sont présents en même temps sur l'interface multimodale, exige une architecture multiprocesseur de l'ordinateur utilisé pour assurer un meilleur parallélisme et temps de réponse.

Deux problèmes importants se posent pendant le développement concernant le threading:

- 1) synthèse de la parole prend une quantité importante de temps pour produire sa première sortie audio lors de l'exécution d'une application.
- 2) La fonction de reconnaissance CMU Sphinx4 primaire n'obéit pas à la règle de file d'attente. parce qu'elle doit terminer prioritairement la reconnaissance.

Dans notre cas, nous avons utilisé un ordinateur de type HP Core I7 à cause de plusieurs caractéristiques. La commercialisation des Core i7 marque le retour de la technologie Hyper threading qui avait disparu depuis les Pentium 4. L'Hyper Threading ou SMT (Simultaneous Multi Threading) à deux voies signifiant que chaque processeur recevra des instructions à traiter par 2 threads et non plus un seul comme auparavant. Ces deux threads partagent des données comme des registres, le cache le pipeline et le bus. Bien évidemment un système de contrôle a été mis en place afin de gérer les dead-locks pouvant survenir. L'Hyper Threading est une technologie très répandue, il existe aussi du SMT à 4 voies, mais celui-ci est plus généralement utilisé pour des machines plus importants (super calculateur, serveur). Cette technique permet d'accélérer le traitement des instructions et donc les performances des machines basées sur un Core i7. Il est possible de désactiver cette option dans le bios. En activant l'HT, l'OS (Operating system) perçoit le processeur comme un processeur à 8 cœurs et non plus 4. On dit que c'est un processeur à 8 cœurs logique.

Cette utilisation du cœur I7 plus l'utilisation du noyau de notre interface multimodale en « time sharing » a donné plus de flexibilité à notre interface multimodale et plus précisément à la commande vocale, en termes de temps de réponse. Et en termes de qualité du signal à traiter nous recommandons un microphone de qualité avec un system de suppression du bruit et une valeur optimale de la carte sons en termes de fréquence d'échantillonnage et temps de traitement.

Selon notre propre expérience sur l'utilisation de la commande vocale au cours de cette analyse préliminaire on peut citer les remarques suivantes :

- Sensible au bruit du fond du matériel utilisé et de l'environnement.
- L'utilisation en tant que système de reconnaissance « mono-utterances ».
- Le système perdre la précision de la reconnaissance lors de l'utilisation de longues phrases.
- Il nécessite un intervalle de silence entre chaque commande pour un meilleur fonctionnement.
- L'utilisation de ce mode séparément sans leur combiné avec d'autre mode s'avère difficile pour accomplir des tâches complexes.

Sur la base de ces remarques, nous avons choisi de diviser l'instruction donnée par le mode vocal en deux étapes pour assurer le bon fonctionnement du système. Si en prenant l'exemple d'activation du mode joystick en utilisant la commande vocale. Dans ce cas, à la place d'utiliser la commande « joystick enable » enchaînée nous avons la divisée en deux sous commandes. Dans la première étape, l'opérateur prononce le nom du mode que nous voulons l'activer, par exemple « joystick ». Si cette commande est correctement détectée elle est affichée dans la zone « Voice Events » dans la zone graphique de la commande vocale, après cette étape, toute commande n'a pas de relation avec le mode joystick n'est pas prise en considération par le système de reconnaissance. Pour la deuxième étape, l'opérateur peut activer ou désactiver le mode joystick en prononçant le mot « Enable » ou « Disable » selon le besoin. Pendant cette étape, la commande est exécutée directement sans donner un temps de vérification pour l'opérateur. Dans l'exemple récapitulatif suivant, nous allons offrir un exemple d'application concernant l'utilisation du mode vocal pour activer au désactiver les modes de commande vocalement. Prenant l'exemple de l'activation du mode vocal par lui-même. La figure 5.1 présente la détection du nom du mode « voice », dans ce cas le mode « voice » est prêt pour être activer. Sur la figure 5.2, nous pouvons remarquer que la LED du mode vocal est allumé en vert pour indiquer que le mode est activé directement après la détection du mot « Enable ».

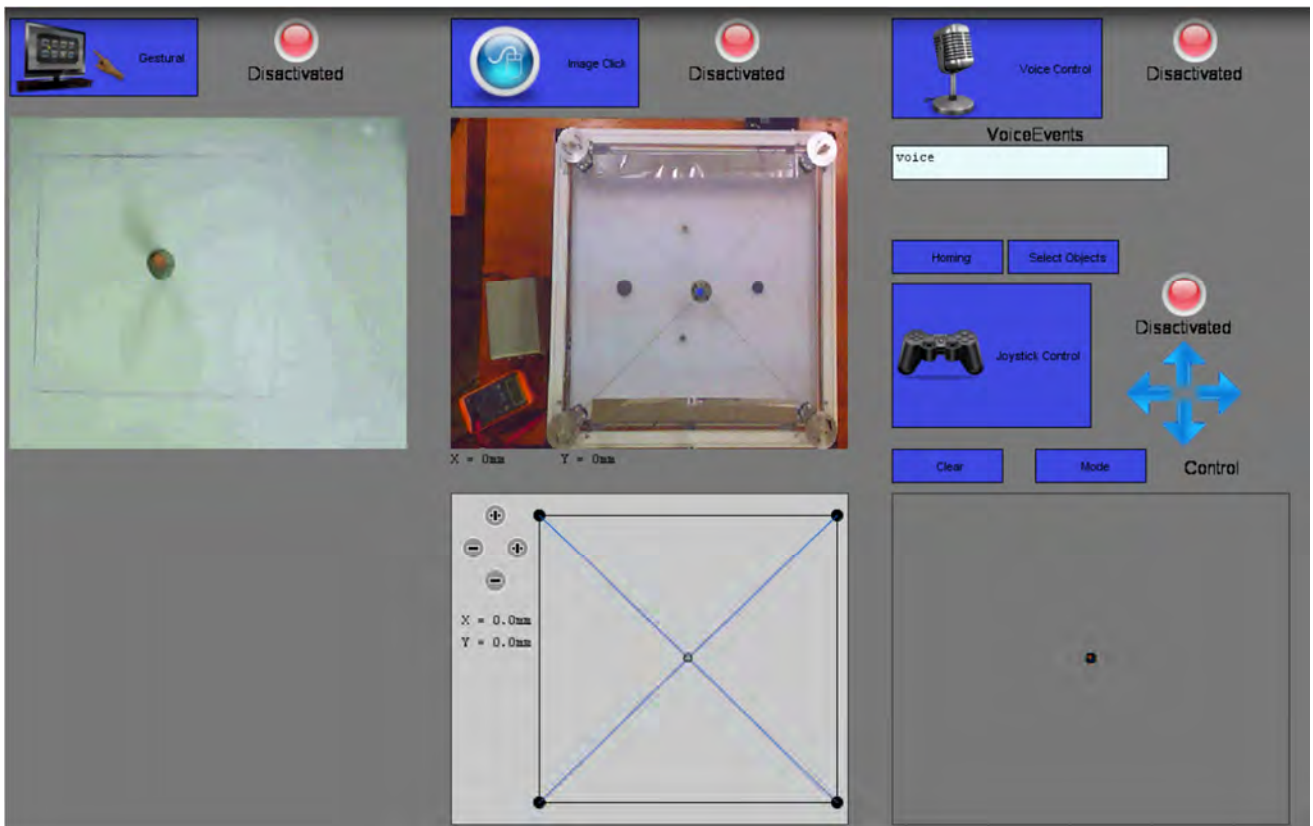


Figure 5.1- L'interface multimodale pendant la détection du mot « voice »

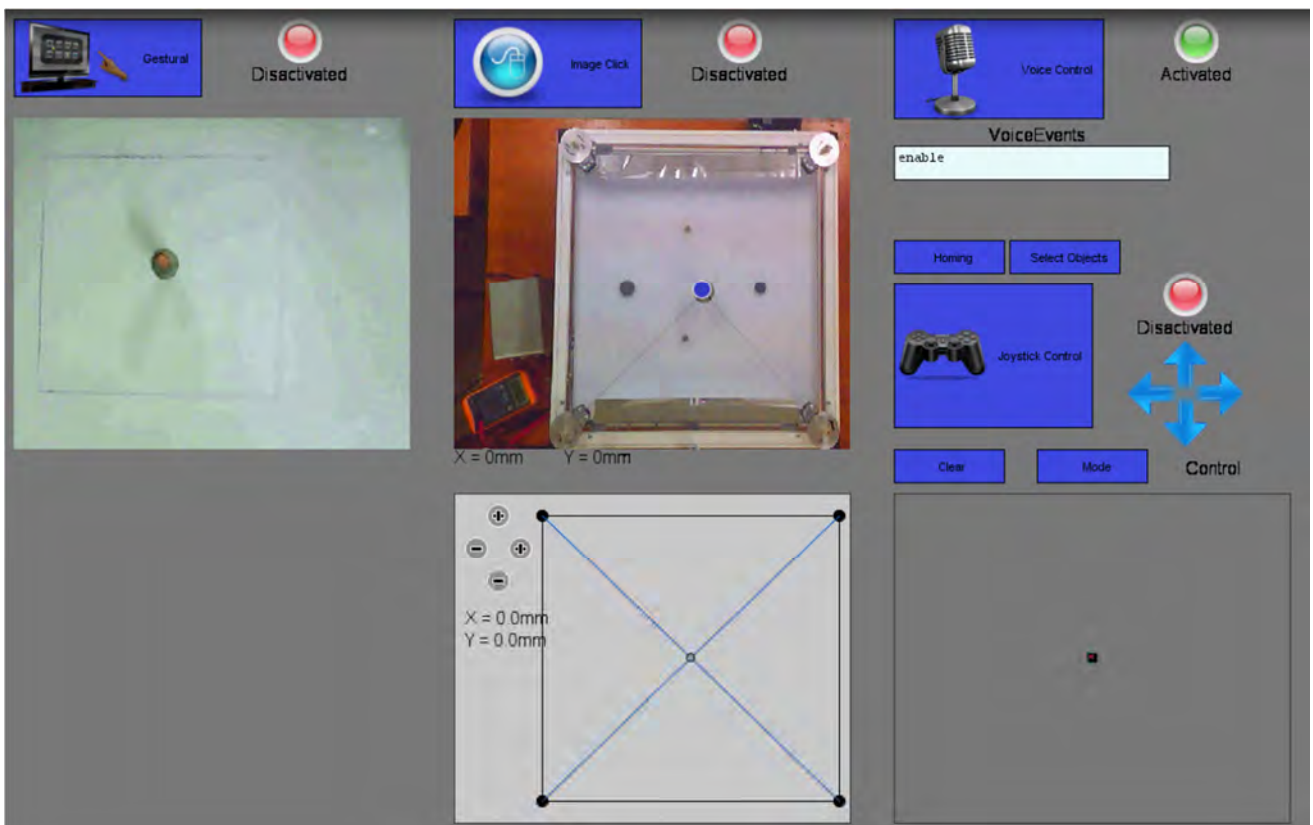


Figure 5.2- L'interface multimodale pendant la détection du mot « Enable »

5.2.2 L'analyse du mode de pointage sur image

Nous analysons dans cette section, le contrôle basé sur l'image qui est souvent nommé le mode « pointer et cliquer ». Ce mode consiste en pointant simplement et uniquement sur un objet apparaissant dans une image reçue depuis un site distant, et de convertir ce clic en une commande de robot vers l'emplacement correspondante dans l'espace réel.

Nous illustrons ci-dessous une expérience de pointage de deux objets en utilisant le système de pointage sur image. Les cibles visées sont des endroits ou des objets situés arbitrairement dans l'espace de travail. Le principe de fonctionnement est le suivant : l'opérateur désigne les cibles à partir de l'image affichée dans l'interface graphique en utilisant les clics de la souris. Ensuite, l'effecteur-final se déplace jusqu'à ce qu'il se superpose sur la cible souhaitée.

Dans la figure 5.3, l'opérateur pointe la cible désirée par la flèche de la souris. Après le clic, l'effecteur final se déplace vers la cible sélectionnée sur l'espace de travail réel (figure 5.4). Sur la zone graphique située dans la partie inférieure droite de l'interface multimodale, nous pouvons remarquer deux points bleus qui désignent le point de départ et le point de destination, le parcours entre les deux points est en rouge.

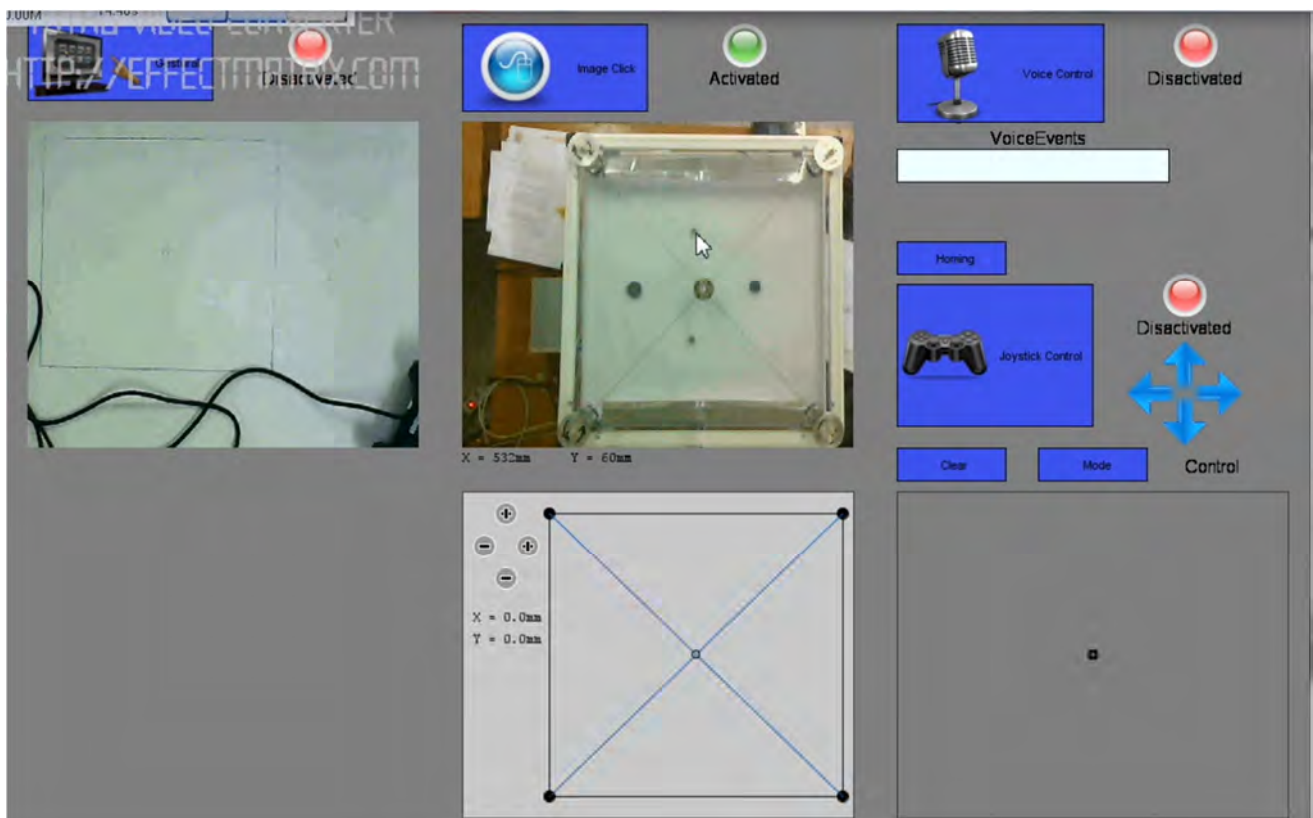


Figure 5.3- Objet sélection en utilisant le mode "point and click"

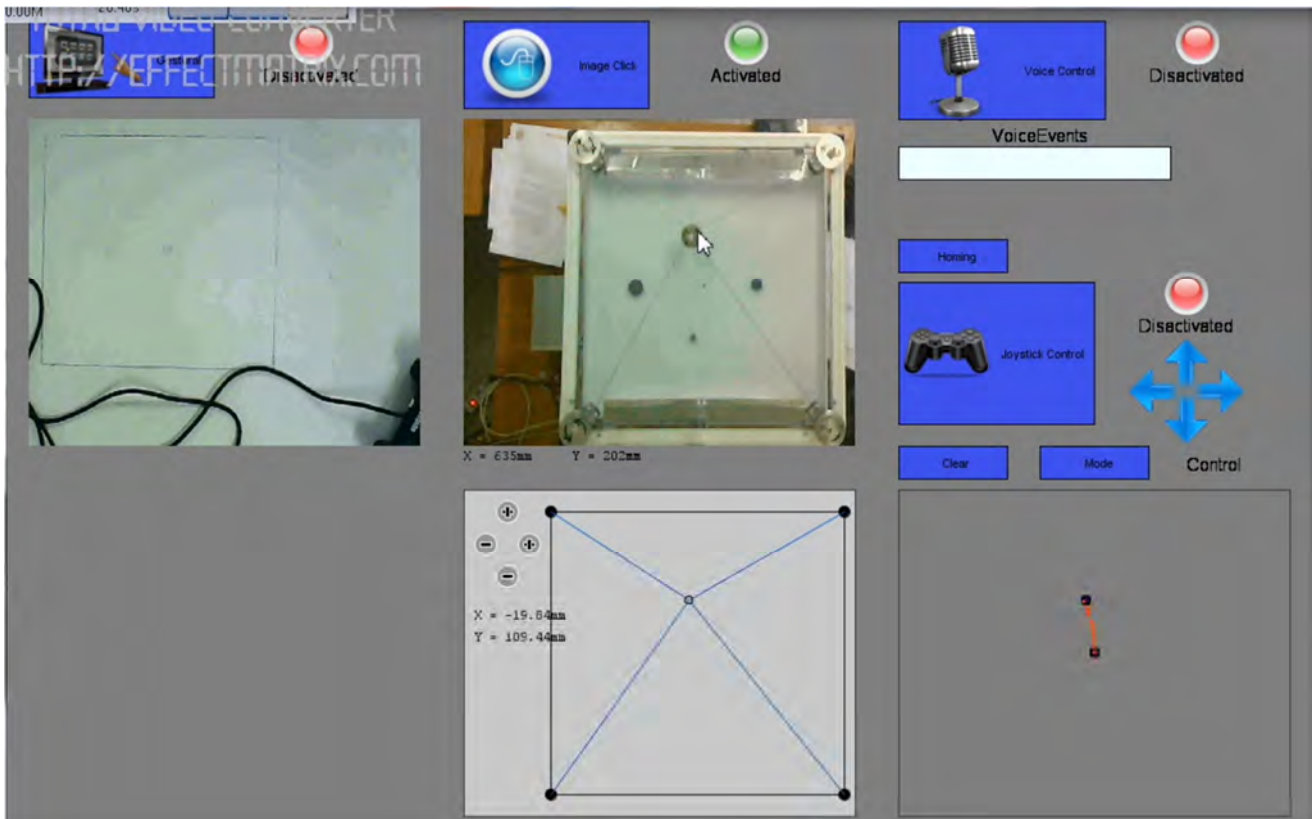


Figure 5.4-Le pointage d'un objet sélectionné

La figure 5.5 présente le pointage successive d'un deuxième objet sur l'espace de travail.

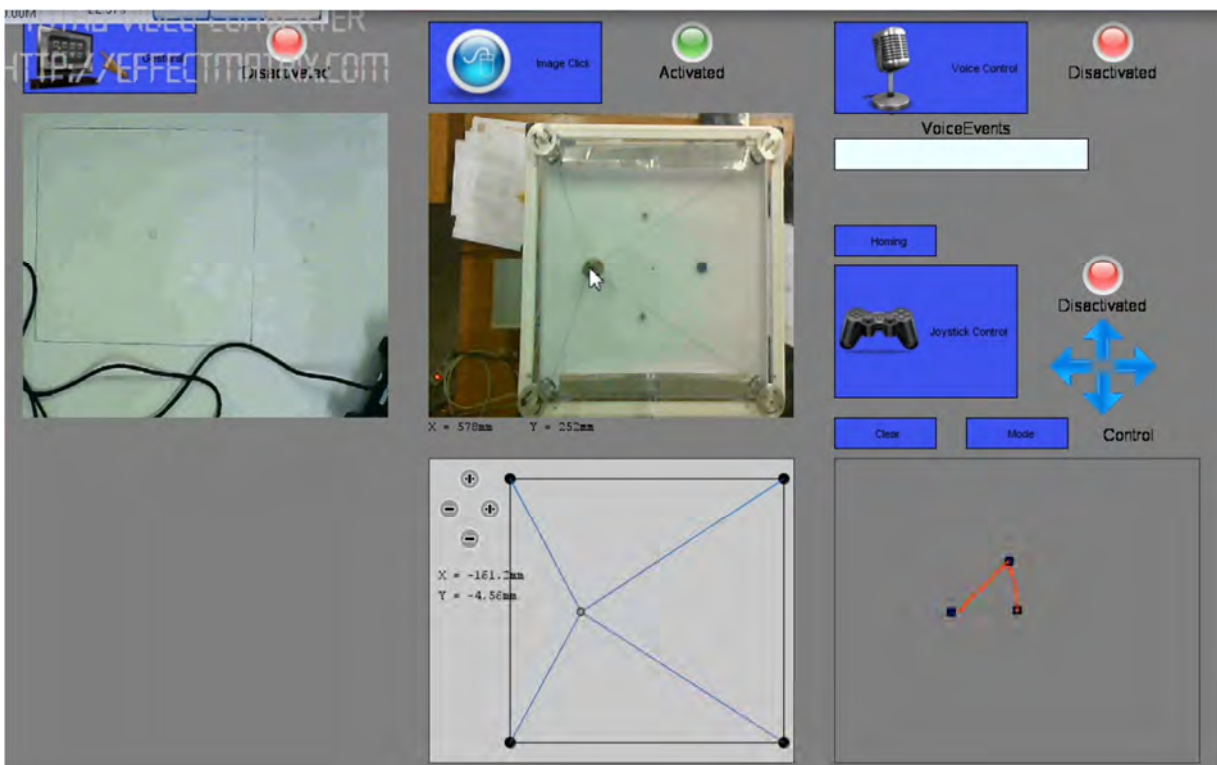


Figure 5.5-Le pointage successive d'un deuxième objet sur l'espace de travail

En se basant sur les expériences que nous avons faites avec de nombreuses personnes de différents âges et niveaux d'éducation. On peut remarquer que :

- Le mode de pointage sur image est avéré être le mode le plus approprié aux applications de la télé-robotique.
- Très simple à utiliser, et ne nécessite pas un temps pour l'apprentissage.
- Il n'y a aucune accumulation significative d'erreurs dans le cas du pointage de plusieurs cibles successivement.
- Très sensible au décalage du repère et de l'orientation de la camera par rapport au repère global.
- La précision du pointage est liée directement à la résolution de la camera et de la carte graphique, ainsi que la précision de la technique de calibration utilisée.

La force de ce mode de command est due de l'exploitation de la fonction de pointage qui est devenue une fonctionnalité standard dans la majorité des interfaces graphique d'aujourd'hui. Comme dans les interfaces des systèmes d'exploitation, nous pouvons remarquer l'existence de deux fonctionnalités dépendantes, la fonction de pointage et la fonction de glissement des objets en exploitant le « listener » des événements de la souris. Dont la première utilise l'événement « mouse click » et le deuxième utilise l'événement « mouse pressed ». Sur la base de ces possibilités, nous avons implémenté une extension de la technique de pointage sur image tout en exploitant l'événement « mouse pressed ». Dans ce cas, la technique de pointage sur image est peut être utilisée pour définir et dessiner des trajectoires continues. L'opérateur appuie sur le bouton de la souris et glisse la souris tout en décrivant la trajectoire désirée dans le site opérateur. Pendant ce temps, la position de la souris est continuellement envoyée en tant que points de consigne au contrôleur pour la conduite de l'organe terminal.

Ci-dessous, nous présentons deux expériences pour la réalisation de trajectoires continues en utilisant cette extension de la technique de pointage sur image. La Figure 5.6 présente la réalisation du nombre « 2 » en utilisant une configuration de pointage pour le site robot. Et la figure 5.7 présente la réalisation du nombre « 3 » en utilisant une configuration d'écriture pour le site robot. Les deux trajectoires affichées dans la zone de dessin (située dans la partie inférieure droite de l'interface) présentes respectivement : la trajectoire de consigne en bleu, et la trajectoire réalisée en rouge. Nous pouvons remarquer que la trajectoire réalisée est totalement superposée sur la trajectoire de consigne.

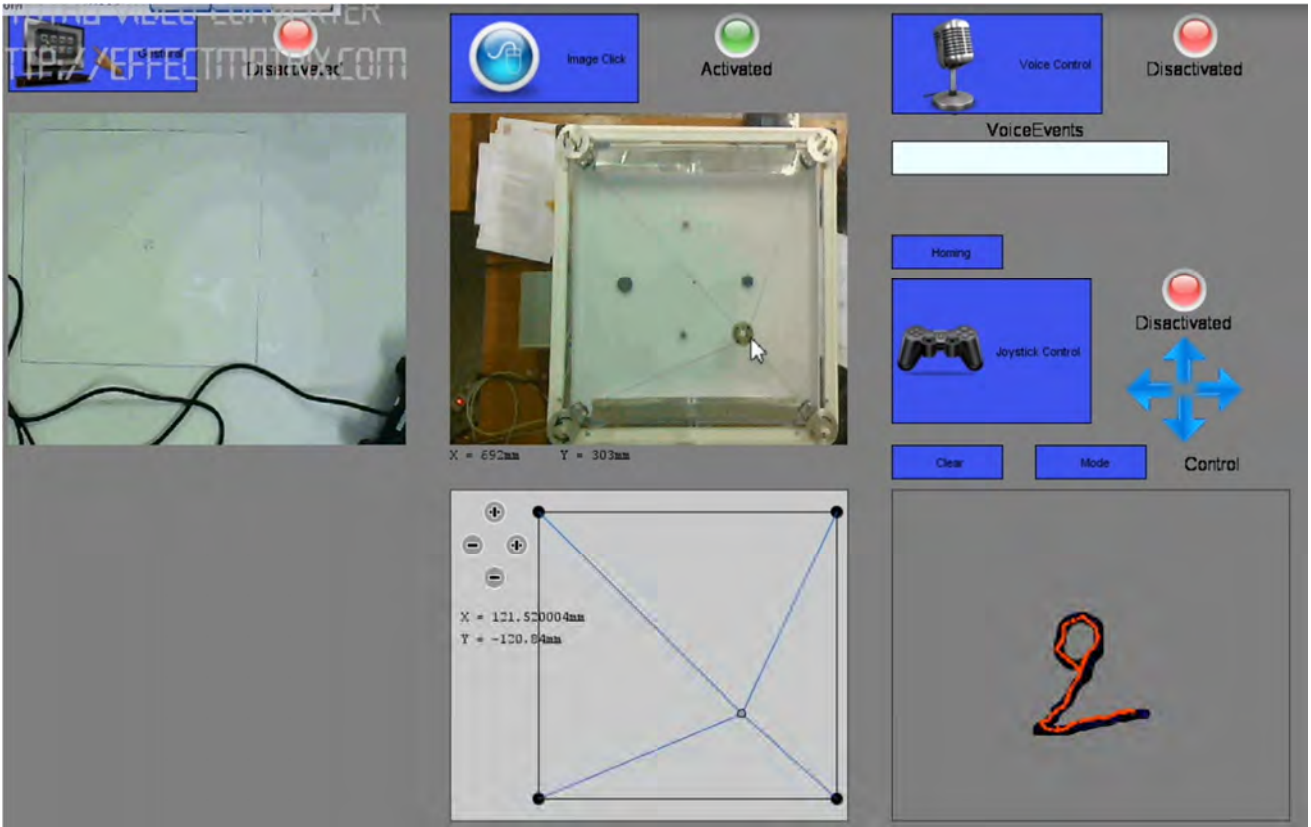


Figure 5.6-L'acquisition du nombre "2" en utilisant le mode "point and click"

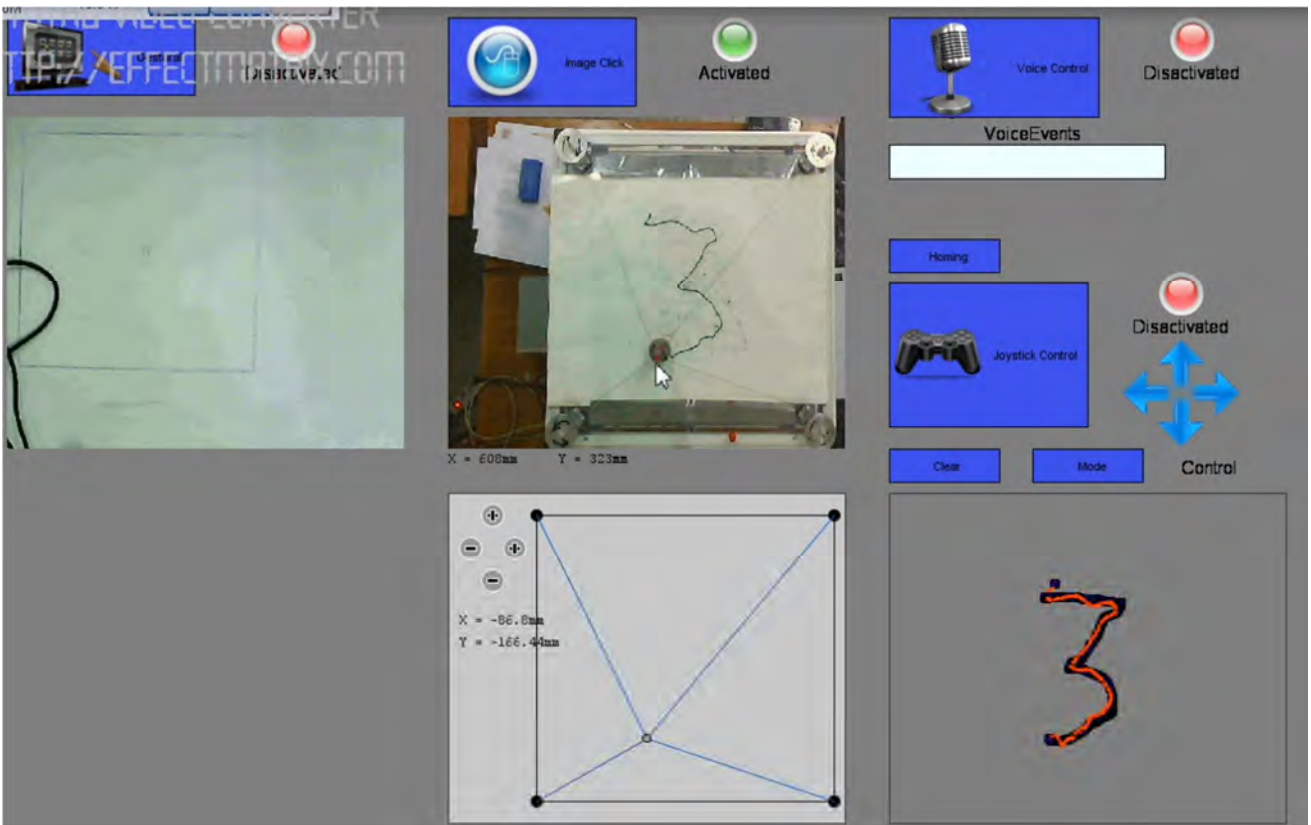


Figure 5.7-L'acquisition du nombre "3" en utilisant le mode "point and click"

5.2.3 L'analyse du mode gestuel

Dans cette section, nous présentons une analyse préliminaire du mode gestuel. Le système interactif est décomposé en deux parties principales. La première partie est le site opérateur et la seconde partie est le site robot. Dans le site de l'opérateur, l'opérateur via l'interface graphique d'utilisateur interagit avec le prototype expérimental au moyen des gestes spécifiques sous l'observation d'une caméra CCD. Les mouvements sont détectés et convertis en des commandes de robot ; Une autre caméra CCD dans le site robot permet à l'opérateur de connaître l'état actuel de l'espace du travail.

Rappelons que la technique mise en œuvre du mode gestuel, est basée sur la détection et le suivi des éléments colorés à partir d'images fournies par la caméra CCD du site opérateur. Et que l'algorithme se compose de deux phases d'exécution. La première partie d'exécution est le processus d'initialisation qui commence en sélectionnant l'objet coloré pour être suivi. Cela se fait au moyen d'un simple clic de la souris sur l'élément comme un doigt coloré de la main ou un stylo coloré. La deuxième phase d'exécution est le processus de suivi de trajectoire, qui est basé sur l'analyse des trames d'image. Chaque trame est entièrement analysée selon l'algorithme présenté dans le chapitre 3, afin de détecter l'élément coloré pour déterminer la direction du déplacement.

Dans la suite, nous allons présenter un exemple récapitulatif pour une expérience d'écriture en utilisant le mode gestuel. La figure 5.8 présente l'état initial de l'interface multimodale et du système interactif, dont nous pouvons remarquer sur l'image du site opérateur que l'opérateur a placé la flèche de la souris sur le stylo coloré pour le sélectionner. Nous pouvons noter que la position initiale du stylo coloré dans l'espace opérateur n'est pas forcément la correspondante de la position initiale de l'effecteur-final dans le site robot.

La figure 5.9, présente l'état de l'interface multimodale et du système après la sélection du stylo coloré par l'opérateur. Nous pouvons remarquer qu'un cercle porte la même couleur du stylo est affiché sur lui-même, pour indiquer que l'objet est en cours de poursuite. Et que l'effecteur final a déplacé de son état initial vers une position donnée comme destination par le clic de l'opérateur, et qui est réellement la superposition du point du stylo dans l'espace opérateur sur l'espace robot. Nous pouvons remarquer aussi sur l'espace robot et sur la zone de dessin (la zone inférieure à droite), le tracé d'une trajectoire entre les deux points. Et dans ce cas, la partie tracée entre le point initial et la position de départ de l'action gestuelle par l'effecteur finale n'appartient pas à la forme demandée par la commande gestuelle.

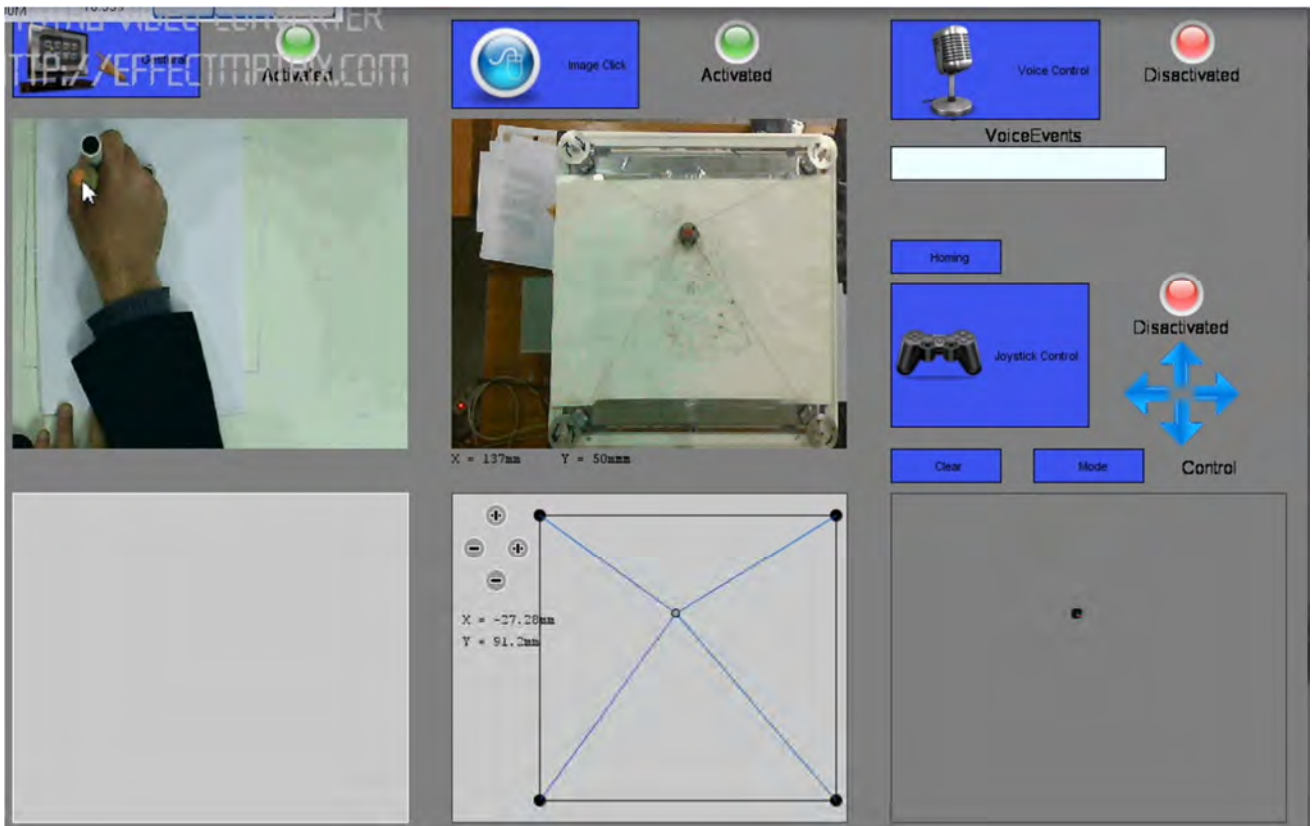


Figure 5.8-L'initialisation du mode gestuel

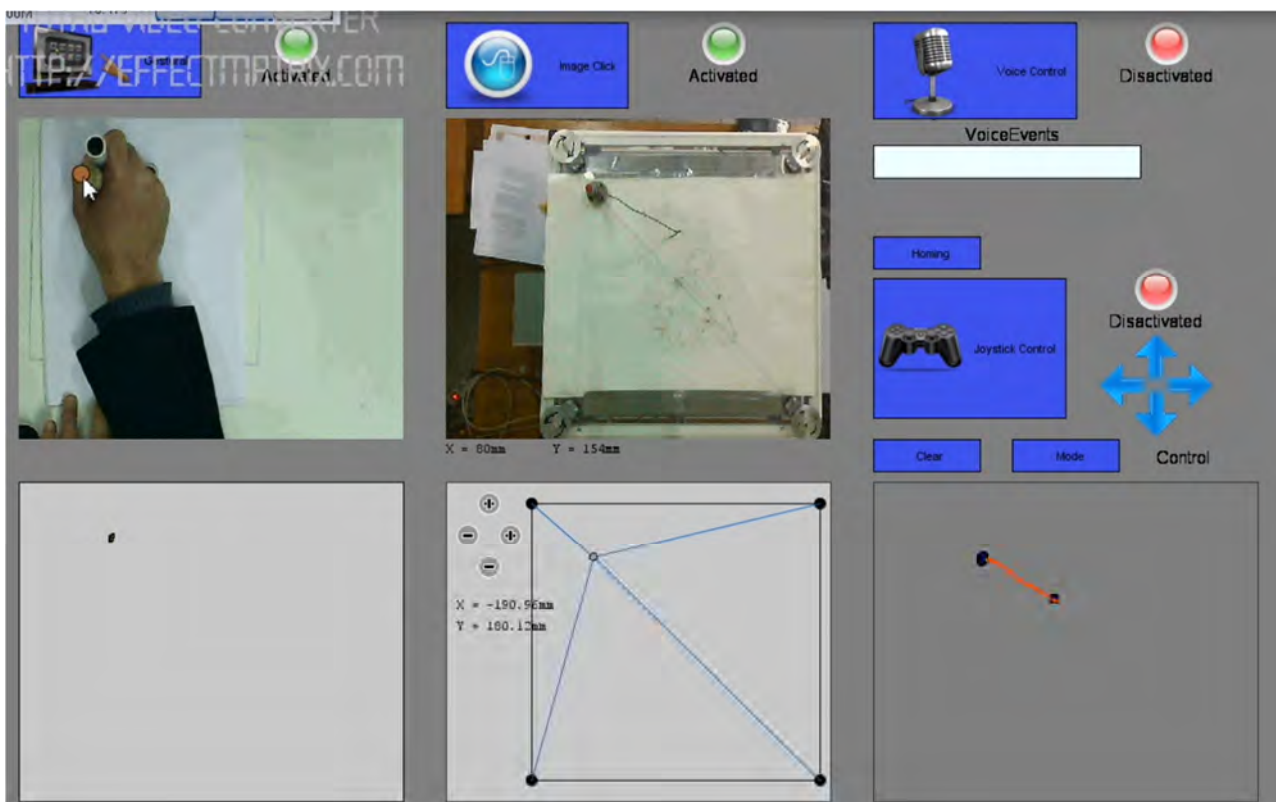


Figure 5.9- La sélection de l'objet coloré

La figure 5.10 présente l'état de l'interface multimodale et du système après l'écriture du nombre « 3 » en utilisant le mode gestuel. Nous pouvons remarquer sur la zone de dessin inférieur-gauche, l'affichage des points de déplacement de l'objet coloré, après la conversion en utilisant la matrice de la projection de la caméra CCD du site robot entre l'espace image et l'espace de travail opérateur. Nous pouvons remarquer une différence entre le geste demandé et le geste détecté, à cause de plusieurs paramètres qui influent sur la précision de la détection comme :

- La vitesse de déplacement de l'objet coloré.
- La fréquence de balayage pour l'affichage des images sur l'interface multimodale.
- Le niveau de luminosité du site opérateur.
- L'orientation de l'objet coloré par rapport au plan horizontal de l'espace opérateur.
- L'existence d'autres objets ont une nuance de couleur proche à la couleur surveillée qui résulte des points de fausses détections.
- La discontinuité des points détectés de l'objet coloré influe sur la finesse de la trajectoire.

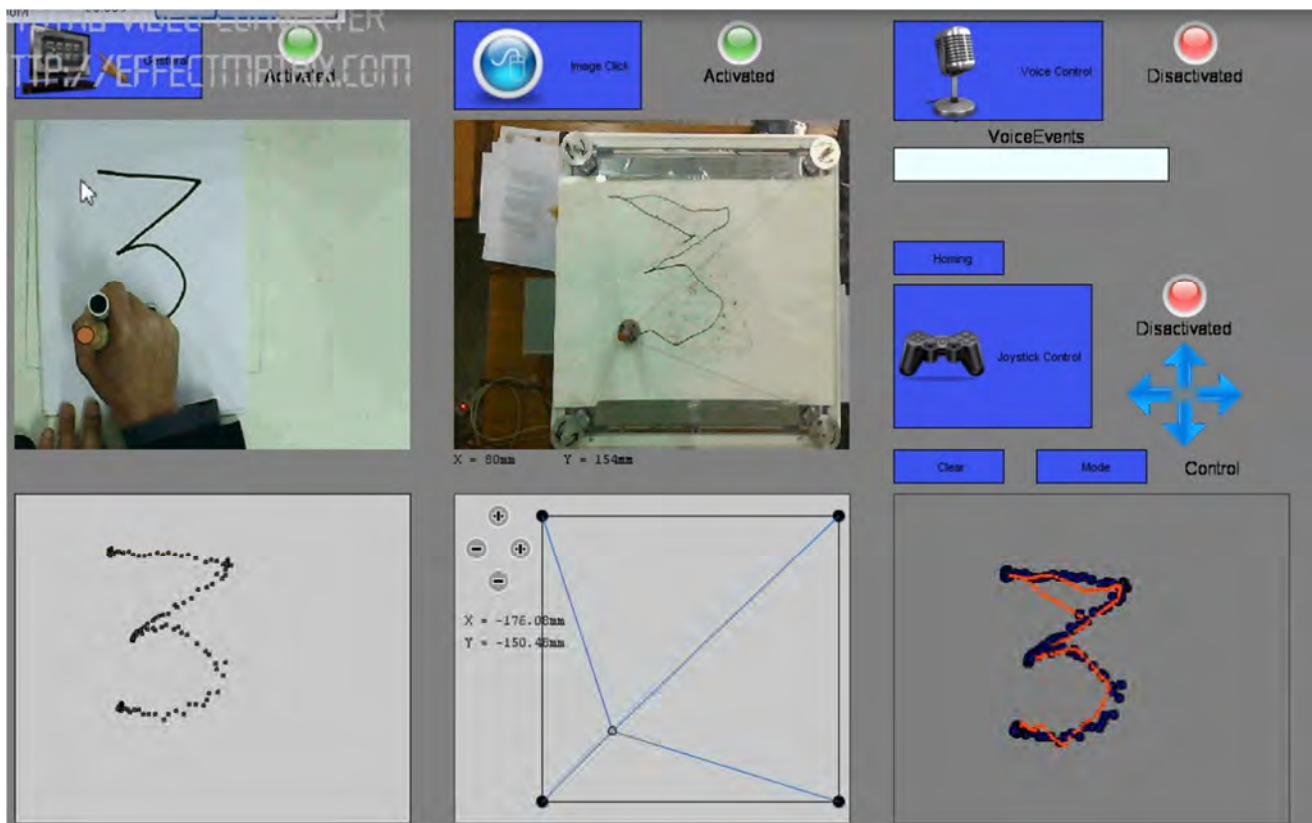


Figure 5.10-L'acquisition du nombre "3" en utilisant le mode gestuel

Les deux figures suivantes (figure 5.11, figure 5.12), présentes respectivement la réalisation de deux types de formes, la première une forme ouverte « ζ » et la deuxième une forme fermée « B ».

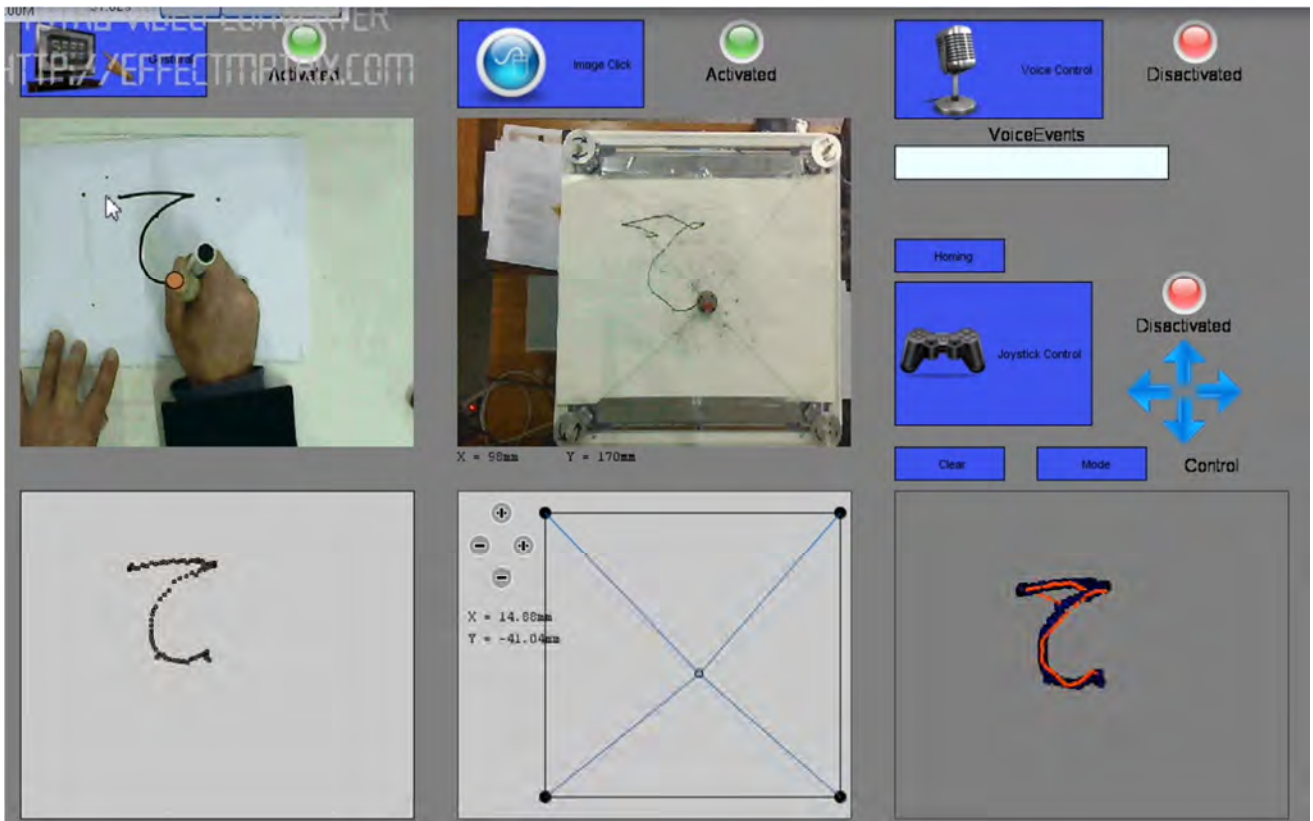


Figure 5.11-L'acquisition de la lettre "z" en utilisant le mode gestuel

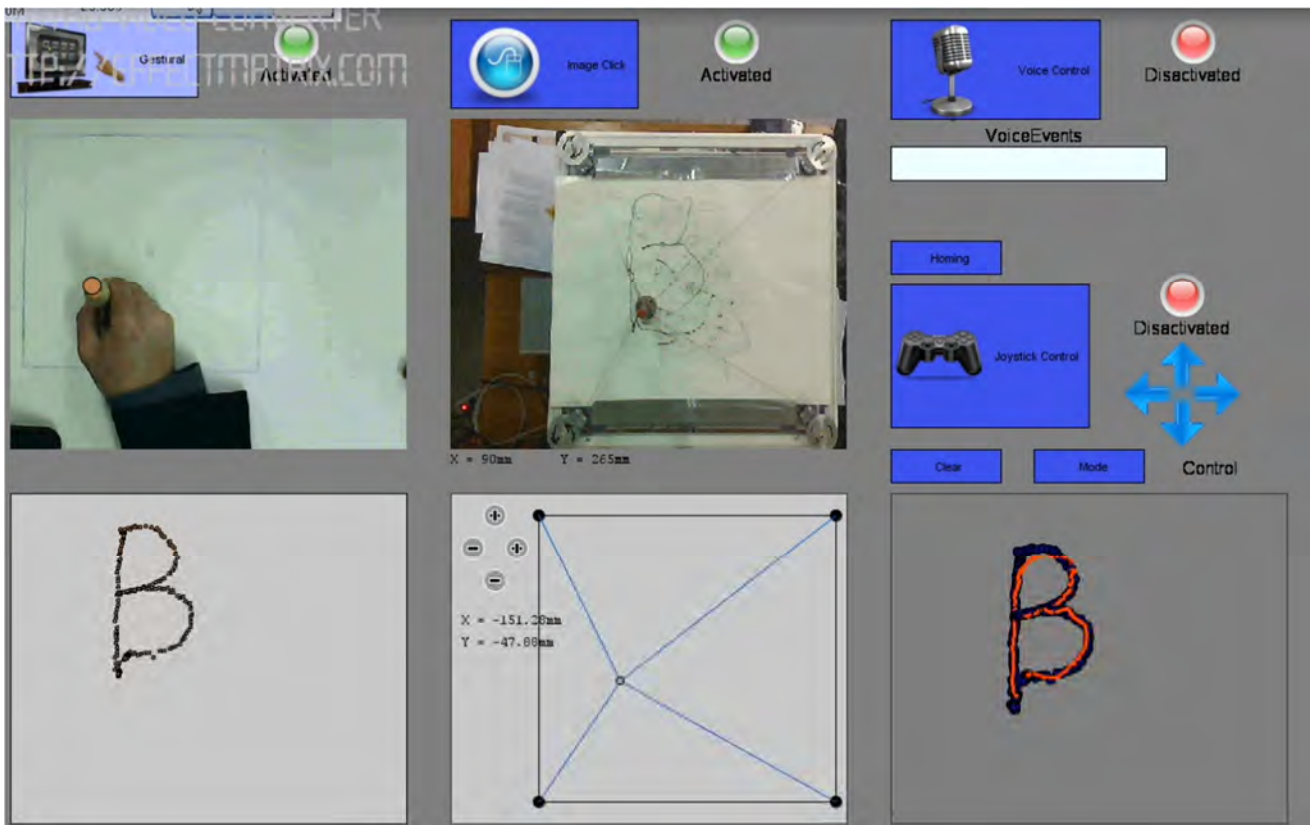


Figure 5.12-L'acquisition de la lettre "B" en utilisant le mode gestuel

Sur la base des différentes expériences réalisées avec le mode gestuel en utilisant le robot parallèle à câbles. Nous pouvons diviser ce mode de command en deux sous mode, le mode gestuel avec contact avec l'environnement et le mode gestuel sans contact avec l'environnement. Les résultats présentés précédemment représentent l'utilisation du mode gestuel avec contact. Dont nous avons remarqué que :

- Le mode gestuel avec contact est le mieux adapté pour les objets à une seule surface colorée.
- Nécessaire pour les applications qui exigent un déplacement contrôlé ou personnalisé comme par exemple : l'écriture, le déplacement d'un objet à partir d'un point vers un autre, ...etc.
- Impose des contraintes sur la procédure d'initialisation telle que le choix de l'objet coloré et de leur position initiale dans l'espace opérateur.
- La concentration de l'opérateur pendant l'acquisition du geste, comme le contrôle de la vitesse de déplacement ainsi que l'orientation, augmente la charge mentale (mental workload).

Dans le cas du mode sans contact, nous avons remarqué que l'absence du contact avec le plan de guidage provoque un mouvement à 3 dimensions. Comme nous avons utilisé une technique de détection monoscopique, le système de vision détecte seulement la projection du mouvement sur le plan, qui résulte dans ce cas une perte d'information concernant le mouvement gestuelle. Nous avons constaté que :

- Les mouvements gestuels sans contact peuvent être effectués avec une vitesse relativement élevée par rapport au mode gestuel avec contact.
- Une initialisation simple par rapport au mode avec contact.
- Minimise la charge mentale de l'opérateur (mental workload) tout en minimisant leur concentration pendant l'acquisition du geste.

Pour cela, nous avons proposé dans notre travail d'utiliser le mode gestuel sans contact pour coder des tâches particulières plus au moins complexes en utilisant un geste unique. La Figure 5.13 présente l'acquisition d'un geste sans contact qui engage le robot pour effacer l'espace de travail.

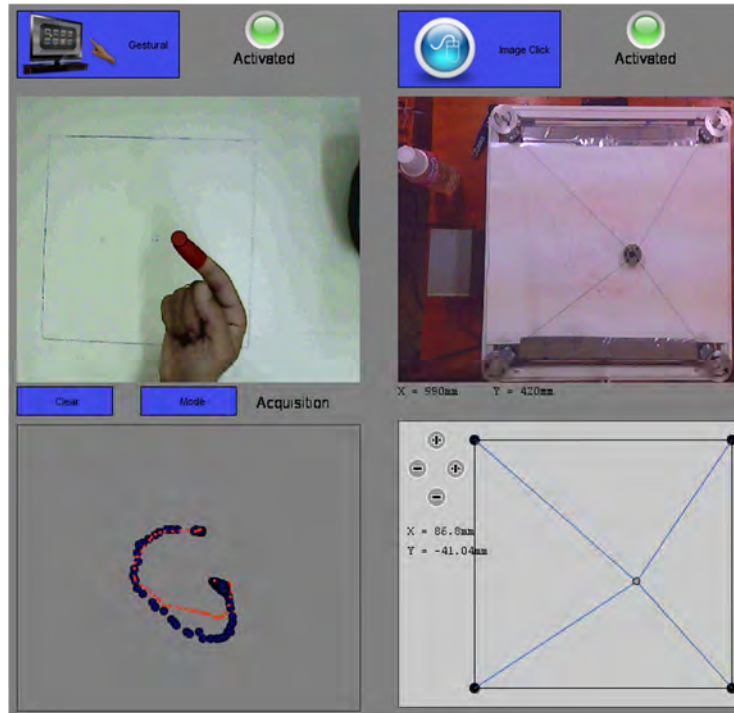


Figure 5.13- L'acquisition d'une commande gestuelle sans contact avec l'environnement

Nous pouvons conclure, sur la base de plusieurs expériences effectuées en utilisant les deux extensions du mode gestuel (le mode gestuel avec contact et le mode gestuel sans contact avec l'environnement); réalisées par des opérateurs avec différents âges et niveaux scientifique. Que dans le cas du mode gestuel avec contact avec l'environnement, la plupart de la charge mentale « mental workload » du travail est affectée sur l'opérateur ; Par contre dans le cas du mode gestuel sans contact avec l'environnement, cette dernière est subie par le système lui-même à cause du codage des tâches complexes par des empreintes gestuelles.

5.2.4 L'analyse de mode de contrôle par joystick

L'implémentation de la commande manuelle par joystick, donne à l'opérateur un accès direct et plus sûre pour commander le prototype expérimental par rapport aux autres techniques de contrôle haut niveau. En effet, au cours de l'utilisation de ce mode de commande la configuration courante du joystick actionnée par l'opérateur est transmise périodiquement et en continue vers l'interface multimodale. Cette interface convertit ces signaux pour élaborer les commandes en position et de l'orientation de l'effecteur final du robot à câbles.

Pour des raisons de sécurité et d'efficacité d'utilisation, nous avons implémenté deux extensions du mode de commande manuel par joystick. Le premier est nommé le mode pas à pas (figure 5.14). Dans ce mode le joystick est utilisé en mode numérique avec un pas de déplacement réglable selon le besoin. Généralement, nous utilisons ce mode avec un pas de déplacement très petit pour pouvoir opérer dans des endroits qui nécessitent plus de précision et assure plus de confiance pour

l'opérateur ; comme par exemple ce mode joystick ce combine très avantageusement avec la plupart des autres modes lorsque nous avons besoin d'une manipulation nécessitant de la dextérité. Dans la plupart des applications industrielles de précision, la combinaison du mode joystick avec le mode de pointage sur image est très intéressante. Dans ce cas, l'approche globale est effectuée par le mode clic sur image pour rapprocher l'end-effecteur vers l'endroit où l'objet à manipuler. Près de cet endroit, on utilise le mode joystick pour effectuer l'opération de manipulation qui nécessite une certaine précision.

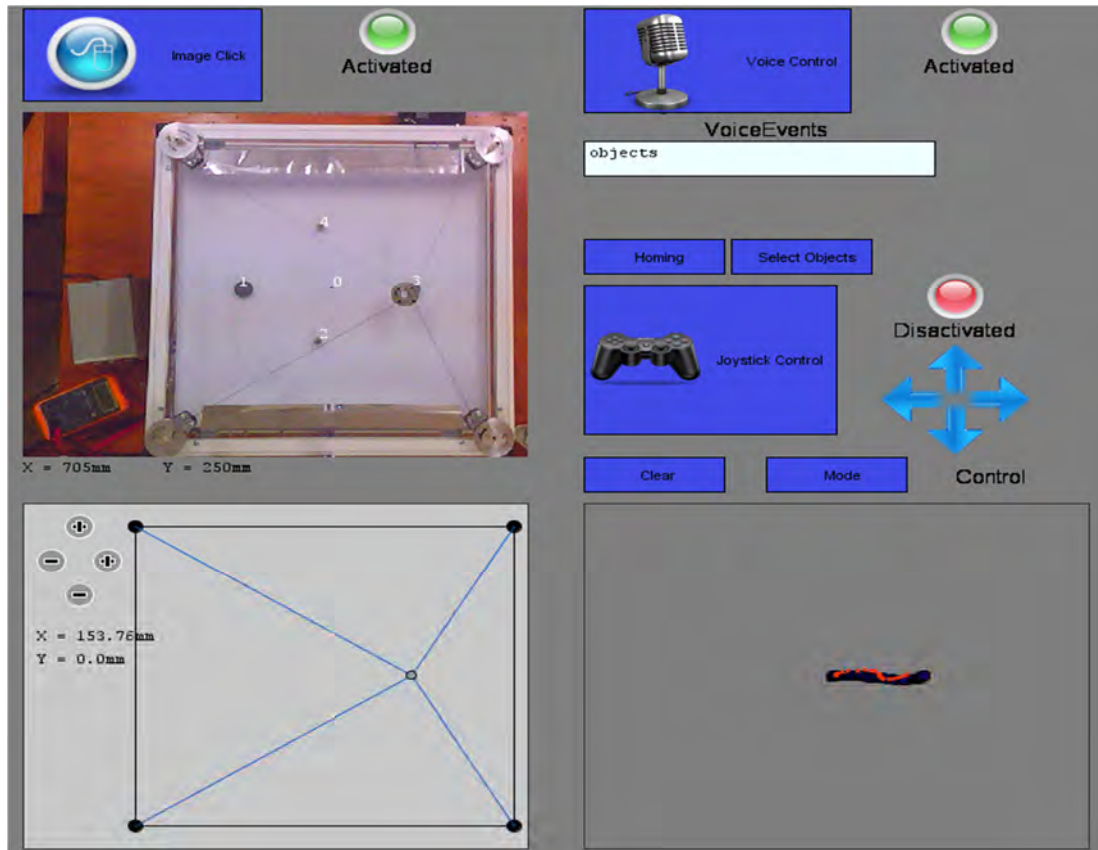


Figure 5.14- La commande par joystick en mode pas à pas

La deuxième extension de ce mode de commande est nommée mode continue (figure 5.15). Dans ce cas, le joystick est utilisé en mode analogique. Les valeurs des différents axes vont de -1 à 1 avec une résolution donnée. Pour pouvoir couvrir l'ensemble de l'espace de travail du robot en utilisant ce mode étendu, nous avons transformé l'intervalle de -1 a +1 en délimitant les bornes de l'espace de travail selon les axes x et y respectivement par les intervalles $[X_{min}, X_{max}]$ et $[Y_{min}, Y_{max}]$.

Cette extension du mode donne à l'opérateur la possibilité d'interrompre l'exécution d'une tâche donnée comme mode complémentaire. A l'aide de ce mode l'opérateur peut intervenir lorsqu'il y'a une défaillance ou d'un blocage d'un autre mode de commandes haut niveau (gestuel, voix, ..., etc). Cela permis de sortir le système des situations critiques.

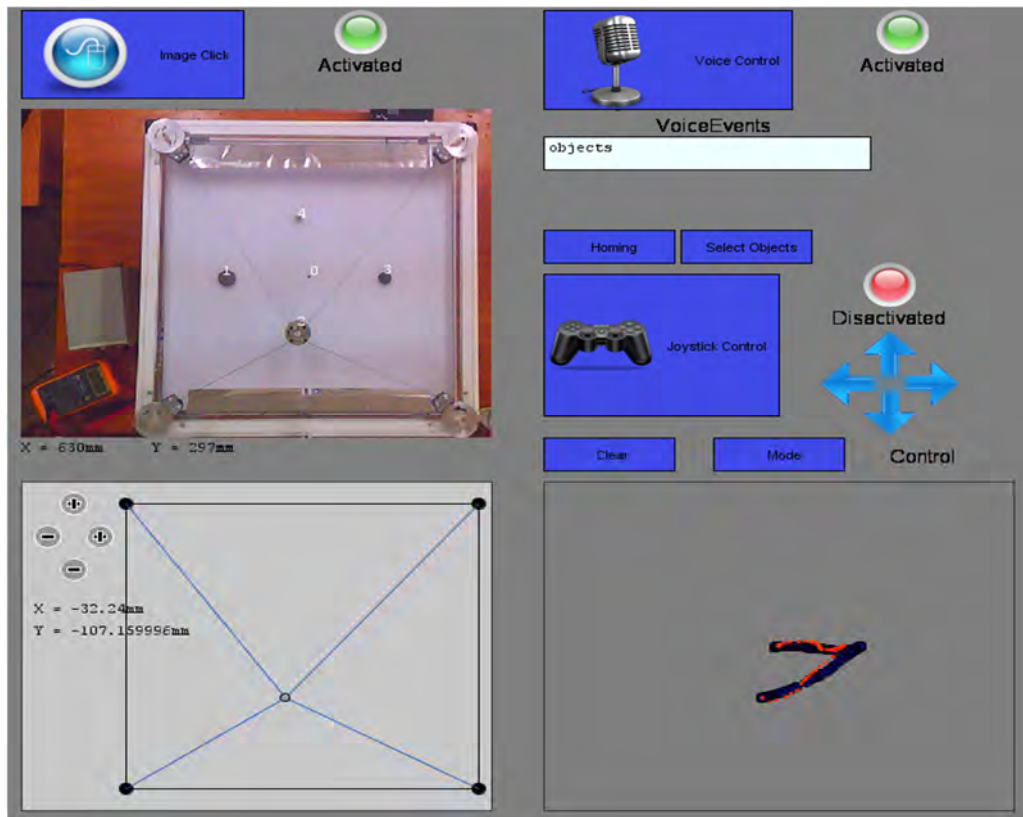


Figure 5.15- La commande par joystick en mode continue

5.3 L'analyse de combinaisons des modes

5.3.1 La combinaison du mode vocal avec le mode de pointage sur image

Dans cette section, un scénario de combinaison entre le mode vocal et le mode de pointage sur image est présenté. La figure 5.16 présente l'état initiale de l'interface multimodale, dont nous pouvons remarquer que l'effecteur final est au point de référence (0,0). Et que cinq objets sont placés arbitrairement sur l'espace de travail. Nous avons décomposé le scénario de combinaison en deux phases d'exécution, la phase d'initialisation et la phase d'utilisation. Pendant la phase d'initialisation, l'opérateur active les deux modes en question vocalement. Ensuite, il clique sur le bouton « Select Objects » pour mettre le système en état d'attente d'enregistrement des objets de la banque de pointage. Dans cette étape, l'opérateur utilise le mode de pointage sur image pour ajouter les objets à la banque de données ; En cliquant sur leurs images, les positions de ces objets sont automatiquement enregistrées.

Pendant la deuxième phase, nous pouvons activer une option de convention avec le système interactif par la commande vocale « Find » (figure 5.16), suivi par la commande « Objects » ; Dans ce cas, un numéro est affecté et affiché sur chaque objet placé sur l'espace du travail (Figure 5.17).

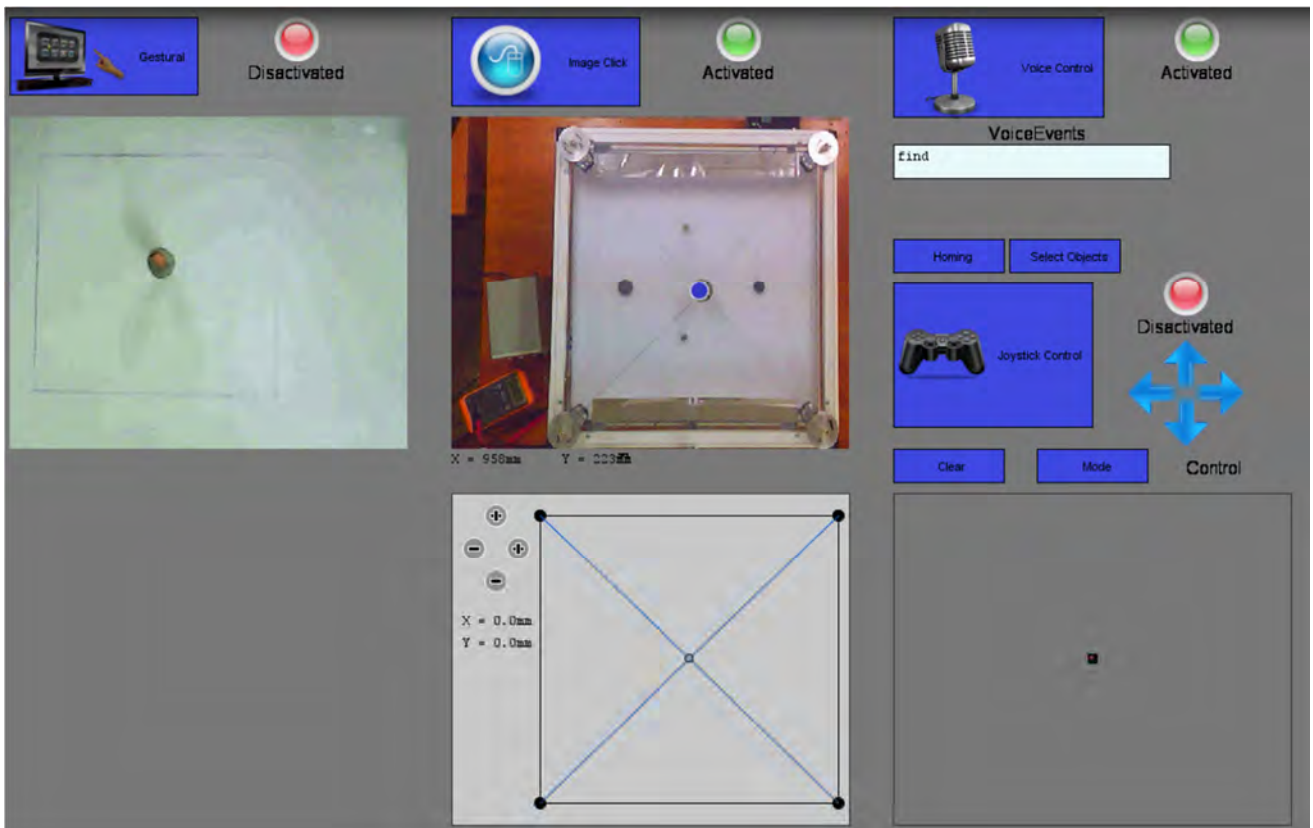


Figure 5.16- La détection de la commande vocale "find"

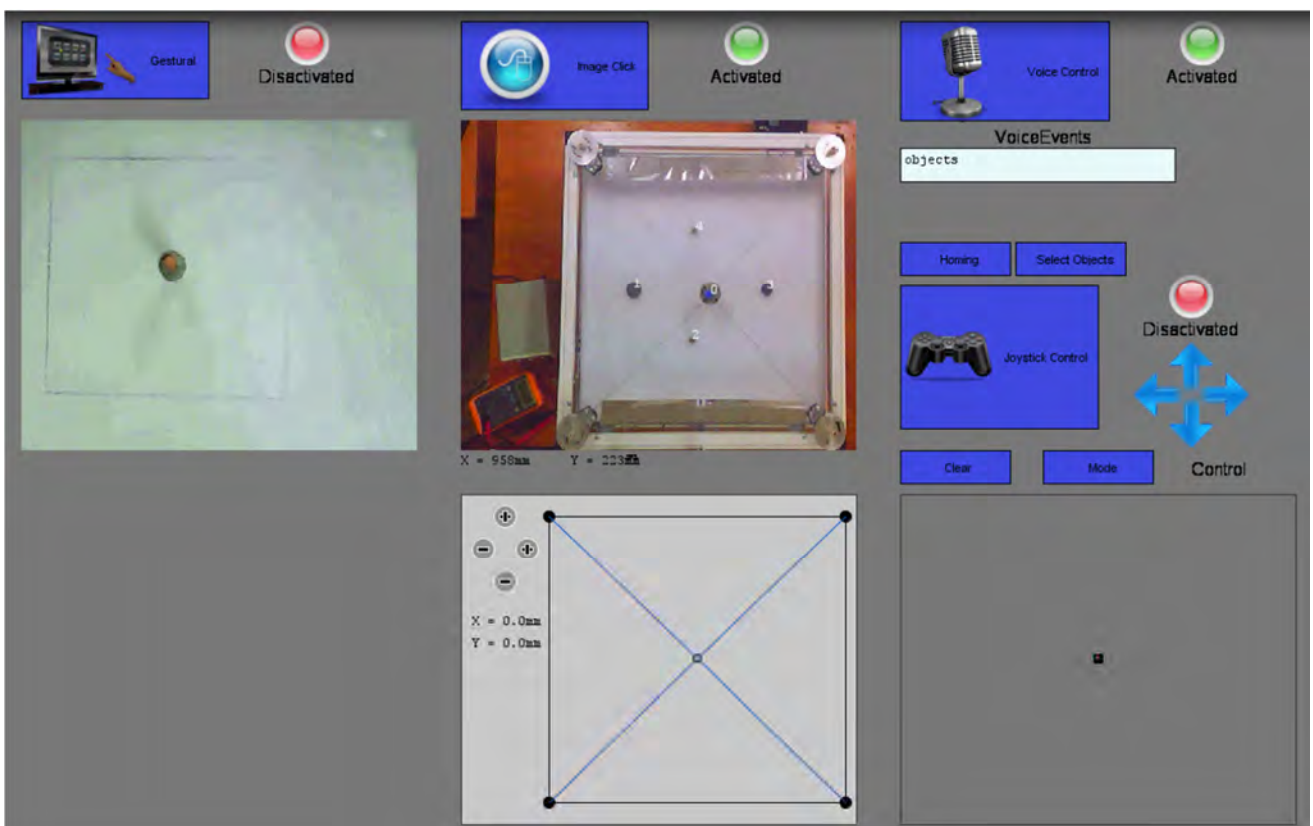


Figure 5.17- L'affichage de la banque d'objets

Nous pouvons noter que, sur la figure 5.17, le choix d'emplacement d'affichage de la numérotation des objets et de la couleur des numéros est fait de telle façon de ne pas perturber l'affichage de l'image de l'espace de travail.

L'opérateur peut utiliser une simple commande vocale « goto number » pour engager le robot de pointer l'objet sélectionné. La figure 5.18 et la figure 5.19 présentent un exemple d'expérience en utilisant le scénario de combinaison pour le pointage de l'objet numéro 1 « one ».

Nous pouvons conclure que la combinaison des modes de commande haut niveau peut augmenter la qualité et les performances de l'interaction homme-robot ; tout en facilitant la façon de générer des commandes nécessaires pour l'exécution des tâches complexes. Dont nous pouvons remarquer qu'il existe deux types d'opérateur en termes de « humain likes » comme facteur humain, le premier type d'opérateurs préfère d'utiliser les systèmes interactifs sans contact avec les instruments (**des opérateurs qui cherchent de se libèrent de tout relation instrumentale directe**), et le deuxième type aime l'utilisation des instruments sophistiqués. Donc nous pouvons dire qu'il existe un facteur humain concernant le choix du mode ou la combinaison et donc la facilité d'utilisation, qui influe sur l'interaction.

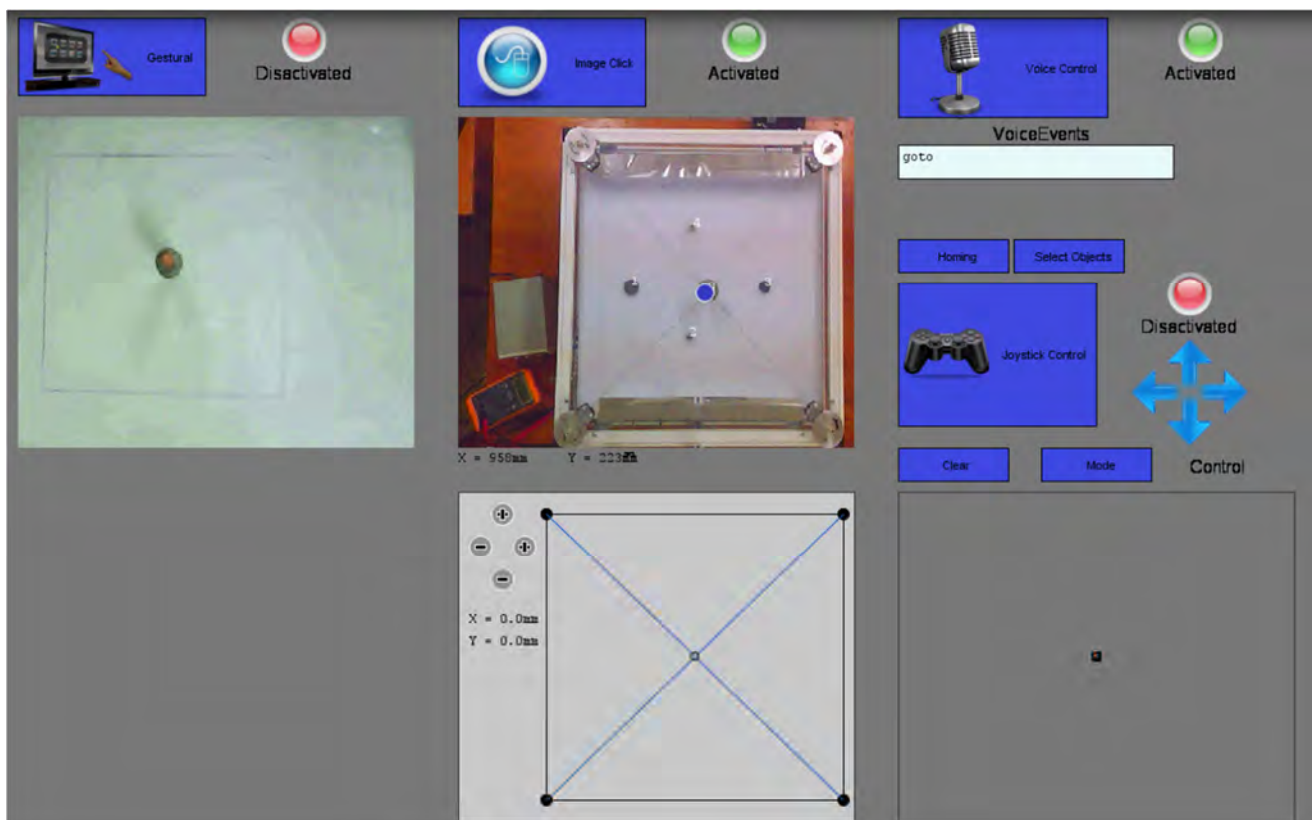


Figure 5.18- La détection de la commande "goto"

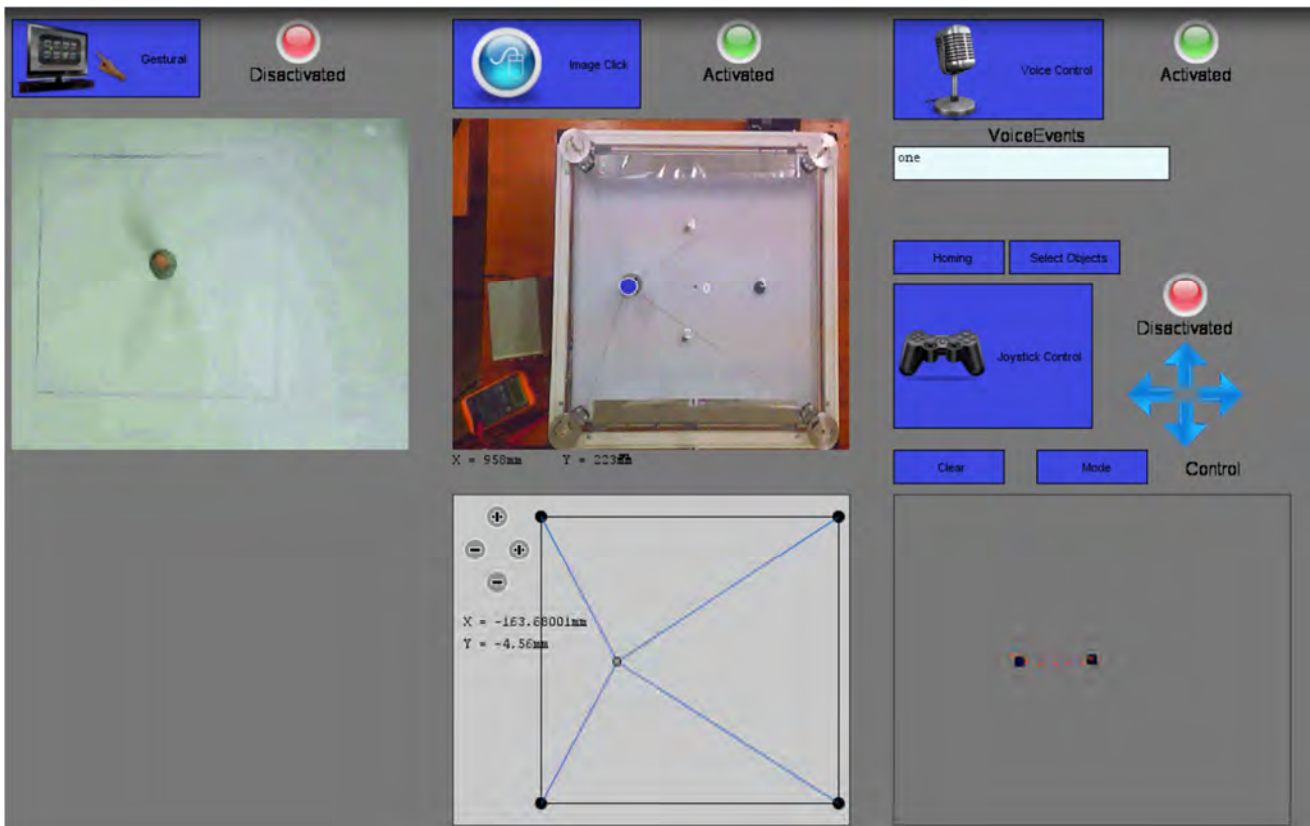


Figure 5.19- Le pointage d'un objet en utilisant la commande vocale

5.3.2 La combinaison du mode gestuel avec le mode basé sur image

Nous avons proposé dans ce scénario de combinaison, l'utilisation du robot à 4 câbles dans une classe spéciale pour un centre de réhabilitation en exploitant plusieurs avantages pour ce type de robots: un grand espace de travail en comparaison avec les manipulateurs parallèles classiques; En plus, les robots à base de câbles ont une structure légère qui a de bonnes propriétés dynamiques; un rapport masse/ charge utile élevée et un coûts construction réduit; une bonne portabilité; une structure facile à monter, démonter, stocker, et les transports. Ainsi que l'exploitant de plusieurs avantages de la multimodalité de notre système interactif.

Notre scénario de l'expérience consiste de la mise en œuvre d'une salle spéciale dédiée aux applications de la réhabilitation. Dans cette salle, un robot à 4 câbles est associé à chaque bureau. Le contrôle de haut niveau fourni par l'interface multimodale, permet à l'enseignant de fournir différentes opérations d'enseignement. Dans ce contexte; nous pouvons diviser le processus d'enseignement en deux opérations principales, dans la première opération, le contrôle gestuel sert à entraîner le doigt d'un étudiant handicapé pour suivre les actions définies par l'enseignant sur le tableau (phase d'enseignement par démonstration), dans la deuxième opération, l'étudiant handicapé tente d'accomplir le mouvement ou le dessin comme il a été enseigné par le professeur plusieurs fois dans la phase de l'enseignement par démonstration. Dans le cas d'un point de blocage à un moment donné dans leur mouvement ou leur dessin, l'enseignant peut prendre le contrôle pour guider ou aider

l'étudiant handicapé à accomplir la tâche (phase d'enseignement par orientation et assistance). Dans ce cas, l'enseignant doit être en mesure d'intervenir à tout moment et en temps réel dans cet endroit de blocage pour l'assistance.

- **Centre de réadaptation**

Dans notre laboratoire, nous avons mis en place une classe expérimentale. ; Un robot à 4 câbles est associé à chaque bureau (figure 5.20). Elle sert à entraîner le doigt d'un étudiant handicapé à la suite des actions définies par l'enseignant sur le tableau (figure 5.21).

L'enseignant interagit avec cet environnement de la classe expérimentale au moyen de l'interface multimodale. Cette interface graphique permet la combinaison du mode gestuel et le mode basé sur l'image afin d'assurer l'efficacité de l'enseignement.



Figure 5.20- La classe spéciale proposée



Figure 5.21-Une expérience pratique

Dans la phase de l'enseignement par démonstration, l'enseignant guide l'élève handicapé pour écrire en utilisant le mode gestuel. L'enseignant sélectionne le stylo de couleur qui sera suivi par le système de vision, puis écrit le mot souhaité, l'effecteur final en temps réel conduit la main des élèves handicapés de suivre la trajectoire souhaitée (figure 5.22).

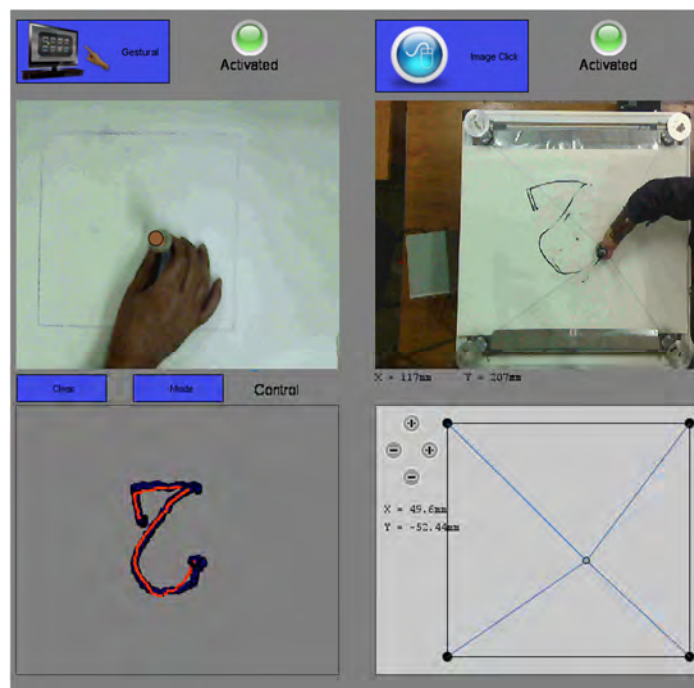


Figure 5.22-La phase de l'enseignement par démonstration

Dans la phase de guidage et d'assistance, l'enseignant peut prendre le contrôle pour aider les personnes handicapées à accomplir la tâche en cas de blocage. Dans cette situation, comme il n'est

pas possible en utilisant le contrôle gestuel, donc; nous devons passer à la commande basée sur l'image pour résoudre ce problème. Dans ce cas, l'enseignant conduit la main de l'élève à l'aide d'un simple bouton de la souris enfoncé sur l'image de la main de l'élève dans le point de blocage réel, afin de le faire sortir de cette situation de blocage pour compléter la tâche (figure 5.23).

Les figures 5.23 et 5.24, présentent l'intervention de l'enseignant en utilisant le mode de commande par pointage par image ; pour guider l'opérateur de sortir des situations de blocages. Cette application nous permis d'analyser la dépendance de notre système interactif par rapport au **degré de complexité de l'environnement du travail**. Dont nous avons remarqué que la multimodalité permis de résoudre plusieurs problèmes dans des environnements complexes.

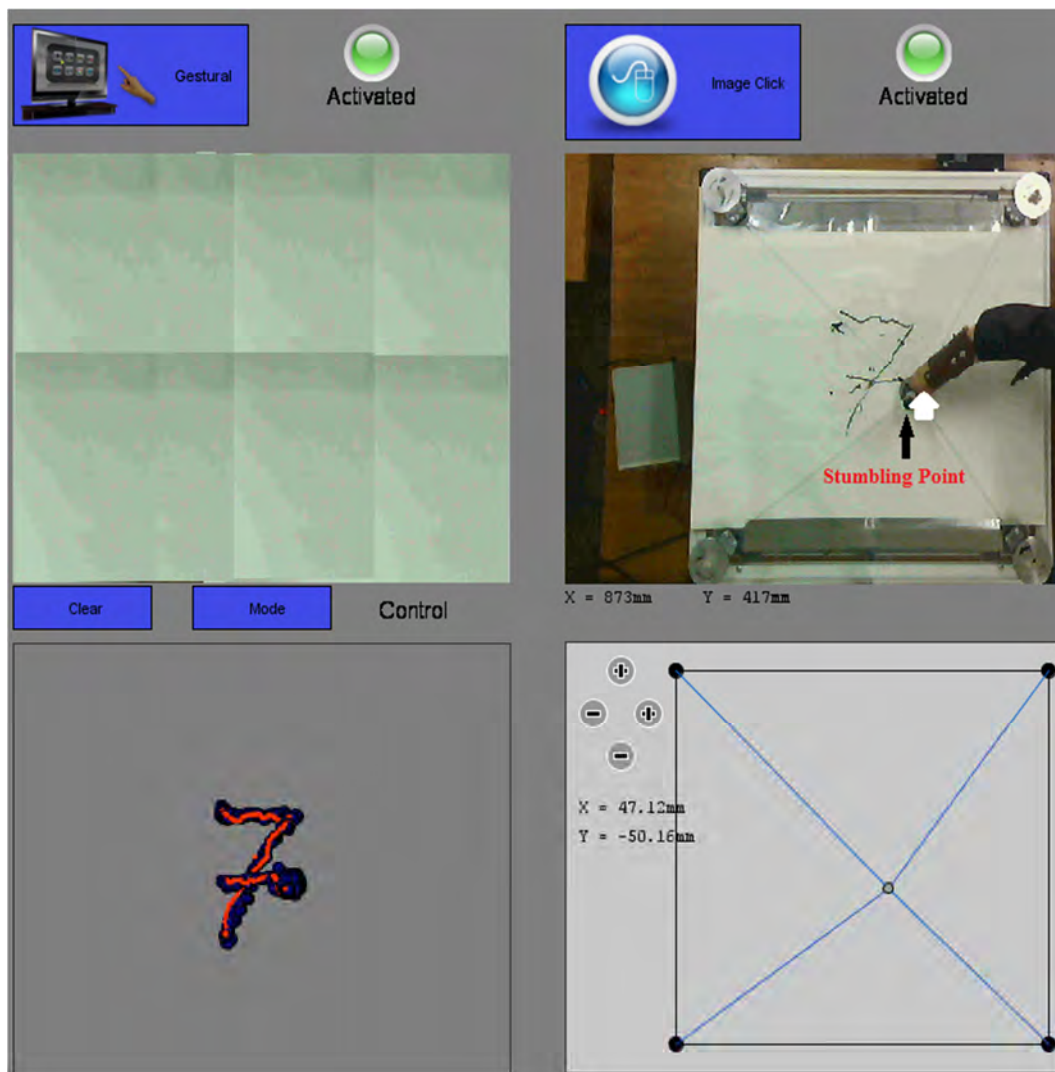


Figure 5.23- Intervention dans un point de blocage

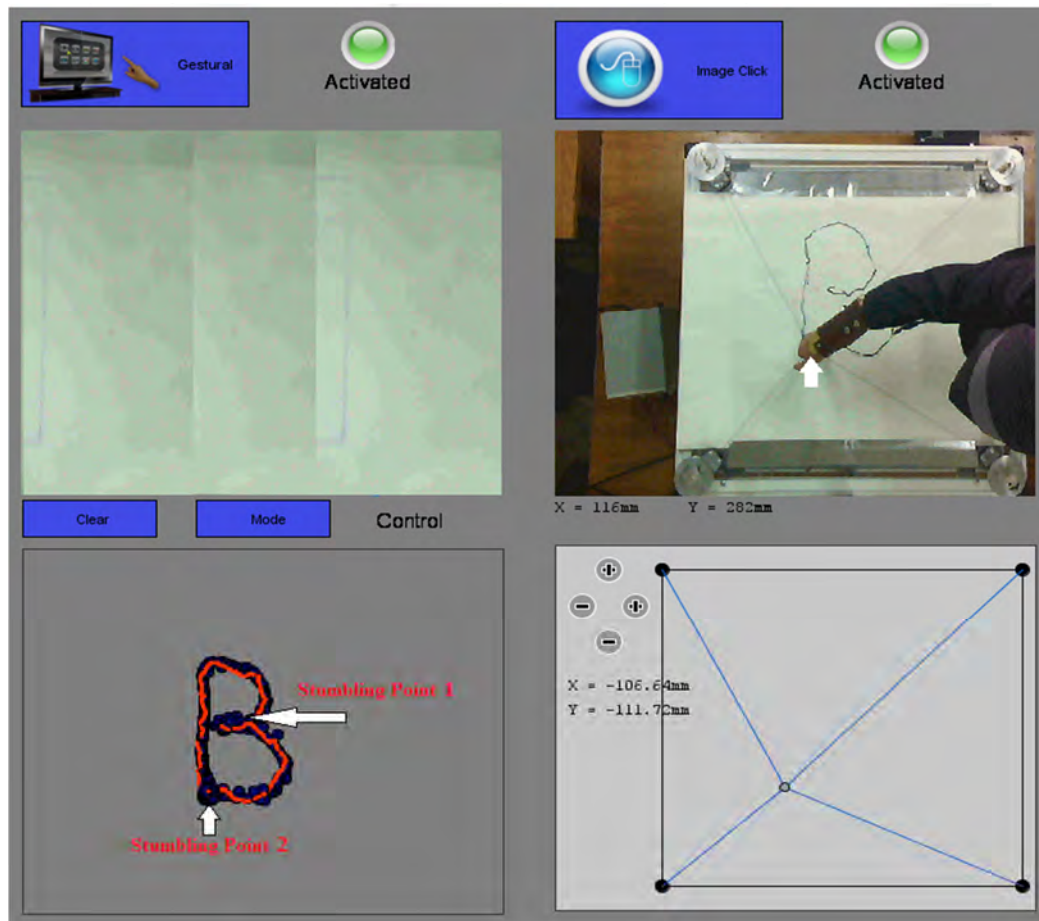


Figure 5.24- Intervention dans deux points de blocage

5.4 L'analyse de la dépendance par rapport à l'environnement de programmation et le système expérimental utilisé

Dans le but d'analyser la dépendance de notre système interactif par rapport à l'environnement de programmation et le système expérimental utilisé. Nous avons conçu une autre interface multimodale pour contrôler un autre système expérimentale (figure 5.25). Cette interface multimodale permis l'intégration de deux modalités, la commande joystick et la commande par pointage sur image. Cette interface est implémentée en utilisant l'environnement de programmation graphique **Labview** « **L**aboratory **V**irtual **I**nstrument **E**ngineering **W**orkbench » à l'aide d'un ensemble d'outils et d'objets. Qui offre plus de flexibilité pour la conception et l'implémentation des interfaces graphiques d'utilisateur. Et qui permet de créer des applications modulaires (notion de sous VI) et extensibles, pour la conception d'applications de mesure, de contrôle et de test. Un VI contient deux éléments face-avant sert d'interface utilisateur et diagramme contient le code source graphique.

Notre système interactif est présenté par la figure 5.25, dont il est composé de l'interface multimodale (face avant) et un prototype expérimental d'un bras manipulateur porté sur une base mobile.



Figure 5.25- Le système interactif pour le bras manipulateur

L'interface graphique d'utilisateur globale (face-avant) est présentée par la figure 5.26 après la réalisation d'une tâche donnée, qui est dans ce cas, l'écriture d'une lettre de la langue arabe (ح).

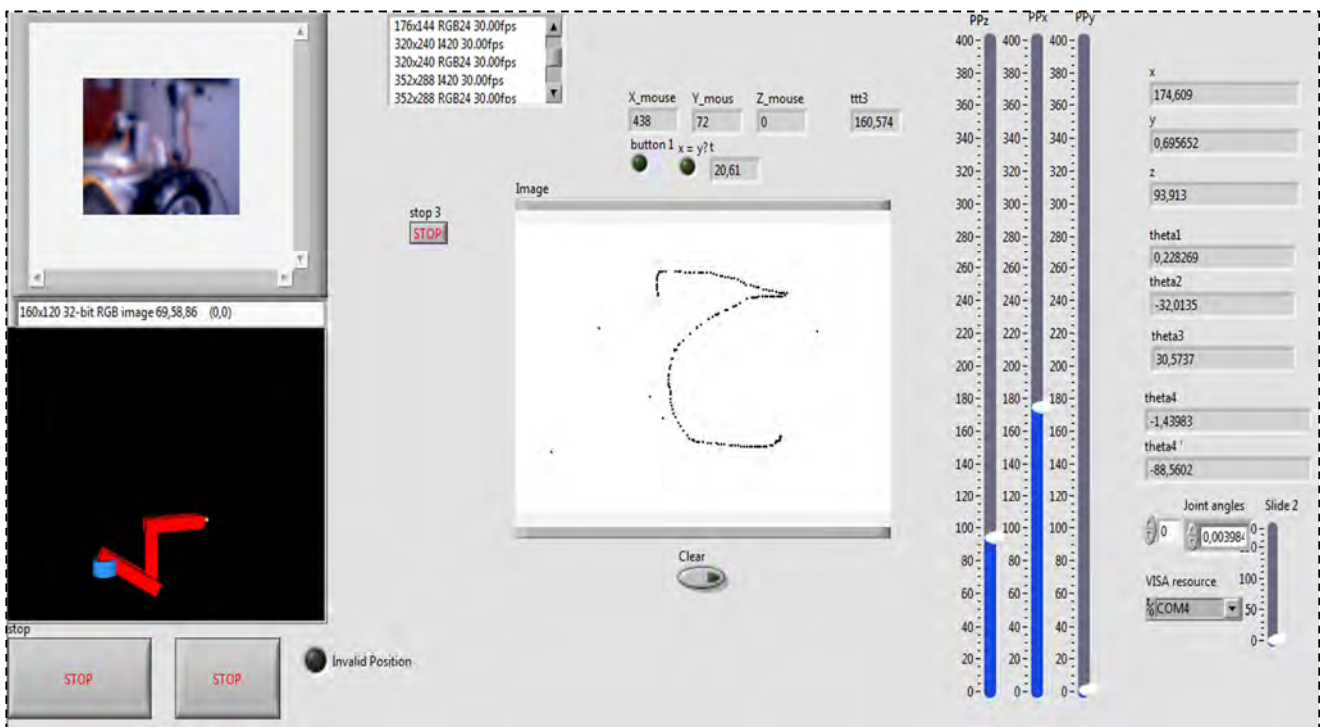


Figure 5.26- L'interface multimodale après l'acquisition d'un geste graphique

Nous avons remarqué sur la base de notre propre expérience d'implémentation de plusieurs interfaces multimodales en utilisant différents environnements de programmation ; Comme Matlab, Simulink, Processing et Labview. Que ces environnements sont équipés pour faciliter la communication avec du matériel tel que le GPIB, VXI, PXI, RS-232, RS-485 et des périphériques

d'acquisition de données enfichables. Et disposent aussi de fonctions intégrées pour connecter les applications à Internet à l'aide du serveur Web de LabVIEW et gère des protocoles standards tels que TCP/IP et ActiveX. Nous avons remarqué aussi l'existence d'une concurrence de développement entre ces environnements, dans le but d'englobé chacun le noyau l'autre environnement. Comme par exemple dans le cas de l'environnement Labview, nous pouvons remarquer l'existence du noyau de l'environnement Matlab ; qui nous permis d'utiliser les applications développées avec Matlab directement sous Labview avec un petit changement.

Sur la base de cette étude, nous pouvons dire que le développement des environnements de programmation est dans la direction de libérer le développement des interfaces multimodales de la dépendance par rapport à l'environnement de programmation. Il suffit de développer un langage de programmation avec un niveau d'abstraction élevé et avec une structure optimale, tout en gardant l'infrastructure existante des environnements de programmation tout en ajoutant des routines d'adaptation.

Nous pouvons dire aussi qu'après l'implémentation de la multimodalité pour contrôler le bras manipulateur, nous pouvons diviser l'interface graphique multimodale globale en deux sous parties. Une partie d'interfaçage operateur et une partie d'interfaçage robot, et que seulement celle de l'interfaçage robot qui a été soubise un changement remarquable. Donc d'une autre façon, en peut libérer l'interface multimodale de la dépendance par rapport au prototype expérimental tout en adaptant la partie d'interfaçage robot.

- **Possibilités de combinaison entre les différents modes de commande**

Sur la base des expériences que nous avons effectuées, nous pouvons distinguer quatre relations possibles d'utilisation entre les modes de commandes : le mode partagé, le mode combiné en parallèle, le mode indépendant et le mode inclusion.

- Le mode partage:

Cette relation (figure 5.27), est due de la possibilité d'utiliser des fonctionnalités de deux modes (d'une façon partielle ou complète) simultanément dans un intervalle de temps par rapport à une tâche donnée.

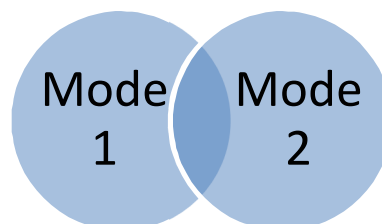


Figure 5.27- Mode partagé

- Mode combiné en parallèle :

Dans ce cas, l'opérateur utilise et combine deux différents types de modes de commande en même temps pour accomplir une tâche donnée (figure 5.28).

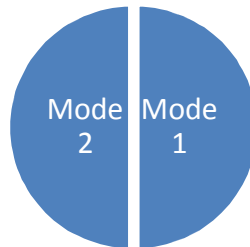


Figure 5.28- Mode combiné en parallèle

- Mode indépendant :

Dans le cas du mode indépendant, nous avons remarqué qu'il existe des situations où l'opérateur ne peut pas combiner des modes de nature différente ou contradictoires par rapport à une tâche donnée. Il existe seulement la possibilité d'utilisation indépendante ou d'enchaîner ce type de modes par rapport à une tâche donnée (figure 5.29).

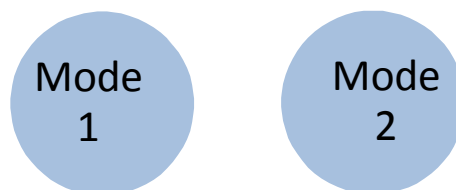


Figure 5.29- Mode indépendant

- Mode inclusion :

Sur la base de nos expériences, nous avons remarqué que l'intervalle d'utilisation de certains modes est inclus dans l'intervalle d'utilisation de d'autres modes de commande (figure 5.30).

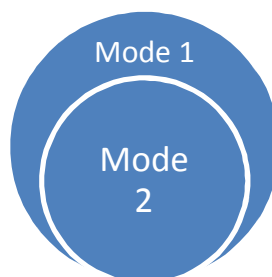


Figure 5.30- Mode inclusion

- **Le degré de complexité**

De la même façon, nous avons mis en évidence une relation entre le degré de complexité de la tâche et de l'environnement par rapport au nombre de modes opérationnels disponibles. A titre d'exemple, nous avons combiné le mode gestuel et le mode de pointage sur image pour certaines applications particulières dans un environnement complexe. Nous avons exploité les avantages des robots parallèles, dans une salle spéciale pour des applications de rééducation pour des étudiants handicapés. Cette application nous a guidé de remarqué que le degré de complexité de l'environnement augmente le degré de complexité de la tâche, et que la multimodalité en terme de nombre de modes combinés diminue le degré de complexité de la tâche (figure 5.31).

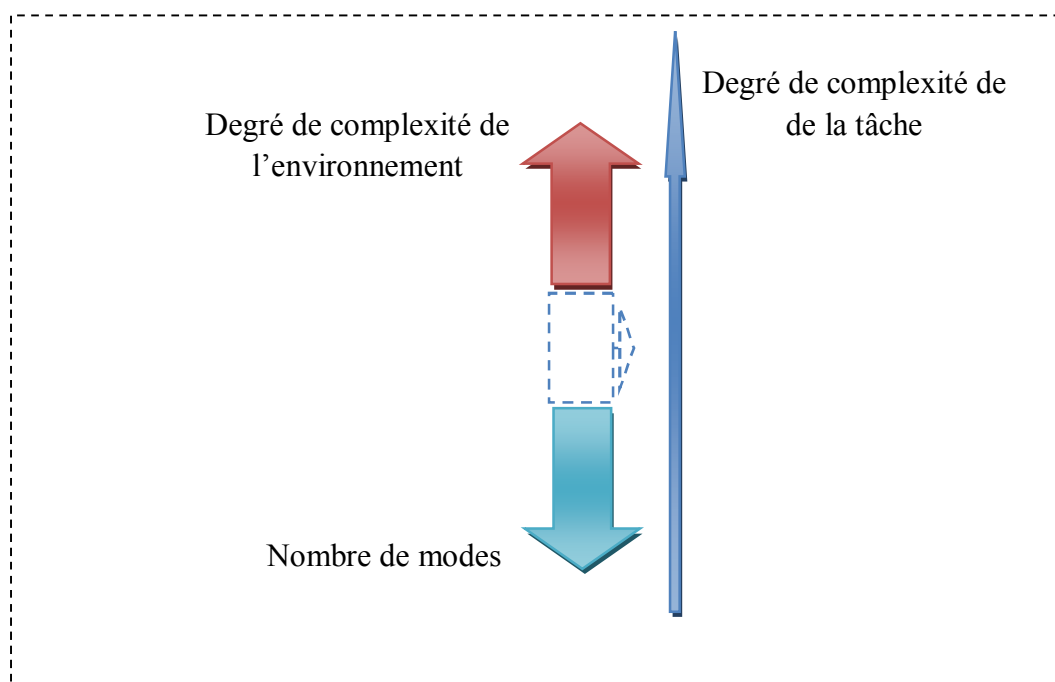


Figure 5.31-Le degré de complexité

- **Le modèle général de l'interface multimodale**

Sur la base de cette analyse expérimentale, nous pouvons proposer le modèle générale de l'interface multimodale présenté par la figure 5.32. Nous avons décomposé cette interface en deux sous couches : la couche d'interfaçage opérateur et la couche d'interfaçage robot. Dans la couche d'interfaçage opérateur nous proposons de travailler dans un niveau d'abstraction élevé orienté mode de commande (M_1, M_2, \dots, M_n). Cette approche est inspirée sur le modèle orienté objet bien connue en programmation informatique. De plus, nous avons assuré une structure optimale pour notre interface qui constitue noyau principal. Des fonctions d'adaptation (FT_1, FT_2, \dots, FT_n) sont ajoutées pour relier le noyau de l'interface multimodale avec l'environnement de programmation installé sur la machine (EV_1, EV_2, \dots, EV_n). Cette idée s'inspire du mode CORBA.

Pour la couche d'interfaçage robot, nous recommandons de travailler à un niveau d'abstraction orienté système (parallèle, sériel, mobile, hybride, ..., etc.). Le block d'interfaçage (BI1, BI2, ..., BI_n) est sélectionné selon le prototype expérimental utilisé.

Ce modèle nous donne la possibilité de développer des interfaces multimodales généralisables, réutilisables et indépendantes par rapport à l'environnement de programmation et au prototype expérimentale utilisé.

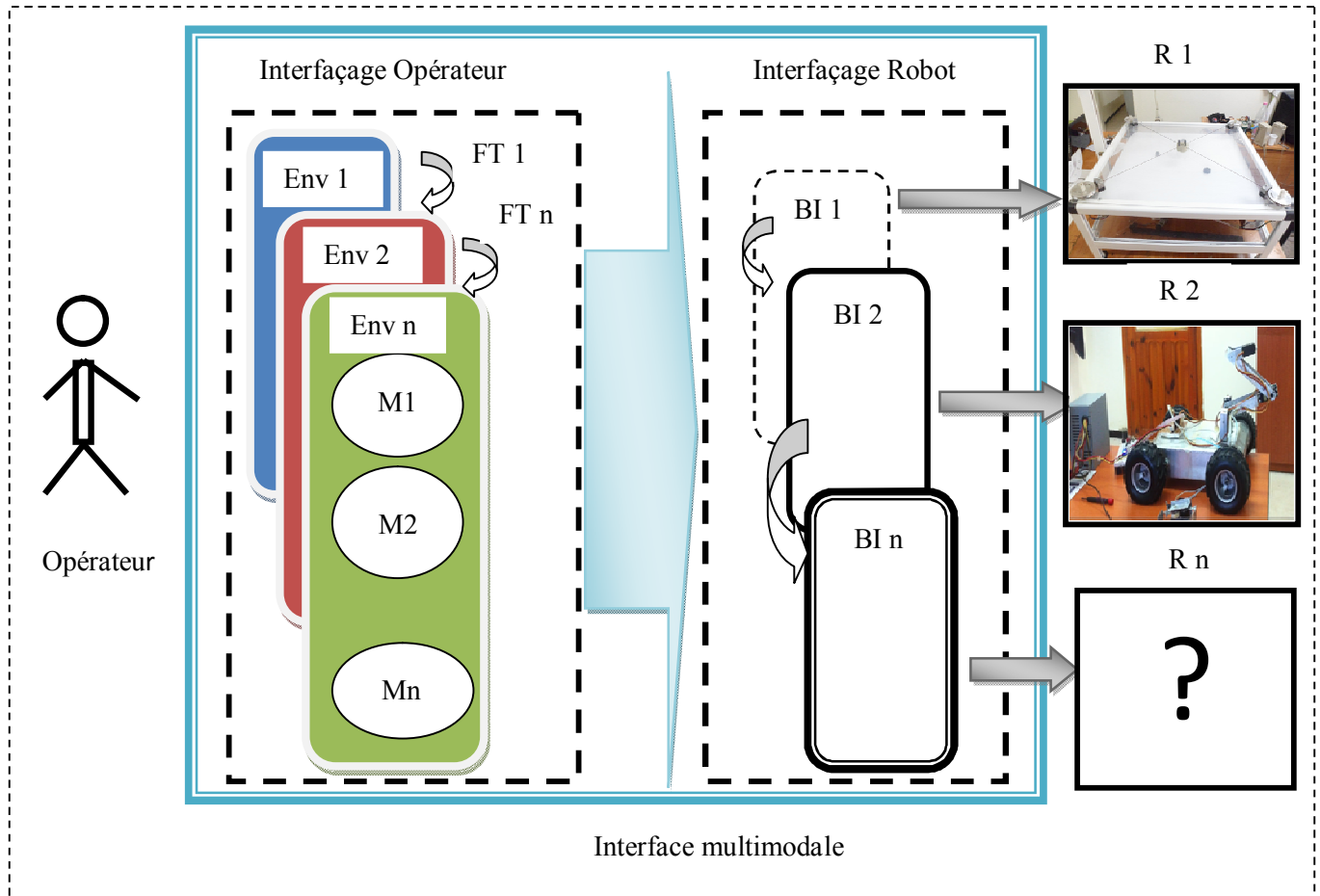


Figure 5.32- Le modèle général de l'interface multimodale

5.5 Conclusion

Dans ce chapitre, nous avons effectué une analyse préliminaire sur l'interaction homme-robot, dans le but de créer une base d'étude expérimentale pour le choix des méthodes mathématiques d'analyse et l'évaluation des performances des systèmes supervisés. Cette analyse nous a permis de tester chaque mode de commande. Elle nous a permis aussi de mettre en évidence les caractéristiques ainsi que les possibilités et les limites d'utilisation de chaque mode tenant compte du système opérationnel utilisé et de l'environnement de travail. Elle a permis également de mettre en évidence

certaines facteurs qualitatifs et quantitatifs relatifs au facteur humain qui influent sur les performances du système global.

Ensuite, pour prendre en charge des missions et des tâches plus ou moins complexes qui nécessitent la multimodalité; une analyse des combinaisons possibles des différents modes a été menée. Dans le but d'établir une relation entre le degré de complexité de la tâche, le degré de complexité de l'environnement ainsi que le nombre de modes disponibles, nous avons conduit nos expériences dans deux types d'environnements de travail: un environnement simple et un environnement complexe.

Les tests expérimentaux ont donné satisfaction; l'interface multimodale a permis en effet d'utiliser les modes individuellement ou en combinaison et enchainement selon le contexte et le souhait de l'opérateur pour mener à bien des tâches plus ou moins complexes. Nous avons pu dégager de nos expériences sur différents systèmes et opérateurs; un modèle conceptuel d'interface multimodal avec un niveau d'abstraction élevée pour assurer une flexibilité pour les opérateurs et une adaptabilité par rapport à l'environnement.

L'exploitation de l'interface multimodale par différents opérateurs a montré qu'il faut tenir compte pour l'analyse des performances des supervisés multimodaux de la combinaison des paramètres qualitatifs et quantitatifs. Ce sera le sujet du chapitre suivant.

Chapitre 6

Évaluation des Performances pour les Systèmes de Contrôle Multimodaux Supervisés

Sommaire

6.1	Introduction	169
6.2	Travaux en relation avec l'AHP	169
6.3	Aperçu sur les SCIS	171
6.4	Interfaces multimodale intégrées	172
6.4.1	IGU pour robots industriels	172
6.4.2	IGU pour le robot a câbles	173
6.4.3	Mesure des performances pour SCIS	175
6.4.3.1	Les mesures des performances et les critères	175
6.4.3.2	Des expériences d'illustration du processus d'apprentissage	175
6.5	Évaluation de la performance des tâches au moyen d'AHP	178
6.5.1	Exemple 1 (robot industriel)	178
6.5.2	Exemple 2 (Robot à 4 câbles)	183
6.6	Conclusion	187

6.1 Introduction

Dans les HMIS « Human-Machine Interaction Systems », un but principal de l'automatisation est la réduction de l'intervention humaine dans l'exécution de la tâche, tout en augmentant les performances du système. Dans les systèmes complexes, l'évaluation de l'exécution des tâches dépend de nombreux critères et des facteurs qui peuvent être quantitatifs, qualitatifs, ou les deux. Certaines approches ont tenté de fournir des moyens d'évaluation de l'exécution de la tâche [93, 94, 95]. Dans les SCIS « Systèmes de Contrôle Intégrés et Supervisés », de nouveaux problèmes apparaissent en raison de la possibilité d'effectuer la même tâche à l'aide de diverses combinaisons de modes de commande. Par exemple, lorsque l'opérateur effectue des tâches au moyen des SCIS, il / elle doit choisir les modes de contrôle (vision, parole, geste, télémanipulation, etc). Dans chaque mode de commande sélectionné, l'opérateur peut également allouer des sous-tâches au module autonome, à l'opérateur, ou en coopération à tous les deux. Ce processus en temps réel de la sélection d'actions et de la répartition des tâches dynamiques est, en général dépendant du système, du contexte et de l'opérateur. Pour chaque expérience, la composition, l'ordre, les instants de commutation, la durée d'utilisation des modes de contrôle sélectionnés, et la répartition des tâches sont tous des facteurs qui influencent sensiblement les performances du système global [96, 97]. Les relations entre les performances du système pour l'exécution de la tâche et les facteurs qui l'influencent ne sont pas toujours faciles à établir. Une question pertinente est donc de savoir comment évaluer la performance du système pour la réalisation d'une tâche donnée par différentes combinaisons de modes de contrôle et / ou par différents opérateurs.

Ce chapitre traite de certains aspects de l'SCIS qui est caractérisé par l'interaction multimodale et le contrôle haut niveau. Une technique pour l'évaluation de l'exécution des tâches est présentée. Nous proposons des mesures de performances pour exprimer principalement l'efficacité de la réalisation des tâches ainsi que la charge de travail soutenue par l'opérateur humain. La méthode que nous avons adaptée pour fusionner des critères quantitatifs et qualitatifs est l' AHP « Analytical Hierarchy Process ». Nous appliquerons cette méthode pour deux types de systèmes: un robot parallèle à câbles et un robot manipulateur sériel. D'autre part, on présentera la méthode AHP ainsi que sont illustration au fur et à mesure de son application sur deux exemples choisis.

6.2 Travaux en relation avec l'AHP

La méthode AHP a été initialement développée par Saaty en 1980 [98]. Il s'agit d'un processus décisionnel puissant et flexible d'aide pour résoudre les problèmes, multicritères complexes où des facteurs qualitatifs et quantitatifs doivent être incorporés. Il est utile dans les situations où le décideur doit choisir entre différentes alternatives sur la base de la façon dont ils répondent à divers attributs.

Dans un travail précédent, nous avons utilisé la technique AHP pour évaluer les performances de la tâche pour les systèmes intégrés de contrôle multimodales. Il a été appliqué à un manipulateur industriel, en tenant compte des attributs et des critères qualitatifs et quantitatifs [99].

L'analyse de l'état actuel de l'art concernant l'application de la technique AHP à IHM montre que nous pouvons diviser ses applications comme un outil d'évaluation ou comme procédure d'évaluation et de prise de décision.

Dans le domaine de la conception de l'interface graphique utilisateur, la référence [100] a proposé un modèle pour la conception d'interface graphique audio-visuelle basée sur la technique AHP. L'objectif de ce modèle est de réduire l'erreur humaine. Sept critères d'évaluation ont été utilisés comme alternatives. La décision est basée sur quatre stratégies de conception pour l'interface utilisateur: l'assistance par la vision, l'assistance par la cognition, l'assistance par opération et l'assistance par mémorisation pour obtenir l'interface la plus appropriée pour chaque utilisateur (stratégies Multi-interface).

En termes de techniques d'évaluation, [101] présente une méthodologie basée sur l'AHP qui a été utilisée pour évaluer et classer les différentes approches de l'automatisation des fonctions des tâches prévues pour la station spatiale de la NASA. L'objectif principal est d'optimiser la coopération homme-machine pour les fonctions et les compétences au niveau des modules et sous-systèmes. En raison du grand nombre de facteurs impliqués dans le modèle, le problème global est décomposé en quatre sous-problèmes qui sont respectivement la productivité individuelle humaine, l'économie, la conception et l'opération. Chaque sous-problème est divisé en plusieurs niveaux de critères. Une attention particulière est accordée à l'impact de l'automatisation avancée sur la productivité humaine. Nous notons que le critère de la productivité humaine est utilisé comme objectif dans le premier sous-problème et il est basé sur l'interface homme-machine comme un facteur important. Nous notons également que dans le problème des opérations, l'acceptabilité a été proposé comme critère et basé sur le facteur « facilité d'utilisation » comme un attribut qualitatif pour évaluer l'opération. Un autre exemple dans [102] indique que l'évaluation quantitative est effectuée en utilisant la méthode AHP sur un système d'étude pour faire une comparaison analytique entre un système humain seulement et la collaboration homme-robot dans les systèmes de fabrication.

En effet; plusieurs recherches montrent la possibilité d'intégrer la méthode AHP avec d'autres techniques d'évaluation ou de prise de décisions. Dans [103], les auteurs décrivent une fusion de l'AHP et d'un contrôleur de type BLS « Brain Limbic System » pour une stratégie de commande des robots mobiles. Il est suggéré pour contrôler la navigation du robot mobile avec évitement d'obstacles dans un environnement inconnu. Dans cet exemple d'application, la méthode AHP est utilisée pour choisir le chemin optimal qui est basé sur trois critères. Le contrôleur BLS est utilisé pour assurer le déplacement du robot vers la position souhaitée.

Sur la base de notre analyse de cet état de l'art, nous avons remarqué qu'il n'y avait pas de travaux rapportés concernant l'utilisation de la technique AHP pour l'évaluation des SIHM (Systèmes d'Interaction Homme-Machine) multimodaux en combinant les attributs à la fois qualitatifs et quantitatifs. De plus, notre technique proposée peut permettre non seulement l'évaluation de la réalisation de la tâche, mais suggère la combinaison la plus appropriée de modes de commande pour conduire de la meilleure façon une mission spécifique pour un opérateur donné.

6.3 Aperçu sur les SCIS

Le premier système physique (figure 6.1) a déjà été décrit dans [99]. Il se compose d'un robot industriel Kuka-361 commandé par une version interactive du logiciel de simulation nommé COMRADE [104]. COMRADE est un logiciel conçu à la K.U. Leuven pour faciliter le développement et l'optimisation des systèmes robotiques intégrant le contrôle de force. Un capteur de force et une caméra CCD sont montés à l'extrémité de l'effecteur du système commandé (le robot distant). La caméra CCD est reliée à un moniteur vidéo pour afficher les scènes capturées à l'opérateur et à une carte d'acquisition DSP. Les images capturées sont transmises à une station de travail Silicon Graphics qui accueillera les routines de traitement d'image [105]. Un système d'acquisition de la parole humaine avec un vocabulaire limité est également mis en œuvre [106]. Sur le site de commande, l'opérateur utilise un contrôleur manuel (Joystick) à 6 degrés de liberté, un module de vision et un module vocal pour contrôler le robot à distance. L'information de rétroaction est assurée par la vision directe si possible, ou par les images captées par la caméra CCD et d'autres capteurs de mesures, tels que la force et la position.

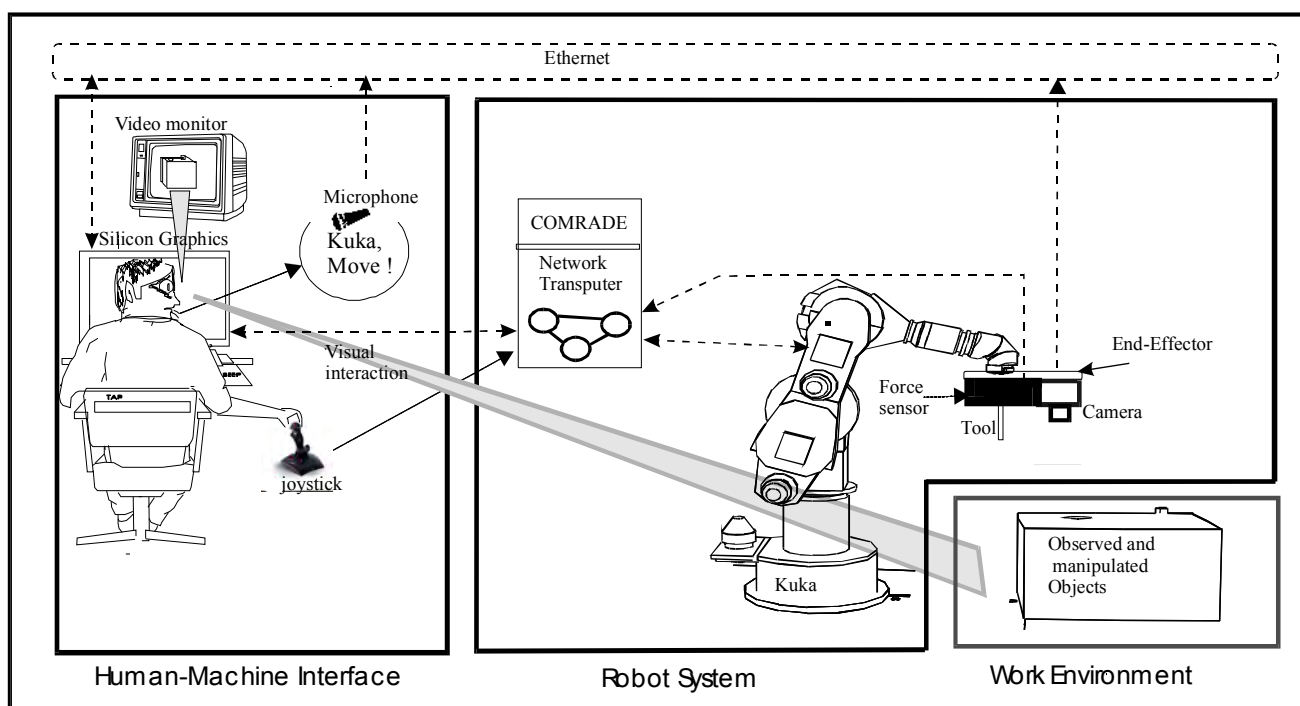


Figure 6.1-Vue d'ensemble du premier système expérimental

Le deuxième système physique (figure 6.2) est déjà décrit dans [107,108]. Il s'agit d'un nouveau type de manipulateur parallèle qui est un robot à câbles. Le système expérimental est contrôlé via une interface multimodale qui permet l'intégration de plusieurs modes contrôle de haut niveau tels que: le contrôle basé sur la voix, le contrôle basé sur le pointage sur image, le contrôle basé sur le geste et le contrôle basé sur le joystick. Le prototype de robot est équipé de deux systèmes de vision; le premier est utilisé pour le contrôle gestuel et le deuxième pour la commande de pointage sur image. A cette fin, deux caméras CCD sont utilisées; la première est placée dans le site de l'opérateur pour suivre les mouvements gestuels et la deuxième sur le site même du robot. L'information de rétroaction est assurée par vision directe, les images captées par la caméra CCD et d'autres capteurs de mesures tels que la force et la position.

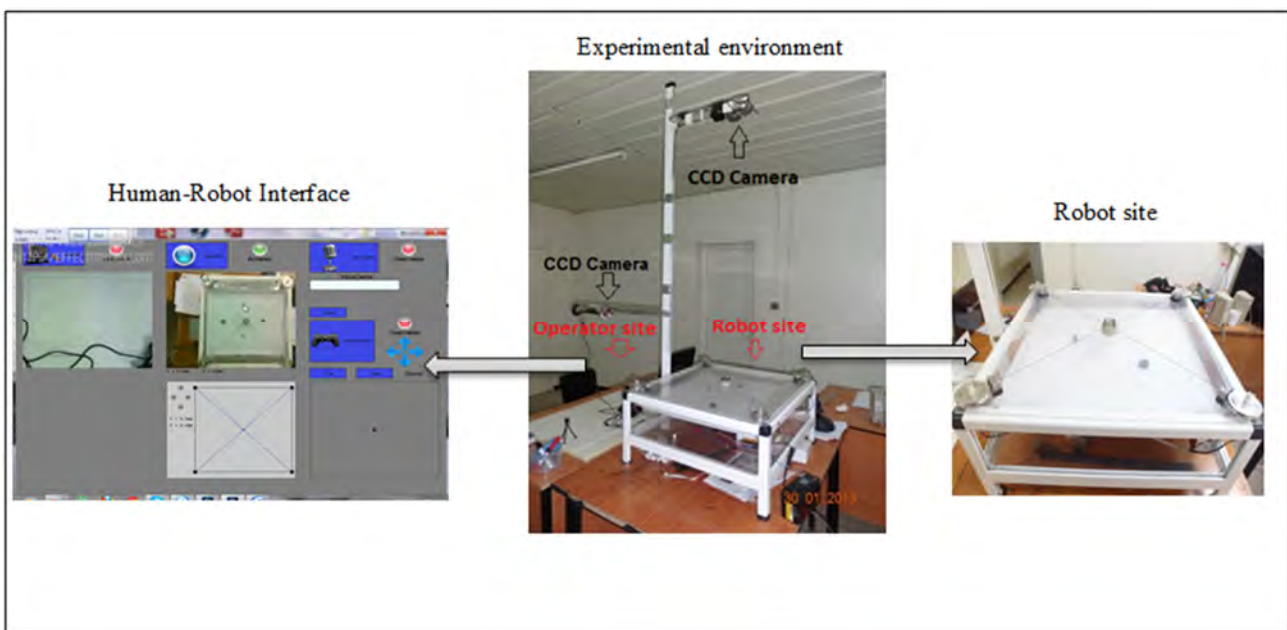


Figure 6.2- Vue d'ensemble du deuxième système expérimental

6.4 Interfaces multimodale intégrées

6.4.1 IGU pour robots industriels

Pour accéder et gérer les différentes fonctionnalités de l'SCIS, une interface intégrée multimodale d'opérateur a été implémentée. L'opérateur peut sélectionner les modes de contrôle qu'il / elle préfère ou considère comme adapté à la réalisation des tâches spécifiques. Les modes de contrôle disponibles vont de la télémanipulation à un contrôle pratiquement autonome. L'opérateur et le système autonome peuvent effectuer des sous-tâches alternativement (traded control), partager les tâches (shared control), superposer leurs actions (superimposed control). D'autres fonctions sont également disponibles enrichissant les possibilités d'interaction la programmation par démonstration est également disponible (PDH) [109, 110]. L'interface opérateur sert à la sélection et de la commutation

entre les modes de contrôle pour la simulation, la télé-programmation et la supervision. Pour maintenir autant que possible des conditions similaires à l'opérateur pour surveiller une tâche dans le monde réel ainsi que dans le monde virtuel, la même interface utilisateur multimodale est utilisée.

Dans l'SCIS conçu, la souplesse dans sélection et le changement des modes a été examiné. Par exemple, l'SCIS décrit peut gérer successivement les modes de contrôle sélectionnés. Mais, il peut également être configuré pour combiner les modes pour toutes les variables contrôlées (articulaires ou cartésiennes) à tout moment. Cette possibilité permet à l'opérateur d'intervenir en ligne au cours d'un état d'attente« wait state » ou lors des opérations autonomes.

6.4.2 IGU pour le robot a câbles

L'interface graphique d'utilisateur conçue et mise en œuvre pour contrôler notre prototype de robot à câbles est représentée par la figure 6.3. Elle permet l'intégration de quatre modes de contrôle haut niveau: le mode pointer et cliquer (figure 6.4), le mode gestuel (Figure 6.5), le mode vocal et le mode basé sur le joystick (figure 6.6). Cette interface utilisateur a été conçue selon les principes de conception centré utilisateur [111].

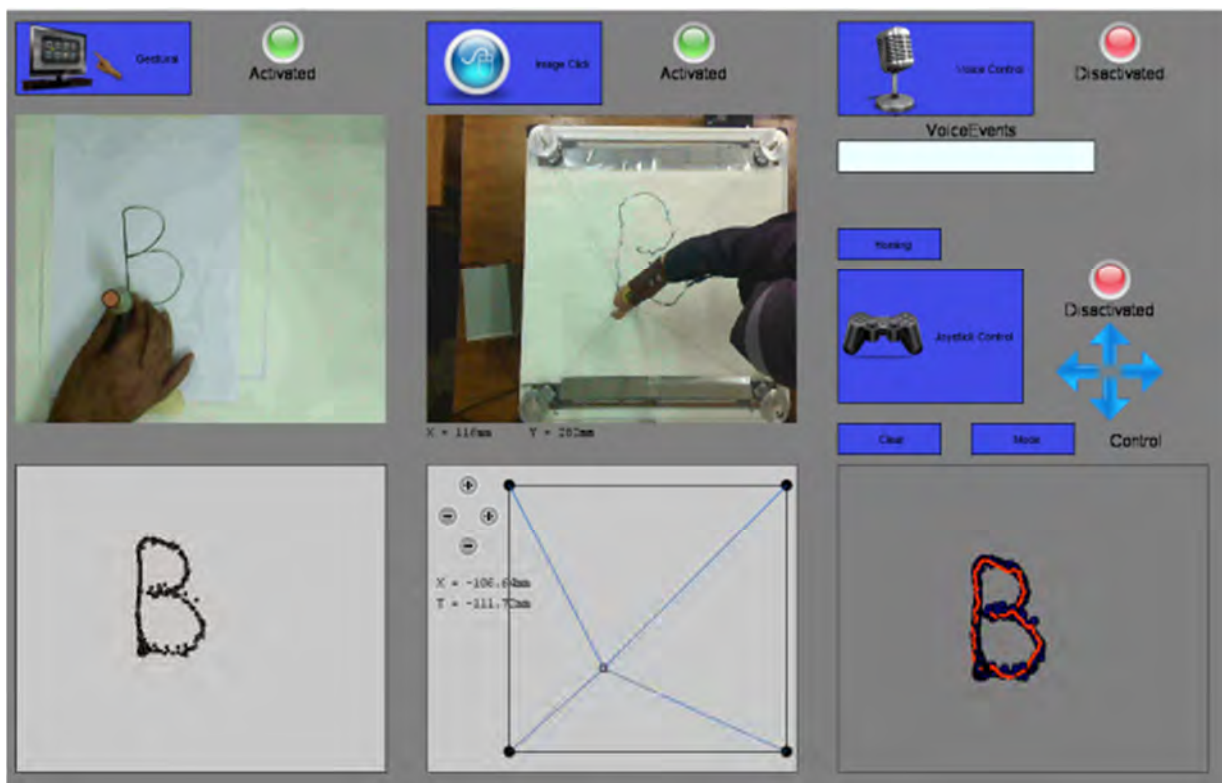


Figure 6.3- L'interface multimodale

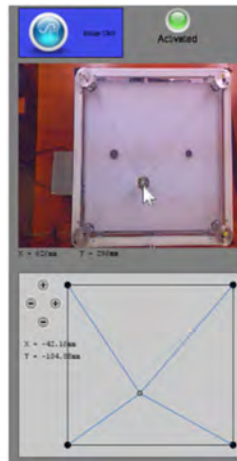


Figure 6.4- Le mode pointer et cliquer

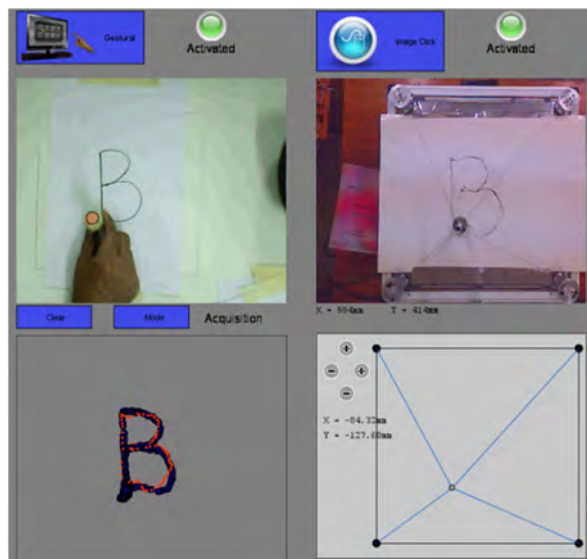


Figure 6.5- Le mode gestuel



Figure 6.6- Le mode vocal et joystick

6.4.3 Mesure des performances pour SCIS

6.4.3.1 Les mesures des performances et les critères

Pour évaluer et comparer les différentes expériences d'une tâche donnée effectuée par le même ou par différents opérateurs, les attributs et les critères de performance sont à définir. Cependant, de nombreux critères et mesures peuvent être définis. Des exemples de tels critères sont: minimiser le temps d'achèvement de la tâche, minimiser la consommation d'énergie, augmenter la sécurité, maximiser la précision, minimiser le taux d'erreurs et d'accidents, maximiser la facilité d'utilisation, maximiser de la confiance des utilisateurs par rapport au système, etc.

Dans notre étude, nous cherchons à comparer deux réalisations différentes de la même tâche effectuées par l'SCIS. Du point vue des facteurs, nous avons proposé trois caractéristiques qui sont importantes pour évaluer un système d'interaction homme-machine: le temps d'exécution, exprimant la durée d'exécution de la tâche; les erreurs, exprimant la qualité du résultat attendu; et la charge de travail mental supporté par l'opérateur humain. Évidemment, le but visé est de choisir le cas qui présente les plus brefs délais d'exécution de la tâche, qui donne la meilleure précision et qui présente le minimum possible de la charge de travail pour l'opérateur humain.

Dans cette application, nous allons nous concentrer sur deux mesures principales: le temps d'exécution de la tâche et certains aspects de la charge de travail. Le temps d'exécution des tâches est notée CT « Completion Time ». En ce qui concerne la charge de travail, qui est considéré dans le cas général comme la pression affectée sur l'opérateur [112, 113]. De plus, la distinction peut être faite entre la charge de travail physique et mental. Bien que la charge de travail physique puisse être mesurée, la charge de travail mental n'est pas directement mesurable. L'implication physique de l'opérateur sera liée au temps de télé-opération (TT) « Tele-operation Time » et du nombre d'interactions de l'utilisateur avec l'interface graphique (NI) « Number of Interactions ». La charge de travail mental sera estimée selon le jugement personnel de l'opérateur humain sur la facilité ou la difficulté pour effectuer la tâche. Pour chaque expérience, CT, TT et NI sont calculés automatiquement une fois que la tâche est réalisée alors que la charge mentale exprimée avec par degré de la facilité d'utilisation (EU) « Ease of Use » est estimée selon l'appréciation subjective de l'opérateur.

6.4.3.2 Des expériences d'illustration du processus d'apprentissage

Quelques exemples d'application visant à mettre en évidence la façon dont le processus d'apprentissage améliore les performances en ce qui concerne le nombre d'expériences seront présentées et discutés. L'expérience a été conduite avec le robot manipulateur industriel (figure 6.7) et avec un robot parallèle à 4 câbles (figure 6.8).

La figure 6.7 montre l'image de l'environnement de travail tel que vu par la caméra placée près de l'effecteur final du robot Kuka-361. Dans cette image, nous pouvons identifier un objet cylindrique posé au milieu du côté gauche de la boîte noire. Cet objet cylindrique contient un trou en son centre reconnu comme un cercle noir et appelé A. Dans le haut de la partie droite de la grande boîte noire, on peut reconnaître un cube sur lequel il y a un papier disposé dans cette zone blanche est appelé B.

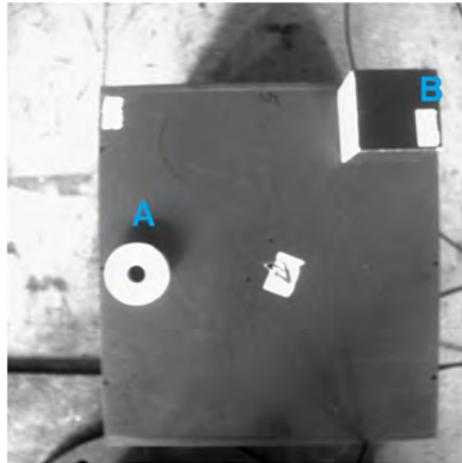


Figure 6.7- Montage expérimental (Kuka-361)

La tâche consiste à diriger, par le système de vision, le robot à partir d'une position initiale vers la position A (figure 6.7). Ensuite, à l'aide de la télémanipulation seulement, de déplacer l'effecteur final de A vers B. Dans cet exemple, le temps de traitement est enregistré avec le numéro d'expérience (Figure 6.8).

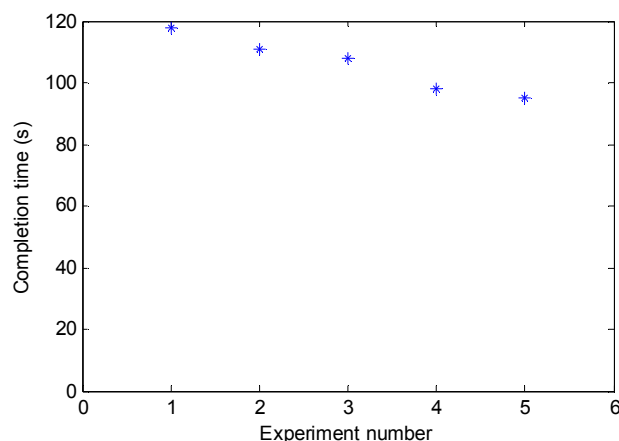


Figure 6.8- L'amélioration des résultats avec les expériences

La seconde application concerne l'exécution d'une tâche au moyen d'un robot à 4 câbles. La réalisation de la tâche sera obtenue par l'utilisation de deux modes de commande. Le premier mode de commande sera le pointage sur image et le second mode de commande sera la télémanipulation par joystick. La tâche consiste à diriger, par le système de vision, le robot à partir d'une position

initiale à une position A (figure 6.9). Ensuite, à l'aide du mode de commande par joystick, nous déplaçons l'effecteur final de A à B. Dans cet exemple, le temps d'achèvement est enregistré par rapport au numéro de l'expérience (figure 6.10).

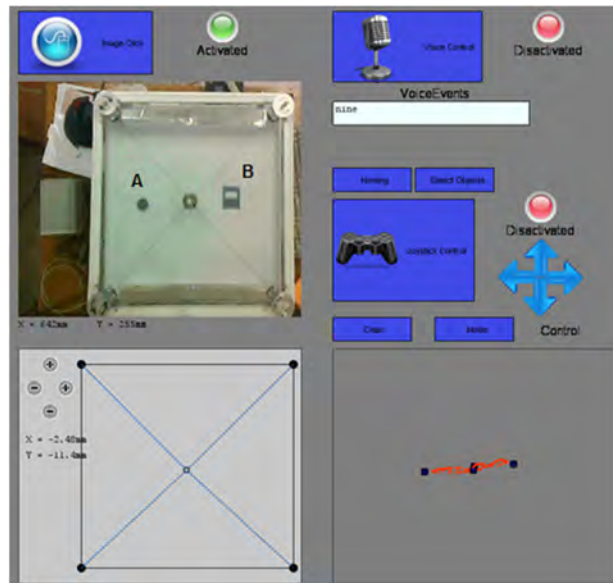


Figure 6.9- Montage expérimental (robot a 4 câbles)

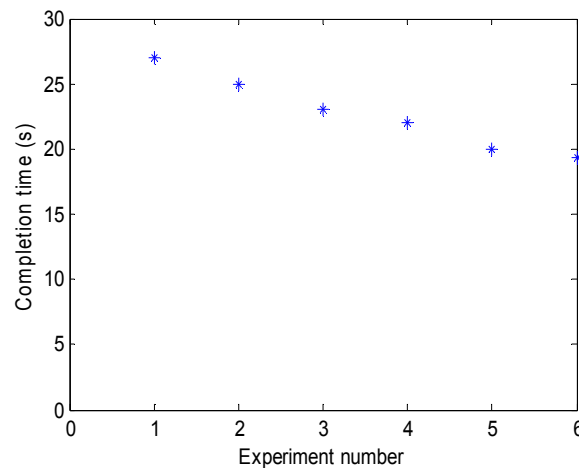


Figure 6.10-L'amélioration des résultats en fonction des expériences

L'observation des résultats présentés concernant les deux systèmes expérimentaux conduit à conclure que le temps d'exécution de la tâche est décroissant et montrant une amélioration des performances de la tâche par rapport au nombre d'expériences. Ils montrent à quelle vitesse l'opérateur apprend à maîtriser la réalisation de la tâche. Dans les deux expériences, le temps d'achèvement par rapport au nombre d'expérience s'améliore et se stabilise au bout d'environ 6 expériences. Ainsi, dans le cas où la séquence des modes de commande est fixe, l'exécution de la tâche va dépendre principalement de l'opérateur. Cependant, pour des tâches complexes, il y a généralement de nombreuses alternatives pratiques pour accomplir une tâche donnée.

6.5 Évaluation de la performance des tâches au moyen d'AHP

Dans cette section, nous allons décrire l'adaptation de la méthode AHP pour l'évaluation de l'exécution des tâches. Nous allons appliquer cette méthode pour deux SCIS différents: sur un robot industriel et un robot à 4 câbles. Dans le cadre de notre application, nous allons présenter au fur et à mesure les étapes essentielles de la méthode AHP.

6.5.1 Exemple 1 (robot industriel)

Cet exemple concerne deux exécutions de la même tâche menées par deux combinaisons différentes de modes de contrôle. La première tâche consiste à utiliser le contrôle par pointage sur image pour aller d'un point initial à A, puis d'utiliser le mode par télémanipulation pour se déplacer de A à B. La deuxième tâche consiste à déplacer l'effecteur final en utilisant le mode de contrôle basé sur le pointage sur image d'une position initiale à A, puis en utilisant de nouveau le contrôle basé sur le pointage sur image pour se déplacer de A à B.

Dans cette technique, des facteurs quantitatifs ainsi que les facteurs et critères qualitatifs peuvent être fusionnés. Les facteurs quantitatifs considérés sont le temps d'achèvement CT, le temps de télémanipulation TT, et le nombre d'interactions NI. CT, TT et NI sont automatiquement enregistrées à la fin de chaque expérience. Le facteur qualitatif et subjectif (facteur de préférence) exprimant le degré de facilité d'utilisation (EU) est estimé comme il est ressenti par l'opérateur. Bien entendu, nombreux autres facteurs qualitatifs et quantitatifs peuvent être ajoutés.

Le modèle AHP présenté dans la figure 6.11 est organisé en une hiérarchie avec un objectif principal (Evaluation de la tâche) en haut du modèle, suivie par les alternatives en tant qu'ensemble de possibilités dans la couche médiane. La hiérarchie se termine sur la partie inférieure du modèle avec les critères requis pour atteindre l'objectif [98].

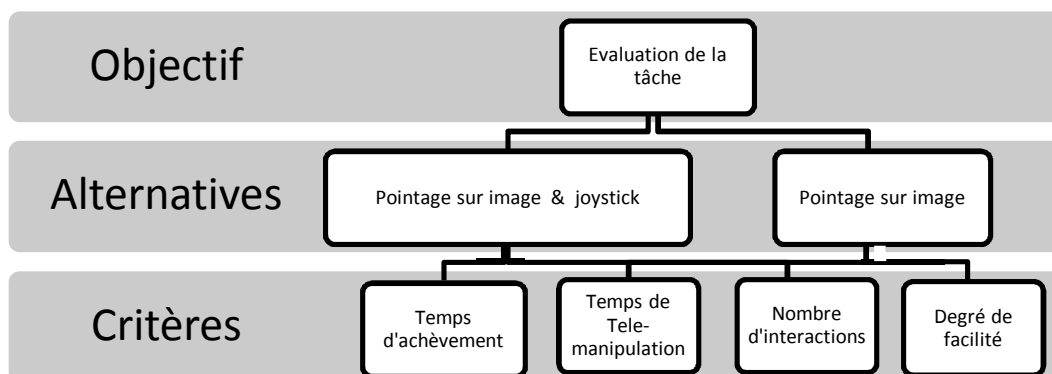


Figure 6.11- Le modèle AHP

Les résultats obtenus pour les deux réalisations de la tâche sont présentés dans le tableau 6.1. Pour les facteurs et critères qualitatifs, une quantification floue est nécessaire en utilisant certaine méthodologie qui sera affiché plus tard (tableau 6.2). Dans le tableau 6.1, le degré de facilité d'utilisation a été quantifié comme on le voit dans la 4^{ème} colonne.

Tableau 6.1- Valeurs des différents attributs (critères) pour différentes alternatives

	CT(sec)	TT(sec)	NI	EU
Expérience 1	94	23	11	6
Expérience 2	146	12	18	2

Comment décider quelle solution est préférable en ce qui concerne les résultats obtenus? Ce qu'on appelle processus de hiérarchie analytique peut être un outil approprié pour résoudre ces problèmes. Il est présenté et illustré en même temps avec l'exemple suivant. La méthode AHP implique les étapes suivantes:

- Développer la représentation hiérarchique du problème (figure 6.11).
- Établir la fonction objective globale. La fonction objective globale pour chaque alternatif k est estimée par l'expression suivante:

$$F_k = \sum_{i=1}^n w_i c_k^i \quad (6.1)$$

Où: w_i est le poids relatif de l'attribut i , c_k^i est le score relatif de l'alternative k pour un attribut i , et n est le nombre total d'attributs. L'évaluation d'une alternative nécessite donc l'estimation du poids relatif w_i et les scores c_k^i .

Estimer les poids w_i des attributs: A cet effet, il faut effectuer la comparaison de l'importance de chaque attribut par rapport aux autres. Cette importance doit être mesurée dans une échelle de valeurs entières de 1 à 9, ou chaque valeur a_{ij} à une interprétation présentée dans le tableau 6.2 [114].

Tableau 6.2- Table de comparaison par paires

valeurs a_{ij}	Interpretation
1	L'objectif i et j sont d'une importance égale
3	L'objectif i est faiblement plus important que l'objectif j
5	L'objectif i est fortement plus important que l'objectif j
7	L'objectif i est fortement très important que l'objectif j
9	L'objectif i est absolument plus important que l'objectif j
2,4,6,8	Sont des valeurs intermédiaires

Cette comparaison est représentée dans une matrice dite matrice de comparaison par paires (tableau 6.3). Pour cet exemple, un élément a_{ij} de la matrice A exprime l'estimation de l'importance de l'attribut i par rapport à l'attribut j selon l'échelle présentée dans le tableau 6.2.

Tableau 6.3- Evaluation pour les quatre attributs

	CT	TT	NI	EU	poïds
CT	1	4	5	6	0,598
TT	1/4	1	2	3	0.203
NI	1/5	1/2	1	2	0,123
EU	1/6	1/3	1/2	1	0,076

Par exemple, dans le présent cas, $a_{12} = 4$, signifie simplement que le premier attribut CT est en milieu entre faiblement et fortement plus important par rapport au deuxième attribut qui est TT. $a_{23} = 3$ signifie que le TT est faiblement plus important que EU. Bien sûr, la réciprocity implique que : $a_{ii} = 1$ et $a_{ji} = \frac{1}{a_{ij}}$.

Les poids relatifs des attributs sont calculés selon les formules suivantes:

$$w_i = \frac{\sum_{j=1}^n a_{ij}^{norm}}{n} \tag{6.2}$$

$$\text{Où } a_{ij}^{norm} = \frac{a_{ij}}{\sum_{i=1}^n a_{ij}} \text{ et } \sum_{i=1}^n w_i = 1$$

Dans ce cas, les valeurs numériques des w_i sont ($w_1 = 0.60$, $w_2 = 0.20$, $w_3 = 0.12$, $w_4 = 0.08$).

Ensuite, nous nous estimons le score de chaque l'alternative pour un attribut. Pour déterminer ces scores, nous construisons pour chaque attribut une matrice de comparaison par paires dans laquelle les lignes et les colonnes sont les alternatives possibles. Dans le cas où il s'agit d'une combinaison d'attributs quantitatifs et qualitatifs tels que l'exemple actuel, les mesures quantitatives ainsi que les jugements qualitatifs sont d'abord remis à l'échelle selon l'échelle donnée dans le tableau 6.2. Les valeurs et les jugements rendus dans le tableau 6.1 doivent d'abord d'être classés en logique floue selon l'échelle fournie dans le tableau 6.2. Dans le présent exemple, il y a deux alternatives (ALT1 et ALT2). La matrice de comparaison est donné par:

$$B^i = \begin{matrix} & \text{alt1} & \text{alt2} \\ \text{alt1} & 1 & b_{12}^i \\ \text{alt2} & \frac{1}{b_{12}^i} & 1 \end{matrix} \quad (6.3)$$

Un élément b_{ks}^i compare la k^{eme} alternative avec l'alternative pour l'attribut i . Pour chaque attribut i , une matrice B^i doit être construite sur la base des données fournies dans le tableau 6.1. La dimension de la matrice B^i est de $(m \times m)$ où m est le nombre total d'alternatives.

Les scores c_k^i sont ensuite estimés selon la formule suivante :

$$c_k^i = \frac{\sum_{s=1}^m b_{ks}^{i,norm}}{m} \quad (6.4)$$

$$\text{Où } b_{ks}^{i,norm} = \frac{b_{ks}^i}{\sum_{l=1}^m b_{ls}^i}, m \text{ et } \sum_{k=1}^m c_k^i = 1$$

Le tableau 6.4 représente la matrice de comparaison des deux alternatives où "V" désigne le contrôle basé sur le pointage sur image (vision) et "T" désigne le mode de commande par télémanipulation. Une évaluation simple de B^i est:

Tableau 6.4- Valeurs d'évaluation pour le temps d'exécution

	V&T	V	Poids
V&T	1	1/2	0.333
V	2	1	0.667

Tableau 6.5- Valeurs d'évaluation pour le temps de télémanipulation

	V&T	V	Poids
V&T	1	2	0.667
V	1/2	1	0.333

Table6.6- Valeurs d'évaluation pour le nombre d'interaction

	V&T	V	Poids
V&T	1	1/2	0.333
V	2	1	0.667

Tableau 6.7- Valeurs d'évaluation pour la facilité d'utilisation

	V&T	V	Poids
V&T	1	3	0,751
V	1/3	1	0,249

- Les tests de cohérence: avant d'appliquer la méthode AHP : la matrices A et Bⁱ qui sont obtenues par le jugement humain sont d'abord soumises à un test de cohérence. Ce test vérifie si les jugements faits pour comparer les attributs et les objectifs sont compatibles ou non. Un indice de cohérence (CI) est défini [98] par:

$$CI = \frac{\lambda_{\max} - n}{n - 1} \quad (6.5)$$

Où n est le nombre d'attributs, et λ_{\max} est la valeur propre principale de la matrice A. λ_{\max} est donnée par:

$$\lambda_{\max} = \frac{1}{n} \sum_{i=1}^n \frac{(Aw^T)_i}{w_i^T} \quad (6.6)$$

Pour vérifier la cohérence, on calcule le rapport: $\frac{CI}{RI}$ dans laquelle RI, appelé l'indice aléatoire, qui est donnée dans le tableau 6.8 par rapport à la dimension n de la matrice considérée [102].

Tableau 6.8- Valeurs de l'indice aléatoire (RI)

n	2	3	4	5	6
RI	0	0.58	0.90	1.12	1.24

Les jugements utilisés pour établir la matrice de comparaison par paires sont considérés comme cohérents si $\frac{CI}{RI} < 0.1$.

Pour l'exemple actuel, la méthode AHP donne une bonne consistance pour la matrices A. Il n'ya pas de cohérence pour les matrices B car leur dimension est 2. En se référant au tableau 6.8 nous avons RI = 0.

- Comparez les alternatives et extraire la meilleure solution: L'évaluation de la fonction objective globale F_k pour chaque alternative est estimée. La meilleure alternative (F_{opt}) est sélectionnée sur la base des critères d'optimisation: $F_{opt} = \underset{k}{\text{opt}}(F_k)$

Étant donnée la matrice A , B^i , et les critères d'optimisation, l'application de cette technique est simple. Les critères d'optimalité consistent à minimiser CT, TT, NI et de maximiser EU. Ainsi, B^i doit être transposée dans le but d'utiliser des critères de minimisation pour toutes les mesures de l'objectif global. Les fonctions objectives globales adaptées à ce problème pour les deux alternatives sont respectivement: $F_1 \cong 0.4$ et $F_2 \cong 0.6$.

- La sélection de la meilleure réalisation de la tâche: La méthode AHP nous permis de choisir la meilleure solution sur la base des mesures et critères définis. La meilleure solution correspond à la première réalisation de la tâche qui est la commande basée sur le pointage sur image suivie par le mode télémanipulation.

6.5.2 Exemple 2 (Robot à 4 câbles)

De la même manière, nous présentons le modèle AHP (Figure 6.11) afin d'évaluer la meilleure combinaison et séquence de modes de commande pour la même opération avec un robot à câbles. Dans cet exemple, l'opérateur effectue la même tâche de deux façons. La tâche souhaitée consiste à écrire deux symboles à deux endroits différents. Le premier élément est un symbole circulaire dessiné dans la position "A". Le deuxième élément est un symbole rectangulaire dessiné dans la position «B».

Dans la première séquence de combinaison de mode, l'opérateur utilise le mode de pointage sur image pour diriger l'effecteur final du robot à la position "A". Lorsque l'effecteur du robot atteint cette position, l'opérateur utilise le joystick pour dessiner le symbole circulaire. Ensuite, dans l'étape suivante, avec le même mode de commande (pointage sur image), l'effecteur final est dirigé vers la position B, où l'opérateur au moyen de la manette joystick, dessine le symbole rectangulaire.

Dans la seconde séquence de combinaison de mode, l'opérateur utilise seulement la commande basée sur le pointage sur l'image pour diriger l'effecteur final du robot à la position "A". Lorsque le robot atteint cette position, l'opérateur utilise à nouveau la commande basée sur le pointage sur image pour dessiner le symbole circulaire. Puis, à l'étape suivante, avec le même mode de contrôle, l'effecteur final est dirigée vers le point B où l'opérateur au moyen de ce mode dessine le symbole rectangulaire. (Figure 6.12).

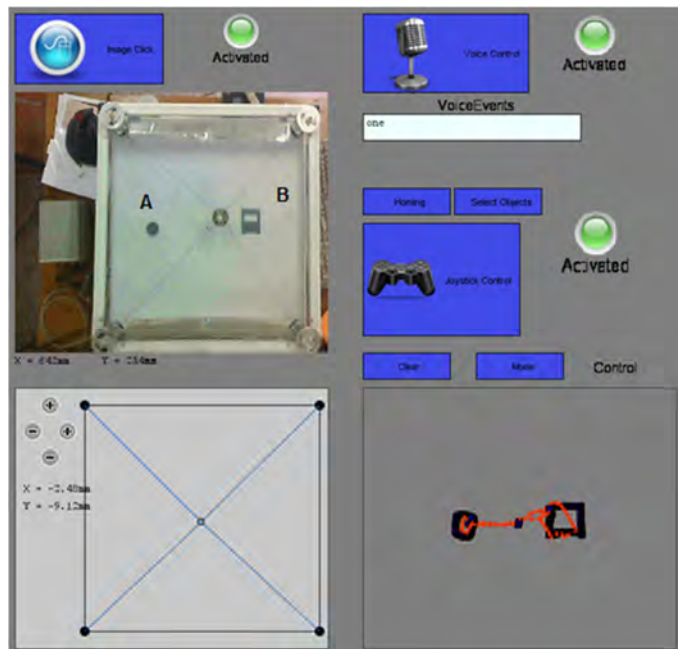


Figure 6.12-Une expérience illustrative

Les résultats obtenus pour les deux réalisations de la tâche sont présentés dans le tableau 6.9.

Tableau 6.9- Valeurs des différents attributs (critères) pour différentes alternatives

	CT(sec)	TT(sec)	NI	EU
Expérience 1	40	5	6	6
Expérience 2	35	4	8	2

De la même façon nous présentons les différentes valeurs d'évaluation des attributs dans les tableaux suivants :

Tableau 6.10- Evaluation pour les quatre attributs

	CT	TT	NI	EU	Poids
CT	1	4	5	6	0,598
TT	1/4	1	2	3	0.203
NI	1/5	1/2	1	2	0,123
EU	1/6	1/3	1/2	1	0,076

Tableau 6.11- Valeurs d'évaluation pour le temps d'achèvement

	V&T	V	Poids
V&T	1	1.142	0.533
V	0.875	1	0.467

Tableau6.12- Valeurs d'évaluation pour le temps de télémanipulation

	V&T	V	Poids
V&T	1	1.250	0.556
V	0.800	1	0.444

Tableau6.13- Valeurs d'évaluation pour le nombre d'interaction

	V&T	V	Poids
V&T	1	0.750	0.589
V	0.333	1	0.411

Tableau6.14 Valeurs d'évaluation pour la facilité d'utilisation

	V&T	V	Poids
V&T	1	3	0,750
V	1/3	1	0,250

De la même manière, nous avons appliqué la méthode AHP pour l'évaluation de l'exécution des tâches dans le cas de notre robot à câbles. En suivant les mêmes étapes que pour l'exemple 1, la fonction objective globale adaptée à ce problème pour les deux alternatives sont respectivement $F_1 \cong 0.6$ et $F_2 \cong 0.4$. En conclusion, la meilleure solution correspond à la seconde réalisation de la tâche qui est l'utilisation du mode basé sur la vision suivie encore du mode basé sur la par la vision. En personnalisant les coefficients des matrices de comparaison A et B, cette technique devient utilisateur-adaptatif.

Cette adaptation de la méthode AHP nous donne la possibilité non seulement de choisir quelle stratégie de combinaison de modes est la mieux adaptée pour une tâche donnée avec un opérateur donné. Elle nous donne aussi la possibilité de :

- Déterminer l'opérateur le mieux adapté pour l'accomplissement d'une tâche donnée en utilisant une stratégie de commande et avec un système expérimental donné.
- Déterminer quel est le système expérimental le mieux adapté pour une tâche donnée effectué par un opérateur donné en utilisant une stratégie de combinaisons des modes.

Dans l'exemple d'application de la figure 6.13, la méthode AHP est utilisée pour le choix de l'utilisateur le mieux adapté pour une tâche/mission donnée. Dans cet exemple, nous avons deux

exécutions de la même tâche menées par deux opérateurs différents (Opérateur1, Opérateur2) en utilisant le même système expérimental. Les opérateurs utilisent dans ce cas la même stratégie de combinaison des modes de commande (M1, M2, ..., Mn) pour accomplir la mission par l'intermédiaire de l'interface multimodale.

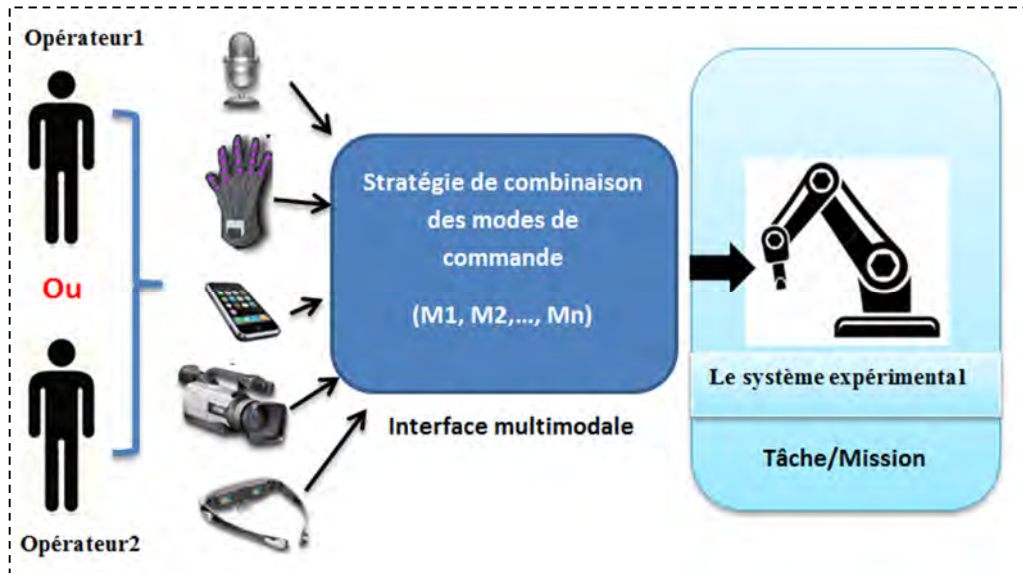


Figure 6.13- AHP pour le choix de l'opérateur

Concernant l'exemple de la figure 6.14, la méthode AHP est utilisée pour le choix du système expérimental le mieux adapté pour la bonne exécution d'une tâche/mission donnée. Dans ce cas, nous avons deux exécutions de la même tâche menées par deux systèmes expérimentaux différents : Système1 (robot sériel), Système2 (robot parallèle à câbles). L'opérateur utilise dans ce cas la même stratégie de combinaison des modes de commande (M1, M2, ..., Mn) pour accomplir la mission, par l'intermédiaire de l'interface multimodale.

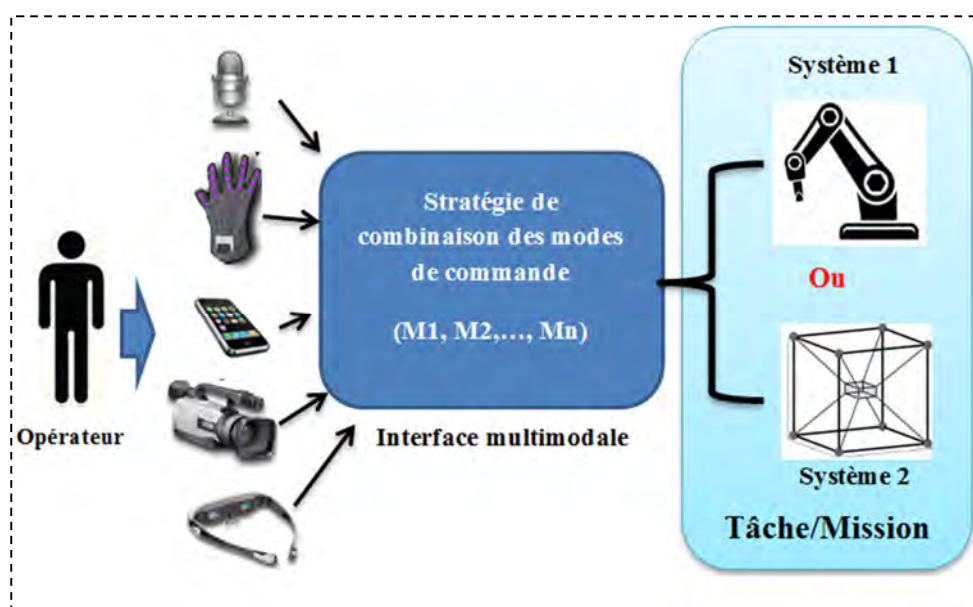


Figure 6.14- AHP pour le choix du système expérimental

Ces différentes propriétés d'adaptations nous donnent la possibilité de concevoir non seulement des interfaces multimodales flexibles et reconfigurables, mais aussi de rendre le système interactif global flexible et reconfigurable tout en garantissant leur **adaptabilité optimale** par rapport à la stratégie de combinaison des modes, au système expérimental utilisé et même à l'opérateur.

6.6 Conclusion

Sur la base de notre analyse et conception de différents SCISs à structure multimodale; nous avons constaté combien l'évaluation des performances de ces systèmes lors de l'exécution des tâches complexes se révèle être une question difficile. En effet, pour réaliser une tâche donnée, nous sommes confrontés au problème de la sélection de la meilleure séquence de modes de contrôle. En outre, l'évaluation des performances du système dans l'exécution des tâches dépend aussi de facteurs humains qui sont inhérents à l'intervention humaine. Néanmoins, malgré la complexité de l'SCIS, la complexité des tâches à effectuer, et la particularité des opérateurs humains, nous avons pu définir des mesures de performances objectives et subjectives par rapport à l'efficacité de la tâche et à la charge de travail soutenue par l'opérateur humain.

Afin d'évaluer les performances des SCIS; nous avons identifié une méthode mathématique d'optimisation appelée "Analytical Hierarchy Process" qui permet de fusionner des critères de performances objectifs et subjectifs en vue de déduire une meilleure alternative parmi plusieurs.

Dans ce chapitre, nous avons présenté comment nous avons adapté cette méthode d'évaluation et de décision pour évaluer les performances de notre système global. Elle a été testée sur plusieurs systèmes notamment pour un robot à câbles et pour un robot manipulateur. Cette méthode nous a fourni effectivement des stratégies pour développer des interfaces multimodales adaptables et indépendantes par rapport à l'utilisateur, au système, aux modes de commande et à l'environnement. Cette méthode peut-être aussi utilisée à la fois comme technique d'évaluation et/ou de décision, pour rendre l'interaction flexible en termes de degré d'autonomie du système global.

Conclusion générale

Récemment, certaines recherches dans le domaine de l'interaction homme-machine s'orientent vers l'étude des systèmes supervisés qui se caractérisent par l'intégration de plusieurs modes de commandes (systèmes multimodaux). Ces systèmes sont destinés à effectuer des tâches et des missions complexes dans des environnements qui peuvent être dynamiques et parfois hostiles. Des exemples de tels modes de commandes sont la commande par la voix, la commande par télémanipulation au moyen d'un joystick, la commande basée sur l'image, la commande gestuelle, la commande physiologique.

L'interaction multimodale est celle qui offre au superviseur plusieurs modes de commande pour lui permettre d'interagir avec un système (robot) de façon aisée pour accomplir des tâches et des missions plus ou moins complexes. Durant la supervision de la tâche, l'interaction multimodale permet à l'opérateur de sélectionner le mode qui lui semble le plus adapté à la situation en cours pour conduire au mieux la tâche en tenant compte des performances du système robotique à superviser.

L'analyse de l'état de l'art dans l'interaction des systèmes multimodaux nous a permis de conclure qu'il existe un problème de méthodologie pour la conception et le développement de ces systèmes, dont la difficulté se retrouve dans la complexité de ces systèmes ainsi que l'intervention de plusieurs paramètres qualitatifs et quantitatifs. Nous avons remarqué que les chercheurs dans ce domaine ne peuvent pas concevoir leurs expériences selon des méthodologies, les relier au série de travaux précédents ; et surtout sur la façon de faire la recherche, par exemple, d'appliquer ou non des méthodes statistiques quantitatives nécessitant des expériences à grande échelle, ou à poursuivre une approche qualitative des études à petite échelle concentrant sur des études pour un cas précis. Et que la majorité de ces travaux de recherche sont limités à l'utilisation dans la mesure maximale de deux modalités d'interaction, dont le plus souvent la commande vocale avec d'autre modalité.

D'autre part, nous avons remarqué que la plupart des systèmes supervisés multimodaux restent fortement tributaires du système robotique réalisé, du superviseur et de la tâche ou de la mission pour laquelle ils ont été initialement conçus. Et que la complexité du système

multimodal ainsi que la complexité des tâches à accomplir génèrent de nouveaux problèmes incluent et font interférer principalement les éléments suivants:

- La nature et le nombre de modes disponibles, la conception de l'interface utilisateur et sa capacité de gestion de l'interaction homme-robot plus ou moins aisée pour la sélection et l'enchaînement des différents modes,
- Les facteurs humains (influence du degré d'apprentissage et d'expérience de l'opérateur humain, préférence et motivations du choix de certains modes et de leur enchaînement par l'opérateur, l'état mental de l'opérateur dans des circonstances habituelles ou critiques, ..),
- Les performances du système opérationnel à conduire (à savoir son degré d'intelligence, d'autonomie et de coopération avec le superviseur),
- Le degré de complexité de la tâche ou de la mission ainsi que celui de l'environnement.

Dans l'objectif de faire face à ces problèmes, et pour analyser et tester ces systèmes sur des prototypes génériques en vue de dégager quelques règles de conceptions qui permettront d'améliorer leur adaptation; Nous avons planifié la stratégie d'étude suivante:

La conception d'une architecture multimodale générique et son illustration par la réalisation effective au moins de deux prototypes expérimentaux intégrant quelques modes de commandes; d'effectuer une analyse critique des différents modes implémentés individuellement; ensuite, d'étudier leur combinaison pour conduire des tâches typiques de complexité croissante. Enfin, de dégager des conclusions sur les meilleures correspondances et couplages entre modes de commandes en fonction de la conception du système robotique et de son degré d'autonomie et de coopération, des facteurs humains, de la complexité de la tâche et de l'environnement. Et à la fin, de proposer une méthode mathématique pour l'évaluation et l'analyse de ces systèmes.

Conformément au programme de travail prédéfini, nous avons réalisé trois prototypes expérimentaux: un robot à 3 câbles, un robot à 4 câbles et un robot mobile avec un bras manipulateur à 5ddl; Nous notons que la réalisation et l'utilisation du robot à 4 câbles dans une classe spéciale pour un centre de réhabilitation est une contribution dans ce domaine; tout en exploitant plusieurs avantages pour ce type de robots: un grand espace de travail en comparaison avec les manipulateurs parallèles classiques; En plus, les robots à base de câbles ont une structure légère qui a de bonnes propriétés dynamiques; un rapport masse/ charge utile élevée et un coût de construction réduit; une bonne portabilité; une structure facile à monter, démonter, stocker et transporter.

Une interface multimodale a été développée pour permettre à l'opérateur d'interagir avec ces prototypes expérimentaux en utilisant et en combinant plusieurs modes de commande. Les modes de commande développés sont des modes de commande haut niveau. Ils incluent la commande basée sur image ou "point and click", la commande vocale, la commande gestuelle et la commande joystick.

Dans le but d'extraire une méthodologie d'analyse de l'interaction multimodale homme-robot, nous avons adopté en premier lieu une première approche expérimentale; qui nous a permis d'effectuer une analyse indépendante de chaque mode de commande. Et qui nous a permis aussi de mettre en évidence les caractéristiques ainsi que les possibilités et les limites d'utilisation de chaque mode tenant compte du système utilisé et de l'environnement de travail. Elle a permis également de mettre en évidence certains facteurs qualitatifs et quantitatifs relatif au facteur humain qui influent sur les performances du système.

Ensuite, pour prendre en charge des missions et des tâches plus ou moins complexes qui nécessitent la multimodalité; une analyse des combinaisons possibles des différents modes a été menée. Dans le but d'établir une relation entre le degré de complexité de la tâche, le degré de complexité de l'environnement ainsi que le nombre de modes disponibles, nous avons étudié deux types d'environnements de travail: un environnement simple et un environnement complexe.

Ainsi que, cette approche expérimentale nous a permis d'analyser les relations entre les différents modes de commande, le degré de complexité de l'environnement, la complexité de la tâche, le nombre de modes disponibles. D'autre part, nous avons remarqué l'influence du facteur humain qui intervient comme facteur qualitatif (l'expérience de l'opérateur, les préférences, ..). Divers scénarios ont été réalisés combinant différents modes de commandes sur différents systèmes avec différents opérateurs et différents environnements. La variabilité des différents éléments (opérateurs, interface; environnement, système utilise) nous a suggérer de relever le niveau d'abstraction de l'interface multimodale qui permet sa réutilisabilité ainsi que son adaptation.

En second lieu; pour résoudre le problème d'un point de vue méthodologique, nous avons adapté une méthode d'évaluation et de décision permettant d'évaluer les performances de notre système global. Cette approche nous a fourni les stratégies pour développer des interfaces multimodales adaptables et indépendantes par rapport à l'utilisateur, au système, aux modes de commande et à l'environnement. Cette méthode est appelé « Analytic Hierarchy Process » « AHP », qui a la particularité de combiner des facteurs qualitatifs et quantitatifs et

donc nous avons pu combiner les deux approches d'analyse dans ce domaine, la méthode qualitative et la méthode quantitative. Cette méthode peut-être aussi utilisée comme technique d'évaluation et/ou de décision, pour rendre l'interaction flexible en terme de degré d'autonomie du système globale.

A la fin, nous pouvons diviser notre contribution dans ce travail en trois parties: Une contribution concernant la réalisation d'un nouveau système robotisé qui est le robot parallèle à câbles contrôlé par des commandes de haut niveau pour qu'il soit utilisé dans un centre de réhabilitation. Ensuite, l'utilisation des nouvelles méthodes d'implémentation pour les techniques d'interaction homme-machine. Et enfin, la proposition d'une nouvelle méthode d'analyse basée sur les deux approches dans ce domaine, l'approche qualitative et quantitative ; dans le but de créer des interfaces multimodales dynamiquement configurables (qui tient en compte l'état en temps réel de l'environnement, la tâche, l'utilisateur ou le système utilisé) selon un modèle de création généralisé toute en travaillant dans un niveau d'abstraction supérieur (orienté mode, tâche ou système).

Bibliographie

- [1] Bellik Y. (1995)., “Interfaces multimodales, concepts, modèles et architectures”, Thèse université d’Orsay.
- [2] Chapelier L. (1996). Dialogue d'assistance dans une interface homme-machine multimodale, thèse, CRIN, Nancy.
- [3] Bos E. ,Huls C. ; Claassen W. “ EDWARD : full integration of language and action in a multimodal user interface”, *International Journal of Human-Computer Studies*, vol. 40, no. 3, p. 473-496, 1994.
- [4] Caelen, J. “Speech and Multimodal Interface: the case of ICPdraw”, *ESCA workshop*, Lautrach, Septembre 1993.
- [5] Nigay, L. and Coutaz, J. A “Generic Platform for Addressing the Multimodal Challenge”, *Proceedings Of CHI'95*, pp 98-105.1995.
- [6] Cohen P. R, “The role of natural language in a multimodal interface”, *UIST'92*, pp. 143-149, 1992.
- [7] Condom J. M, “Un système de dialogue multimodal pour la communication avec un robot manipulateur”, thèse, Toulouse 1992.
- [8] Coutaz J. & al. “Four easy pieces for assessing the usability of multimodal interaction: the CARE properties”, *Interact'95*, Lillehammer, Norway, 1995.
- [9] Coutaz J. “Multimedia and multimodal user interfaces: a taxonomy for software engineering research issues”, *EWCHI*, St Petersburg, 1992.
- [10] Lian. C, “Etude de l'usage de la parole dans les interfaces multimodales”, thèse, Institut National Polytechnique de Grenoble I.N.P.G, 1998.
- [11] Coutaz. J et Caelen. J, “A taxonomy for multimedia and multimodal user interface”. *Proceedings 1st ERCIM Workshop on Multimodal Human-Computer Interaction*, INESC: 00143–148, 1991.
- [12] Ronald M. Baecker, “Readings in Human-Computer Interaction: Toward the Year 2000”, livre, Morgan Kaufmann - 1995 (ISBN 9781558602465).

-
- [13] Van Dam. A, "Post-wimp user interfaces". *Communications of the ACM*, vol. 40, no. 2, pp. 0063–67, 1997.
- [14] Yannis Vassiliou, "Human Factors and Interactive Computer Systems", *Proceedings of the NYU Symposium on User Interfaces*, , vol. 1982, Intellect Books - 1984 (ISBN 9780893911829). New York, May 26-28, 1982.
- [15] Michel. N, Christine. P, "La communication homme-machine en robotique: analyse comparée de différentes interfaces de programmation", *TIS*, vol.3, n°2-3, 1991.
- [16] http://www-clips.imag.fr/geod/User/jean.caelen/Publis_fichiers/SystemeMultimodal.pdf
- [17] Anas Fattouh et al, "Force feedback joystick control of a powered wheelchair: preliminary study", *IEEE International Conference on Systems, Man, and Cybernetics* 2003.
- [18] Haton J.P., "Reconnaissance automatique de la parole", *8-ième école d'été d'informatique de IIAFCET*, Namur 1975.
- [19] Liénard O.S, "Les processus de la communication parlée", Masson, Paris, 1977
- [20] Duda R.O., Hart P.E., "Pattern classification and scene analysis", John Willey, New York, 1973.
- [21] Fu K.S., "Digital pattern recognition", *Springer-Verlag*, New York, 1980.
- [22] Tou J.T. and Gonzales, R.C. "Pattern Recognition Principles", *Addison-Wesley*, 1974.
- [23] Abdelatif Mokeddem, "analyse de la parole: reconnaissance multilocuteur de mots isolés pour les systèmes miniaturisés", Thèse, *université de neuchâtel Institut de microtechnique*, 1985.
- [24] cf. techniques de l'ingénieur, vol. H1 940, p.3
- [25] Levinson S. et Liberman H., "La reconnaissance de la parole par ordinateur, dans L'intelligence de l'informatique", *bibliothèque pour la science*, diffusion Belin, 1984.
- [26] Haton J.P, "Reconnaissance automatique de la parole", *8-ième école d'été d'informatique de IIAFCET*, Namur 1975.
- [27] Dragon Naturally Speaking <http://www.scansoft.com/naturallyspeaking>
- [28] Microsoft Speech SDK 5.1 <http://www.microsoft.com/speech/download/sdk51>
- [29] CMU Sphinx <http://cmusphinx.sourceforge.net>
- [30] FreeTTS <http://freetts.sourceforge.net>

- [31] <http://voce.sourceforge.net/files/VoceWhitePaper.pdf>
- [32] Cadoz C., “Le geste canal de communication homme/machine – la communication « instrumentale »”, *Techniques et Science Informatiques*, vol.13, pp.31-61, 1994.
- [33] LRP. Rapport d’activités 1990-1993 Laboratoire de Robotique de Paris, Juin 1993.
- [34] Foley, “Les communication entre l’Homme et l’Ordinateur”, *Pour la science*, Décembre 1987.
- [35] Ellis S. R. “What are Virtual Environments?”, *IEEE Computer Graphics & Applications*, pp.17-22, Janvier 1994.
- [36] Latta J. N. et Oberg D. J., “A conceptual Virtual Reality Model”, *IEEE Computer Graphics & Applications*, pp. 23-29, Janvier 1994.
- [37] Gibet S. “Codage, représentation et traitement du geste instrumental – Application à la synthèse de sons musicaux par simulation de mécanismes instrumentaux”, Thèse, INPG Grenoble.
- [38] Cassell J., Pelachaud C., Badler N., Steedman M., Achorn B., Becket T., Douville B., Prevost S. et Stone M. “Animated conversation : Rule-based Generation of Facial Expression, Gesture & Spoken Intonation for Multiple Conversational Agents”, *Actes de SIGGRAPH’94*, Orlando, USA.1994.
- [39] Lee J. et Kunii T. L. “Computer Animated Visual Translation From Natural Language to Sign Language”, *The journal of visualization and computer animation*, vol. 4, pp. 63-78. 1993.
- [40] Iwai Y., Watanabe K., Yagi Y. et Yachida M. “Gesture Recognition Using Colored Gloves”, in *the13th International Conference on Pattern Recognition (ICPR ’96)*, vol .1, pp 662-666, Vienna, Austria, Aout 1996.
- [41] Hienz H., Grobel K. and Offner G. “Real-Time Hand Arm Motion Analysing using a Single Video Camera”, *Proceedings of Gesture Workshop’96*, Univ.Of York, pp. 323-327, 1996.
- [42] Berard F., Couteaz J., et Crowley J. L. “Finger Tracking as Input Device for Augmented Reality”, *Proceedings of International Workshop on Automatic face and Gesture-Recognition*, Zurich, Switzerland, June 1995.
- [43] T . F. Cootes, C. J .Taylor, D .H . Cooper and J . Graham, “Active Shape Models : their Training and Application”, *Computer Vision and Image Understanding*, vol . 61, n 1, p .38-59, 1995.

- [44] J . Martin and J .L . Crowley, “An Appearance-Based Approach to Gesture-Recognition”, *Proceedings of the 9th Conf. on Image Analysis and Processing*, Italy, 1997.
- [45] Master Thesis: B .Dorner, “Chasing the color glove: visual hand tracking”, Simon Fraser University, 1994.
- [46] J .M .Regh and T. Kanade, “Visual Tracking of Height DOF Articulated Structures: an Application to Human Hand Tracking”, *Proceedings of 3'd ECCV*, vol 2, 1994, pp . 37-46.
- [47] J .J .Kuch, and T. S . Huang, “Vision Based Hand Modeling and Tracking for Virtual Teleconferencing and Tele-collaboration”, *Proceedings of fifth International Conference on Computer Vision*, MIT, Cambridge, Massachusetts, pp .666-671, June 20-23, 1995.
- [48] T .Baudel and M .Beaudouin-Lafon, “Charade : Remote Control of objects using Free-Hand Gestures”, *Communications of the ACM*, Vol. 36, no 7, p . 28-35, 1993.
- [49] Y. Bellik, “Modality Integration : Speech and Gesture”, *Survey of the State of the Art in Human Language Technology*, , Chapter 9: MultiModality, Cambridge University Press, ISBN 0-521-59277-1 Hardback, 1997.
- [50] A. Braffort, “Reconnaissance et Compréhension de gestes, application à la langue des signes”, Thèse, Université de Paris XI, spécialité informatique, juin 1996.
- [51] Van Sommers P., “Drawing and Cognition. Descriptive and experimental studies of graphic production processes”, Cambridge University Press, 1984.
- [52] LEBAHAR J-C., “Le dessin d'architecte : simulation graphique et réduction d'incertitude”, Marseille, Editions Parenthèses, 1983.
- [53] Landay J.A., Myers B.A., ”Sketching Interfaces: Towards More Human Interface Design”, in *IEEE Computer*, Vol. 34, N. 3, Mars, p. 56-64, 2001.
- [54] M. Seelinger, M. Robinson and S.B. Skaar, “A Vision-Guided, Semi-Autonomous System Applied to a Robotic Coating Application”, *Sensor Fusion and Decentralized Control in Robotic Systems. Proc. SPIE 3209*, Pittsburgh, PA, pp. 133-144, Oct. 1997.
- [55] A. Saffiotti, D. Driankov, and T. Duckett., “A System for Vision Based Human-Robot Interaction”, *Proc. of the IEEE Int. Workshop on Safety, Security, and Rescue Robotics (SSRR)*. Bonn, DE, May 2004.

- [56] Dan Stoianovici, Robert Webster, Louis Kavoussi, M.D., “Robotic Tools for Minimally Invasive Urologic Surgery”, *Chapter in Complications of Urologic Laparoscopic Surgery: Recognition, Management and Prevention*, Dec 2002.
- [57] P. Velagapudi, S. Owens, P. Scerri, M. Lewis and K. Sycara, “Environmental Factors Affecting Situation Awareness in Unmanned Aerial Vehicles”, *In Proceedings of AIAA Infotech*, 2009.
- [58] Mansoor M. Nasir and Shi-Yin Qin, “Investigation of Human Factors in UAV Accidents Based on Analysis of Statistical Data”, *in the Proceedings of the First International Conference on Instrumentation, Measurement, Computer, Communication and Control*, pp. 1011-1015, October 2011.
- [59] Ebken. J, Bruch. M, Lum. J, “Applying unmanned ground vehicle technologies to unmanned surface vehicles”, *In Proceedings of SPIE 5804: Unmanned Ground Vehicle Technology VII*, Orlando, Florida, USA. (2005).
- [60] Weisbin, R. C., Rodriguez, G., Schenker, S. P., Das, H., Hayati, S., Baumgartner, T. E., , “Autonomous Rover Technology for Mars Sample Return” ,*Artificial Intelligence, Robotics and Automation in Space, Proceedings of the Fifth International Symposium, ISAIRAS '99*, held 1-3 June, 1999.
- [61] Kerstin Dautenhahn “Methodology & Themes of Human-Robot Interaction: A Growing Research Field”, *International Journal of Advanced Robotic Systems*, Vol. 4, No. 1 ISSN 1729-8806, pp. 103-108, 2007.
- [62] Leonardo Giusti and Patrizia Marti, “Interpretative Dynamics in Human Robot Interaction”, *The 15th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN06)*, Hatfield, UK, September 6-8, 2006.
- [63] Kazuyoshi Wada and Takanori Shibata “Robot Therapy in a Care House - Results of Case Studies”, *The 15th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN06)*, pp. 581-586, Hatfield, UK, September 6-8, 2006.
- [64] Jan Frederik Maas, Thorsten Spexard, et al, ”BIRON, what’s the topic?A Multi-Modal Topic Tracker for improved Human-Robot Interaction”, *The 15th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN06)*, Hatfield, pp. 26-32, UK, September 6-8, 2006.
- [65] Sheridan T. B, “Tele-robotics, Automation, and human supervisory control”. *MIT Press*, 1992.

- [66] Backes P.G., Beahan J. and Bon B, "Interactive command building and sequencing for supervised autonomy", *Proc. IEEE Int. Conf. on Robotics and Automation*, Atlanta, pp.795-801, 1993.
- [67] Backes P. G, TsaoK. S, "UMI: an interactive supervisory and shared control system for telerobotics", *in: Proc. of IEEE Int. Conf. on Robotics and Automation*, Cincinnati, Ohio, pp.1096-1101, 1990.
- [68] Hirzinger, Heindl J, Landzettel K. and Brunner B , "Multisensory shared autonomy- a key issue in the space robot technology experiment ROTEX", *in: Proc. of IEEE/RSJ, Int. Conf. on Intelligent Robots and Systems*, Raleigh, NC, pp.221-230, 1992.
- [69] Zaatri A ,VanBrussel H, "An integrated multi-modal control architecture for Tele-autonomous Systems", *in: Proc. IAS-5*, Sapporo, Japan, pp.775-783, 1998.
- [70] Laboratoire de robotique université de laval.
- [71] E. Ottaviano, M. Ceccarelli, M. Toti, C. Avilacarrasco "CaTraSys (Cassino Tracking System): A Wire System for Experimental Evaluation of Robot Workspace", *journal of robotics and mechatronics*, Vol.14, pp78-87, 2002.
- [72] Surdilovic D. and Bernhardt R, "STRING-MAN: a New Wire-Robot Gait Rehabilitation", *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA)*, New Orleans, 2004, pp. 2031-2036, 2004.
- [73] Parallel Robot Projects at Ohio University .Robert L. Williams II. Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators October 3-4, 2002, Quebec City, Canada.
- [74] Pott, A. ; Fraunhofer IPA, Stuttgart, Germany ; Mutherich, H. ; Kraus, W. ; Schmidt, V, "Cable-driven parallel robots for industrial applications: The IPAnema system family", *Robotics (ISR), 2013 44th International Symposium on* ,pp. 1 – 6, 2013.
- [75] Rosati G., Gallina P., Masiero S., and Rossi A., "Design of a New 5 D.O.F. Wire-Based Robot for Rehabilitation", *Proceedings of the 2005 IEEE 9th International Conference on Rehabilitation Robotics*, Chicago, 2005, pp. 430-433, 2005.
- [76] <http://robot.gmc.ulaval.ca/fr/recherche/theme201.html>
- [77] Robert L. Williams II and Paolo Gallina, "Planar Cable-Direct-Driven Robots, Part I: Kinematics and Statics ", *ASME Design Technical Conferences 27th Design Automation Conference*, September 9-12, 2001, Pittsburgh, PA.

- [78] Robert L. Williams II and Paolo Gallina and Aldo Rossi, "Planar Cable-Direct-Driven Robots, Part II: Dynamics and Control", *ASME Design Technical Conferences 27th Design Automation Conference September 9-12, 2001*, Pittsburgh, PA.
- [79] <http://www.ni.com/tutorial/7109/fr/>
- [80] <http://arduino.cc/en/Main/ArduinoBoardMegaADK?from=Main.ArduinoBoardADK>
- [81] Rabiner L.R. and Sambur M.R., "An algorithm for determining the endpoints of isolated utterances", *Bell System Technical Journal*, vol. 54, no. 2, pp. 297-315, 1975.
- [82] Lamel L.F., Rabiner L.R., Rosenberg A.E. and Wilpison J.G., "An Improved endpoint detector for IWR", *ASSP-Z9*, 1981.
- [83] Stanley Smith Stevens, John Volkman et Edwin B. Newman, "A scale for the measurement of the psychological magnitude pitch", *Journal of the Acoustical Society of America*, vol. 8, no 3, p. 185-190, 1937.
- [84] MFCC Fang Zheng, Guoliang Zhang and Zhanjiang Song, "Comparison of Different Implementations of MFCC", *Computer Science & Technology*, Vol. 16, no. 6, pp. 582-589, 2001.
- [85] MacQueen, J. B., "Some Methods for classification and Analysis of Multivariate Observations", *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability I. University of California Press*. pp. 281-297.1967.
- [86] E.W. Forgy, "Cluster analysis of multivariate data: efficiency versus interpretability of classifications", *Biometrics*, Vol. 21, pp. 768-769, 1965.
- [87] Linde, Y., Buzo, A., and Gray, R. M., "An Algorithm for Vector Quantizer Design", *IEEE Transactions on Communications COM-28*, pp. 84-95, 1980.
- [88] M. Oussalah, A. Zaatri and H. VanBrussel, "Kalman Filter Approach for Lane Extraction and Following", *Journal of Intelligent and Robotic Systems*, vol. 34,no. 2, pp 195-218, June 2002.
- [89] Z. Zhang, "A flexible new technique for camera calibration", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11,pp 1330-1334, 2000.
- [90] R.Y. Tsai, "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Miami Beach, FL, pp. 364-374, 1986.
- [91] La méthode CCU Greenbaum & Kyng (eds): Design At Work - Cooperative design of Computer Systems, Lawrence Erlbaum 1991.

- [92] http://www.iso.org/iso/catalogue_detail.htm?csnumber=52075
- [93] McLean G.F, Prescott B, Podhorodesky R, “Tele-operated system performance evaluation”, in: *IEEE trans. on Systems, Man, and Cybernetics*, vol.24, n. 5, pp.796-804, 1994.
- [94] Jones P. M, Chu R. W, Mitchell C. M., “Human-Computer Co-operative Problem Solving: theory, design, and evaluation of an intelligent associate system”, in: *IEEE trans. on Systems, Man, and Cybernetics*, vol.25, No. 7, pp. 1039-1052, 1995.
- [95] Skubic M. et al., ”A Tele-robotics Construction Set with Integrated Performance Analysis”, in: *Proc. of the IEEE/RSJ. Intl. Conf. on Intelligent Robots and Systems*, vol. 3, Pittsburg, PA, pp.20-26, 1995.
- [96] Kai Z. Emmanuel C. and Dongqing S., “Centralized and Distributed Task Allocation in Multi-Robot Teams via A Stochastic Clustering Auction”, *ACM Transactions on Autonomous and Adaptive Systems*, Vol. 7, Issue 2, Article 21, pp. 21:1-21:22, July 2012.
- [97] Kai Z. Emmanuel C. and Dongqing S., “Centralized and Distributed Task Allocation in Multi-Robot Teams via A Stochastic Clustering Auction”, *ACM Transactions on Autonomous and Adaptive Systems*, Vol. 7, Issue 2, Article 21, pp. 21:1-21:22, July 2012.
- [98] Saaty T. L., “How to make a decision: the Analytical Hierarchy Process”, *European Journal of Operation Research*, 48, 9-26, 1990.
- [99] Doctoral thesis: Adelouahab Zaatri., “Investigations into ISCS”, K.U.Leuven. June 2000.
- [100] Sumie Y, Mio O, Takako. N and Tomohiro H., “GUI for AV remote controllers based on analytic hierarchy process in consideration of human error”, *The 1st IEEE Global Conference on Consumer Electronics*.pp.177-181., 2012.
- [101] Jonathan F. Bard., “evaluating Space Station Applications of Automation and Robotics”, *IEEE Transactions on Engineering Management*.Vol.EM-33. NO. 2. pp.102-111., MAY.1986.
- [102] Jeffrey. T, Chuan. T, Tamio. A., “Analytic Evaluation of Human-Robot System for Collaboration in Cellular Manufacturing System”, *IEEE/ASME International Conference on Advanced Intelligent Mechatronics Montréal, Canada*.pp.515-520., July 6-9. 2010.
- [103] Changwon K, Reza L., “Analytical Hierarchy Process and Brain Limbic System Combined Strategy for Mobile Robot Navigation”, *IEEE/ASME International Conference on Advanced Intelligent Mechatronics Montréal. Canada*. pp. 968-972, July 6-9. 2010.
- [104] Van P. de Poel, Witvrouw W, Bruyninckx H, DeSchutter J., “An environment for developing and optimising compliant robot motion tasks”, in: *International Conference*

- on Advanced Robotics*. Tokyo.pp.738-741, 1993.
- [105] Zaatri A, VanBrussel. H., “Investigations in tele-robotics using co-operative supervisory modes of control”, in: *Proc. of the SPIE Conf*, Pittsburg, PA.pp-41-52, 1997.
- [106] Master’s thesis: Viljoen J et al, “Force controlled robot programming using natural language”, *dept of computer science. Katholieke University Leuven*, Belgium, 1997.
- [107] Bouchemal B, Zaatri A., “Gestural and Image-Based Control Combination for rehabilitation applications using cable-based robots”, *First international conference on technology for helping people with special needs*, Riyadh, KSA.pp.121-125, February 18-20, 2013.
- [108] Bouchemal B and Zaatri A. “Image-based Control for Cable-based Robots”, *International Journal of Control, Automation, and systems*, vol 12,n 01. pp 118-125. 2014.
- [109] Graves S., “An environment for compliant motion programming by human demonstration”, in: *Proc. of IEEE/RSJ, Int. Conf. on Intelligent Robots and Systems*. pp.795-801.1996.
- [110] Wang Q, DeSchutter J., “Force controlled robot: Programming by Human Demonstration”, *Internal technical report 95R64*, KULeuven, PMA.1995.
- [111] Adams J. A., “Critical Considerations for Human- Robot Interface Development”, *AAAI Fall Symposium: Human Robot Interaction Technical report FS-02-03*. pp. 1-8.2002.
- [112] Rouse W. B, Edwards S. L, Hammer J. M., “Modelling the dynamics of mental workload and human performances in complex system”, in: *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, n. 6. pp. 1662-1671.1993.
- [113] Ph.D. Thesis: Wei Z. G., “Mental load and performance at different autonomous levels”, *University of Delft, Nederlands*.1996.
- [114] Winston W. L., “Operations research: Applications and algorithms”, *Duxbury Press*.1994.

ANALYSE DE L'INTERACTION HOMME - ROBOT DANS LES SYSTEMES SUPERVISES MULTIMODAUX

Résumé: Le contrôle et la commande des systèmes supervisés locaux ou distants et destinés à la conduite des missions et des tâches complexes requièrent différents niveaux et modes d'interaction avec l'opérateur humain compte tenu des différentes situations rencontrées. Des exemples de tels systèmes se retrouvent dans de nombreux domaines d'applications y compris des domaines critiques comme l'art militaire, le démantèlement de centrales nucléaires, l'exploration spatiale, la télé-chirurgie, etc.

Le sujet est donc relatif à un domaine très complexe de la technologie du futur. Il a pour objectif d'effectuer l'analyse, le développement, l'implémentation ainsi que l'évaluation des performances d'une interface multimodale destinée à assurer une interaction Homme-Robot flexible et adaptable. Cette interface multimodale sera le pivot du système global Homme-Robot et s'inspire de l'interaction naturelle Homme-Homme qui exploite et combine des modes de commande de haut-niveau (paroles, gestes, localisation d'objets par la vision, manipulation manuelle).

L'objectif principal de ce sujet de thèse consiste à effectuer une analyse de l'interaction homme robot dans les systèmes supervisés multimodaux. Il est question d'étudier l'influence de la nature et du nombre de modes disponibles, des facteurs humains, des performances du système opérationnel à conduire, du degré de complexité de la tâche ou de la mission à accomplir. Bien entendu, de part sa propre complexité due à la multiplicité des modes et aux possibilités de leurs combinaisons; le degré de complexité structurelle du système devrait augmenter et mobiliser les multiples compétences sensorielles de l'opérateur. Mais de part la conception adoptée qui est centrée utilisateur et de niveau d'abstraction élevé, la complexité fonctionnelle et opérationnelle du système global sera masquée permettant une flexibilité et une adaptabilité d'utilisation pour assurer de meilleures performances.

Les contributions majeures de cette thèse sont :

- Le développement d'une interface multimodale combinant différents modes de commande de haut niveau: la commande vocale, la commande gestuelle, la commande par pointage sur image (click-and-Point) et la commande manuelle par joystick.
- La conception et la réalisation d'un nouveau système robotisé qui est le robot parallèle à câbles. Du fait de leurs caractéristiques particulières, ces robots ouvrent des axes de recherche pour des applications originales, en particulier dans les domaines de la réhabilitation.
- Le développement d'une application originale qui est l'utilisation d'un robot à 4 câbles dans une classe spéciale pour un centre de réhabilitation et d'apprentissage exploitant la multimodalité.
- Proposition d'une méthode d'évaluation des performances pour les systèmes supervisés multimodaux tenant compte de certains facteurs humains par adaptation d'une méthode d'évaluation appelée Analytical Hierarchy Process (AHP) qui permet de fusionner des critères qualitatifs et quantitatifs.

Mots Clés: Interaction homme-robot, Systèmes multimodaux supervisés, Contrôle haut niveau

ANALYSIS OF THE HUMAN - ROBOT INTERACTION FOR SUPERVISED MULTIMODAL SYSTEMS

Abstract: The monitoring and control of local or remote supervised systems dedicated to carry out missions and complex tasks require different levels and modes of interaction with the human operator when facing different situations. Examples of such systems are found in many application areas including critical areas such as the art of war, the dismantling of nuclear power plants, the space exploration, the télé-surgery, etc.

The subject is related to a very complex area of futurist technology. It aims to carry out the analysis, development, implementation and performance evaluation of a multimodal interface to ensure flexible and adaptable Human-Robot interaction. This multimodal interface will be the pivot of the whole Human-Robot system and is inspired from the natural Human-Human interaction which operates and combines high-level control modes (speech, gestures, locating objects by vision, manual manipulation).

The main objective of this thesis is to conduct an analysis concerning the human robot interaction in multimodal supervised systems. It intends to study the influence of the nature and the number of available modes, the human factors, the performance of the operational system, the complexity of the task or mission. Of course, because of its own complexity caused by the mode multiplicity and by the possibilities of their combinations; the degree of structural complexity of the system is expected to increase and mobilize the multiple sensory skills of the operator. But from the adopted design that is user centered and that uses a high abstraction level; the functional and operational complexity of the overall system will be masked enabling flexibility and adaptability of use ensuring therefore better performance.

The major contributions of this thesis are:

- The development of a multimodal interface combining different high-level control modes: voice control, gesture control, control by pointing on an image (click-and-point) and joystick control.
- The design and implementation of a new robotic system which is the parallel cable-based robot that open lines of research for original applications, particularly in the areas of rehabilitation.
- The development of an original application that uses a 4 cables robot and the multimodality to built up a special class of a rehabilitation and learning center.
- The proposal of a method for performance evaluation of multimodal supervised systems that takes into account some human factors by fitting an evaluation method called Analytical Hierarchy Process (AHP), which allows merging qualitative and quantitative criteria.

Keywords: Human-robot interaction, Supervised multimodal systems, High level of control

تحليل التفاعل بين الإنسان و الآلة في الأنظمة متعددة أنماط التحكم تحت الإشراف

ملخص:

رصد و مراقبة الأنظمة محليا أو عن بعد المخصصة للقيام بمهام و عمليات معقدة تتطلب مستويات و أنماط تحكم مختلفة مع المستخدم البشري الذي تواجهه حالات مختلفة. و توجد أمثلة على هذه الأنظمة في الكثير من مجالات التطبيق بما في ذلك الحساسة منها مثل: فن الحروب، تفكيك محطات الطاقة النووية، استكشاف الفضاء، الجراحة عن بعد، .. الخ.

يتعلق هذا الموضوع بمجال جد معقد من تكنولوجيا المستقبل، حيث يهدف إلى إجراء التحليل، التطوير، والقيام بتقييم أداء الواجهة الرسومية متعددة أنماط التحكم لضمان المرونة و القدرة على تكيف التفاعل بين الروبوت و البشر. هذه الواجهة متعددة أنماط التحكم سوف تكون محور نظام التفاعل روبوت- بشر كله، و التي هي مستوحاة من التفاعل الطبيعي بين الإنسان و الإنسان الذي يعمل و يجمع بين العديد من أنماط التحكم عالية المستوى (الكلام، الإيماءات، تحديد مكان الكائنات عن طريق الرؤية، التحكم اليدوي).

الهدف الرئيسي من هذه الرسالة هو إجراء تحليل للتفاعل بين الإنسان و الآلة في النظم متعددة أنماط التحكم تحت الإشراف. ويأتي لدراسة تأثير طبيعة و عدد الأنماط المتاحة، العوامل البشرية، أداء نظام التشغيل بما في ذلك تعقيد المهام. و بطبيعة الحال، بسبب تعقيدها الناجم عن تعدد الأنماط و إمكانية جمعها، من المتوقع أن يزيد تعبئة المهارات الحسية للمستخدم من التعقيد الهيكلي للنظام. لكن باعتماد التصميم المتمحور حول المستخدم على مستوى عال من التجريد، تمكنا من حجب التعقيد الوظيفي و التشغيلي للنظام العام مما يتيح المرونة و القدرة على التكيف و بالتالي ضمان أداء أفضل.

المساهمات الكبرى لهذه الأطروحة هي:

- تطوير واجهة متعددة أنماط التحكم عالية المستوى: التحكم الصوتي، التحكم عن طريق الإيماءات، و التحكم عن طريق الضغط على الصورة (التحديد و النقر) بما في ذلك التحكم اليدوي.
- صنع نوع جديد من الروبوتات المتوازية ألا وهو روبوت الكوابل الذي يفتح الطريق لمجالات بحث جديدة و تطبيقات أصلية لاسيما في مجال إعادة التأهيل.
- تطوير تطبيق أصلي باستخدام روبوت الكوابل و الواجهة متعددة أنماط التحكم لفئة خاصة في مركز لإعادة التأهيل و التعليم.
- تقييم أداء النظم تحت الإشراف و المتعددة أنماط التحكم عالية المستوى مع الأخذ في الاعتبار العوامل البشرية و ذلك باستخدام طريقة تقييم تسمى عملية التحليل الهرمي (AHP)، و التي تسمح بدمج معايير النوعية و الكمية.

الكلمات المفتاح: التفاعل بين الإنسان و الآلة، الأنظمة متعددة أنماط التحكم تحت الإشراف، التحكم عالي المستوى.