

SOMMAIRE

REMERCIEMENTS	
RESUME	ii
SOMMAIRE	iii
LISTE DES ABREVIATIONS	vi
LISTE DES FIGURES.....	vii
LISTE DES TABLEAUX.....	ix
INTRODUCTION GENERALE	1
CHAPITRE 1: GENERALITES	2
1.1. Généralités sur le signal de la parole.....	2
1.1.1. Définitions.....	2
1.1.2. Description de la parole	2
a. Niveau acoustique	2
b. Niveau phonétique	3
c. Niveau phonologique	3
d. Niveau morphologique.....	3
e. Niveau syntaxique	3
f. Niveau sémantique.....	4
g. Niveau pragmatique	4
1.1.3. Production de la parole	4
1.1.4. Les caractéristiques du signal de la parole.....	5
a. Les sons voisés (ou sonores)	5
b. Les sons non voisés	6
1.1.5. Classification des phonèmes	6
1.1.6. Les paramètres du signal de parole	6
a. La fréquence fondamentale	7
b. La hauteur de la voix.....	7
c. La durée	7

d. L'intensité	7
1.1.7. Les paramètres d'un son	8
1.1.8. La perception de la parole	8
1.2. Généralités sur la robotique	9
1.2.1. Définitions.....	9
1.2.2. Classification des robots.....	10
1.2.3. Le robot mobile	10
1.2.4. Constitution d'un robot mobile	10
a. La structure mécanique	11
b. La charge utile.....	11
c. La structure de commande	11
1.2.5. Principe de base de la navigation d'un robot mobile	11
a. Les systèmes de perception	12
b. Les méthodes de localisation	13
1.3. Conclusion	13
CHAPITRE 2: RECONNAISSANCE DE LA PAROLE	14
2.1. Extraction des paramètres	14
2.1.1. Acquisition et préaccentuation du signal.....	14
2.1.2. Segmentation et fenêtrage	15
2.1.3. La transformé de Fourier rapide	16
2.1.4. Représentation cepstrale	17
2.1.5. L'échelle Mel	18
2.1.6. Calcul des coefficients MFCC	18
2.2. La reconnaissance automatique de la parole	19
2.2.1. Introduction	19
2.2.2. Les problèmes de la reconnaissance automatique de la parole	20
2.2.3. L'outil de la reconnaissance automatique de la parole : le réseau de neurone artificiel.....	21
a. Le neurone biologique.....	21
b. Le neurone artificiel	21
c. Présentation d'un réseau de neurone artificiel	22
d. Apprentissage	23

e. Le perceptron multicouche	24
f. RNA à mémoire associative	26
2.3. Conclusion	27
CHAPITRE 3: REALISATION DU PROJET : COMMANDE D'UN ROBOT MOBILE PAR LA PAROLE	28
3.1. Présentation du projet.....	28
3.2. Présentation du côté robot à commander	29
3.2.1. Schéma du circuit	29
3.2.2. Arduino Uno.....	30
a. Présentation de la carte Arduino	30
b. La programmation Atmel ATmega328	32
3.2.3. Le driver demi-pont en H L293D	34
3.2.4. Le motoréducteur	35
3.2.5. Le module Bluetooth	35
3.3. Présentation du côté ordinateur et interface	37
3.3.1. Le langage utilisé	37
3.3.2. La commande par la parole	38
a. Calcul des coefficients MFCC	38
b. Le RNA utilisé	38
3.3.3. L'interface de commande	40
3.4. Conclusion	42
CONCLUSION GENERALE	43
ANNEXE 1 : Le VAD	44
ANNEXE 2 : DATASHEET DU MOTEUR	46
ANNEXE 3 : LE BLUETOOTH	47
ANNEXE 4 : LE LANGAGE PYTHON.....	51
REFERENCE	53

LISTE DES ABREVIATIONS

CAN : Convertisseur Analogique Numérique
DEL : Diodes ElectroLuminescent
EDI : Environnement de Développement Intégré
ESSF : Etalement de Spectre par Saut de Fréquence
IBM : International Business Machines
IEEE : Institute of Electrical and Electronics Engineers
IHM : Interface Homme Machine
L2CAP : Logical Link Control and Adaptation Protocol
LPC : Linear Prediction Coding
MCC : Moteur à Courant Continu
MFCC : Mel Frequency Cepstral Coefficients
MLI : Modulation de Largeur d'Impulsion
MMC : Modèle de Markov Caché
PIN : Personal Identification Number
PLP : Perceptually-based Linear Prediction
PMC : Perceptron MultiCouche
RAP : Reconnaissance Automatique de la Parole
RNA : Réseau de Neurone Artificiel
SDP: Service Discovery Protocol
SIG : Special Interest Group
TCD : Transformé de Cosinus Discrète
TCDI : Transformé de Cosinus Discrète Inverse
TF : Transformé de Fourier
TFD : Transformé de Fourier Discrète
TFR : Transformé de Fourier Rapide
USB : Universal Serial Bus
Wi-Fi : Wireless Fidelity

LISTE DES FIGURES

Figure 1.1: Appareil vocal	4
Figure 1.2: Son voisé	5
Figure 1.3: Son non voisé	6
Figure 1.4: Appareil auditif	8
Figure 1.5: Champ auditif humaine	9
Figure 1.6: Un robot mobile	10
Figure 1.7: Composition d'un robot mobile.....	11
Figure 1.8: Chaîne fonctionnelle d'un système de navigation.....	12
Figure 2.1 : Etape à suivre pour avoir les coefficients MFCC.....	14
Figure 2.2 : Acquisition du signal de la parole	15
Figure 2.3 : Fenêtre de Hamming	16
Figure 2.4. : Représentation du signal après la segmentation et le fenêtrage a) mot « mandroso » b) mot « mihemotra »	16
Figure 2.5 : Spectre du mot « mandroso »	17
Figure 2.6 : Banc de filtres Mel	18
Figure 2.7 : Etapes de reconnaissance automatique de la parole	20
Figure 2.8 : Neurone biologique	21
Figure 2.9 : Mise en correspondance neurone biologique/neurone artificiel	22
Figure 2.10 : Représentation d'un RNA	22
Figure 2.11: Courbe de la fonction sigmoïde.....	23
Figure 2.12 : Perceptron multicouche	24
Figure 2.13 : Structure d'une mémoire associative.....	26
Figure 3.1 : Structure générale de la commande à distance	28
Figure 3.2: Photo du projet a) L'ensemble du système b) Le robot à commander	29
Figure 3.3 : Schéma du circuit	30
Figure 3.4 : ATmega328P-PU	31
Figure 3.5 : La carte Arduino UNO Révision 3	31
Figure 3.6 : L'interface de l'environnement de développement Arduino.....	32
Figure 3.7 : Organigramme de commande du robot	33
Figure 3.8 : Le driver L293D	34
Figure 3.9 : Principe de commande du motoréducteur	35

Figure 3.10 : Photo du motoréducteur.....	35
Figure 3.11 : Le module Bluetooth HC-05	36
Figure 3.12 : Courbes représentatives d'apprentissage de classeur RNA.....	39
Figure 3.13 : Matrice de confusion	39
Figure 3.14 : Interface de commande du robot	40
Figure 3.15 : Organigramme de commande vocale	41
Figure A. 1 : Résultat de VAD d'une suite de parole	44
Figure A. 2 : Etapes de l'algorithme VAD	44
Figure A. 3 : Datasheet du MCC FC-130RA/SA.....	46
Figure A. 4 : L'interface de l'environnement de développement PyCharm	51



LISTE DES TABLEAUX

Tableau I : Commande et impact sur le moteur	34
Tableau A. 1 : Différence entre les normes.....	48
Tableau A. 2 : Les classes Bluetooth	49

INTRODUCTION GENERALE

Depuis 1970, le monde scientifique et industriel porte toujours des efforts sur l'étude et la réalisation d'engins et robots mobiles commandés et autonomes. Le but de l'homme est de se libérer du travail dans ce qu'il a d'astreignant, de fatigant, d'inintéressant, donc de simplifier la vie. Les robots mobiles seront capables de se déplacer n'importe où, surtout dans un milieu hostile où l'homme ne peut pas accéder, et d'effectuer des tâches à la place des hommes surtout dans un milieu domestique.

Aujourd'hui, les chercheurs tentent de trouver différents moyens de faciliter la commande des robots comme par exemple la commande par la reconnaissance de la parole dans la robotique. Le domaine d'application s'étend dans la domotique, l'automatique et de la mécanique automobile pour rendre un engin ou un robot obéissant et automate. Réaliser un tel projet reste toujours un grand défi surtout à Madagascar sur la connaissance requise à la réalisation du projet et sur le manque de matériel. Par conséquent, l'existence des robots mobiles à Madagascar est insuffisante pour qu'on puisse rendre la vie des peuples Malagasy facile.

Une option s'offre à nous, c'est de miniaturiser ce projet au niveau matériel pour pouvoir l'implémenter à l'avenir dans des grands projets. Ce choix nous permet d'exposer le fonctionnement minimum du projet pour ne pas rentrer dans les détails. C'est pour cette raison que notre projet s'intitule « Reconnaissance automatique de la parole : Utilisation à la commande d'un robot mobile ». La commande du robot se compose de deux sous-commandes : la commande par ordinateur et la commande par la parole.

Ce travail sera divisé en trois chapitres. Dans le premier chapitre, on va développer la généralité sur la parole et sur le robot. Ensuite le deuxième chapitre traite essentiellement le signal de parole ainsi que sa reconnaissance. Le dernier chapitre sera consacré à la réalisation pratique du projet sur la commande proprement dite du robot mobile.

CHAPITRE 1: GENERALITES

Avant d'entrer dans les détails, il est nécessaire de savoir au préalable ce qu'est la parole et le robot. Ce premier chapitre introduit donc ces deux notions pour la compréhension du travail. En effet, la section 1.1 va concerner la parole en générale, la section 1.2 parle du robot.

1.1. Généralités sur le signal de la parole

La parole est la capacité de l'être humain de communiquer la pensée par l'intermédiaire de sons articulés. Due à son importance, la parole a préoccupé depuis toujours les scientifiques. Ainsi quelques-unes des sciences qui se préoccupent de l'étude de la parole ont déjà des centaines d'années.

1.1.1. Définitions

Le signal de la parole est un signal continu, varie infiniment dans le temps donc non stationnaire, réel et d'énergie finie. Il se distingue des autres sons par ses complexités ainsi que sa variabilité dans le temps. Les sons de la parole sont produits soit par des vibrations des cordes vocales, soit par l'écoulement turbulent de l'air dans le conduit vocal, soit lors du relâchement d'une occlusion de ce conduit (source de bruit) [1].

1.1.2. Description de la parole

Il y a généralement plusieurs niveaux de description non exclusifs de la parole telle qu'acoustique, phonétique, phonologique, morphologique, syntaxique, sémantique, et pragmatique.

a. Niveau acoustique

La parole apparaît physiquement comme une variation de la pression de l'air causée et émise par le système articulatoire. La phonétique acoustique étudie ce signal en le transformant dans un premier temps en signal électrique grâce au transducteur approprié : le microphone (lui-même associé à un préamplificateur). De nos jours, le signal électrique

résultant est le plus souvent numérisé. Il peut alors être soumis à un ensemble de traitements statistiques qui visent à en mettre en évidence les traits acoustiques : sa fréquence fondamentale, son énergie, et son spectre. Chaque trait acoustique est lui-même intimement lié à une grandeur perceptuelle : pitch, intensité, et timbre.

b. Niveau phonétique

La phonétique s'intéresse au phénomène de production de la parole dans l'appareil articulatoire mais pas au niveau de signal. La phonétique est une branche de la linguistique ayant pour objet la description des sons de la parole, indépendamment de leur valeur dans le système de la langue.

c. Niveau phonologique

La phonologie est une branche de la linguistique qui s'attache à décrire les systèmes de phonèmes des langues en termes de différences et de ressemblances fonctionnelles (pertinentes pour la communication). La phonologie (parfois appelée phonétique fonctionnelle) est l'interface nécessaire entre la phonétique et les descriptions linguistiques de niveau plus élevé.

d. Niveau morphologique

La morphologie est la branche de la linguistique qui étudie comment les formes lexicales sont obtenues à partir d'un ensemble réduit d'unités porteuses de sens, appelées morphèmes.

e. Niveau syntaxique

Le niveau syntaxique vise à ce que le groupement des différents mots soit syntaxiquement correct. La syntaxe est une étude descriptive des relations qui existent entre les mots, les syntagmes, et de leurs fonctions dans la phrase. Elle restreint l'ensemble de phrases acceptables pour une langue donnée. Par exemple une phrase doit comporter un groupe nominal et un groupe verbal au minimum.

f. Niveau sémantique

La syntaxe a des limites car même s'il y a une phrase syntaxiquement exacte, celle-ci peut être inadmissible. La sémantique lexicale est l'étude des significations des mots, du langage du point de vue du sens (polysémie, synonymie, changements de sens, relations unissant les unités signifiantes, etc.). L'étude de grammaires et d'analyseurs sémantiques est un sujet de recherche actuel en linguistique informatique. Seules des solutions partielles ont pu être obtenues jusqu'à présent (analyse dans un domaine sémantique restreint).

g. Niveau pragmatique

Au contraire du sens sémantique (qualifié souvent d'indépendant du contexte), le sens pragmatique est défini comme dépendant du contexte. Tout ce qui se réfère au contexte, souvent implicite, dans lequel une phrase s'inscrit à la relation entre le locuteur et de son auditoire, a quelque chose à voir avec la pragmatique. Son étendue couvre l'étude de sujets tels que les présuppositions, les implications de dialogue, les actes de parole indirects, etc. Elle est malheureusement bien moins développée encore que la sémantique.

1.1.3. Production de la parole

L'appareil vocal, ou système phonatoire de la Fig. 1.1, comprend quatre éléments fondamentaux fonctionnant en étroite synergie pour produire des signaux acoustiques [2].

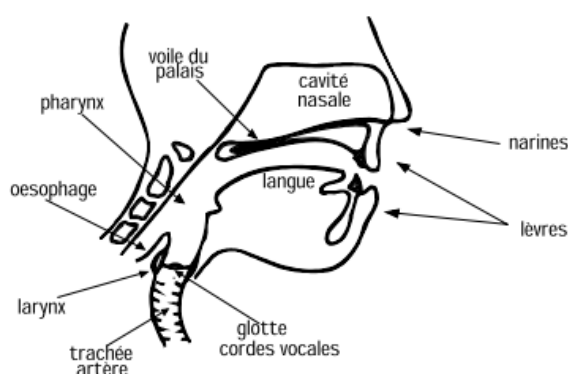


Figure 1.1: Appareil vocal [2]

Ce sont, dans l'ordre où ils s'élaborent.

- La soufflerie : constitue un réservoir d'air, les poumons, actionnés par les muscles

du thorax et de l'abdomen, et d'un tube, la trachée artère, qui conduit l'air aux cordes vocales;

- Le vibreur : le vibreur est le larynx, qui engendre les ondes aériennes;
- Le corps sonore, il est constitué d'un ensemble complexe de résonateurs, dont le pharynx et la bouche sont les principaux;
- Le système articulateur, enfin, se compose d'éléments fixes et mobiles permettant de modifier largement la forme de l'onde laryngée. Tous ces éléments sont placés sous la dépendance étroite du système nerveux central, qui en assure le synchronisme et la coordination.

1.1.4. Les caractéristiques du signal de la parole

L'une de plus importantes caractéristiques du signal vocal est la nature de l'excitation. Il existe deux types élémentaires d'excitation qui produisent les sons voisés et les sons non voisés.

a. Les sons voisés (ou sonores)

Les sons voisés sont produits à partir d'une excitation qui actionne sur le conduit vocal et qui consiste en une suite d'impulsions périodiques d'air fournies par le larynx. Les cordes vocales au début sont fermées. Sous la pression continue de l'air qui vient des poumons elles s'ouvrent graduellement délivrant cette énergie potentielle. Pendant cette ouverture la vitesse de l'air et l'énergie cinétique augmentent jusqu'à ce que la tension élastique des cordes vocales égale la force de séparation du courant d'air, autrement dit les cordes vocales vibrent. Les signaux de sons voisés sont des signaux périodiques dont un échantillon est présenté à la Fig. 1.2 [2].



Figure 1.2: Son voisé [2]

b. Les sons non voisés

Les sons non voisés sont générés lorsque l'air passe librement dans le conduit vocal. Ils sont générés sans l'apport du larynx et ne présentent pas de structure périodique comme représenté à la Fig. 1.3 [2].



Figure 1.3: Son non voisé [2]

1.1.5. Classification des phonèmes

Il y a plusieurs façons de classifier les phonèmes. Un phonème est stationnaire ou continuant si la configuration du conduit vocal ne change pas pendant la production du son. Un phonème est non continuant si pendant sa production il y a des changements dans la configuration du conduit vocal :

- **Les voyelles** sont des sons voisés, continus, normalement avec la plus grande amplitude parmi tous les phonèmes et elles peuvent varier beaucoup en durée, entre 40 et 400 ms [1]. Elles sont différentes de tous les autres sons par le degré d'ouverture du conduit vocal.
- **Les diphtongues** sont essentiellement des sons non continus, non-stationnaire.
- **Les consonnes** sont des sons pour lesquels le conduit vocal est plus étroit pendant la production, par rapport aux voyelles. Les consonnes impliquent les deux formes d'excitation pour le conduit vocal et elles peuvent être continuantes ou non. Les consonnes fricatives, les consonnes occlusives, les consonnes nasales, les consonnes liquides sont différents types de consonne.
- **Les semi-consonnes**, ce sont des sons non continus et voisés qui possèdent des caractéristiques spectrales semblables aux voyelles.

1.1.6. Les paramètres du signal de parole

Le signal de la parole est caractérisé par quatre paramètres qui sont : son pitch (ou sa fréquence fondamentale), sa hauteur, sa durée ainsi que son intensité.

a. La fréquence fondamentale

Elle s'appelle souvent « le pitch », noté F_0 . Celle-ci constitue une caractéristique très importante de la voix. Elle correspond à la fréquence de vibration des cordes vocales lors de la production de voyelles ou de consonnes voisées. Elle génère des variations prosodiques, c'est à dire de mélodie et d'intonation, qui contribuent à l'identification du sexe, de l'âge, de l'identité du locuteur, ainsi que la signification du message prononcé. Donc à l'intérieur des zones voisées, la fréquence fondamentale évolue lentement dans le temps. Elle s'étend approximativement de 70 à 250 Hz chez les hommes, de 150 à 400 Hz chez les femmes, et de 200 à 600 Hz chez les enfants [2].

b. La hauteur de la voix

La hauteur c'est la qualité sonore et elle dépend du nombre de vibration des cordes vocaux. En effet, il faut savoir qu'en matière d'acoustique, pour qu'un corps soit capable d'émettre un son, il doit avoir la propriété de vibrations. Ces vibrations sont alors transmises à l'oreille par l'intermédiaire de l'air et leur fréquence peut varier entre 16 et 20000 vibrations par seconde (16 Hz à 20 kHz). Ainsi plus le son est aigu, plus les vibrations sont nombreuses et inversement.

c. La durée

La durée ou la tenue des sons dépend de la pression de l'air expiré.

d. L'intensité

L'intensité du son se mesure en général en [dB] et celui-ci distingue un son fort d'un son faible. Par exemple une conversation normale a une intensité entre 20 et 50 dB, tandis qu'une discussion animée se mesure de 50 à 65 dB.

1.1.7. Les paramètres d'un son

Un son est représenté mathématiquement par plusieurs paramètres :

- **La fréquence d'échantillonnage**

Elle désigne nombre des valeurs définissant l'amplitude pour une seconde d'enregistrement. Ainsi 11 025 Hz signifie 11 025 échantillons pour une seconde de son mémorisé. Plus la fréquence du son est élevée, plus le son est aigu. Plus la fréquence d'échantillonnage est élevée, meilleure est la qualité des échantillons.

- **L'amplitude**

C'est l'ordonnée du point le plus élevé (et le plus bas) de la courbe représentative d'un son. Plus l'amplitude est élevée, plus le son est fort, bruyant. C'est l'amplitude qui est numérisée par le Convertisseur Analogique Numérique (CAN). Par exemple, en 8 bits, l'amplitude possède une résolution de 256 valeurs. Plus la résolution est élevée, plus l'échantillon est proche du son original.

- **Le nombre de bits** d'un échantillon

- **Le nombre de voies** (un seul correspond à la monophonie, deux à la stéréophonie)

1.1.8. La perception de la parole

Dans le cadre du traitement de la parole, une bonne connaissance des mécanismes de l'audition et des propriétés perceptuelles de l'oreille est aussi importante qu'une maîtrise des mécanismes de production. Les ondes sonores sont recueillies par l'appareil auditif, ce qui provoque les sensations auditives. Ces ondes de pression sont analysées dans l'oreille interne qui envoie au cerveau l'influx nerveux qui en résulte. La figure 1.4 montre l'appareil auditif qui comprend l'oreille externe, l'oreille moyenne, et l'oreille interne [2].

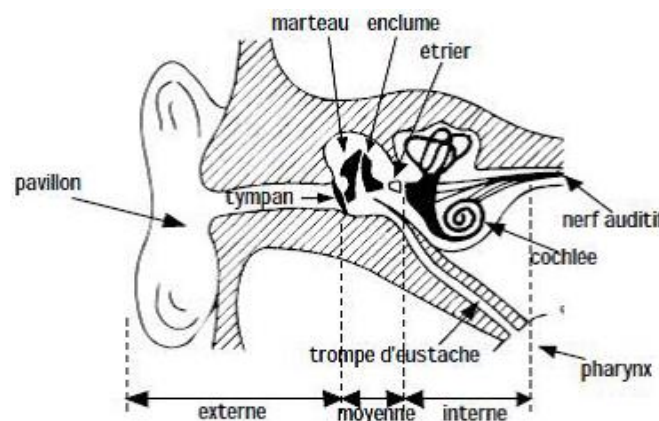


Figure 1.4: Appareil auditif [2]

L'oreille ne répond pas également à toutes les fréquences. La figure 1.5 présente le champ auditif humain, délimité par la courbe de seuil de l'audition et celle du seuil de la douleur [2]. Sa limite supérieure en fréquence (≈ 16000 Hz, variable selon les individus) fixe la fréquence d'échantillonnage maximale utile pour un signal auditif (≈ 32000 Hz).

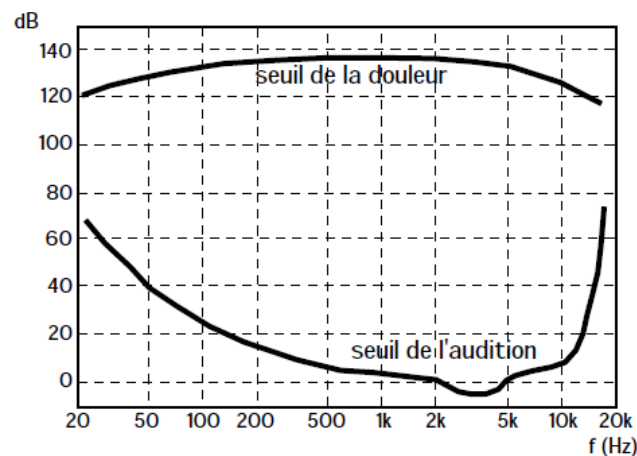


Figure 1.5: Champ auditif humaine [2]

1.2. Généralités sur la robotique

1.2.1. Définitions

- Le terme « robotique » se définit comme ensemble des techniques permettant la conception et la réalisation de machines automatiques ou de robots.

Ce terme a été utilisé pour la première fois par le romancier Isaac Asimov dans son récit de science-fiction «Menteur !», publié en 1941 dans Astounding Science-Fiction. Dans certains autres ouvrages d'Asimov, il affirme que la première utilisation du mot "robotique" était dans sa courte histoire Runaround (Astounding Science-Fiction, Mars 1942. Asimov n'était initialement pas conscient d'avoir créé le mot. Il a supposé que le terme existait déjà, par analogie avec «mécanique» (comme «positronique» avec «électronique») [3].

- Un robot est un appareil automatique capable de manipuler des objets, ou d'exécuter des opérations selon un programme fixe ou modifiable. Souvent un robot est défini comme un manipulateur automatique à cycles programmables.

Un système mécanique est un robot s'il possède une certaine flexibilité, caractérisée par la versatilité, l'auto-adaptativité : un robot doit pouvoir s'adapter à un environnement changeant au cours de l'exécution de ses tâches [4].

1.2.2. Classification des robots

Actuellement, il existe deux grandes catégories de robots :

- **Les robots manipulateurs** : ce sont des robots fixes et sont utilisés généralement dans les industries de pointes. Ils sont programmés pour exécuter fidèlement des actions répétitives avec un haut degré de précision.
- **Les robots mobiles** : ce sont des robots à base mobile (par opposition notamment aux robots manipulateurs) commandés par un système de commande. Ils sont capables de se déplacer muni d'un ou plusieurs capteurs. Ils offrent alors des informations environnementales. De plus ils sont alimentés par une source d'énergie électrique autonome.

Puisque le travail est focalisé sur l'étude des robots mobiles, il est alors nécessaire d'entrer en détail dans le fonctionnement du système.

1.2.3. Le robot mobile

Un robot mobile un véhicule capable de naviguer de manière autonome dans un environnement complexe, parfois évolutif, partiellement connu ou inconnu, et d'exécuter des tâches programmées sous contrôle humaine réduite. La figure 1.6 montre un robot mobile [5].

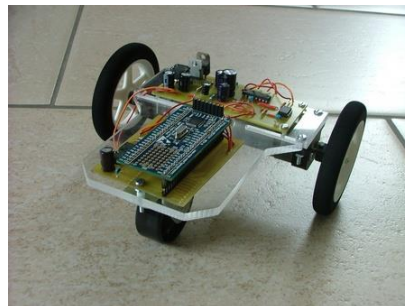


Figure 1.6: Un robot mobile [5]

1.2.4. Constitution d'un robot mobile

Un système mécatronique est décomposé de trois sous-systèmes tels que la structure mécanique, la charge utile et le système de commande. Le système de commande contrôle les deux autres sous-systèmes. La figure 1.7 illustre la décomposition d'un robot mobile.

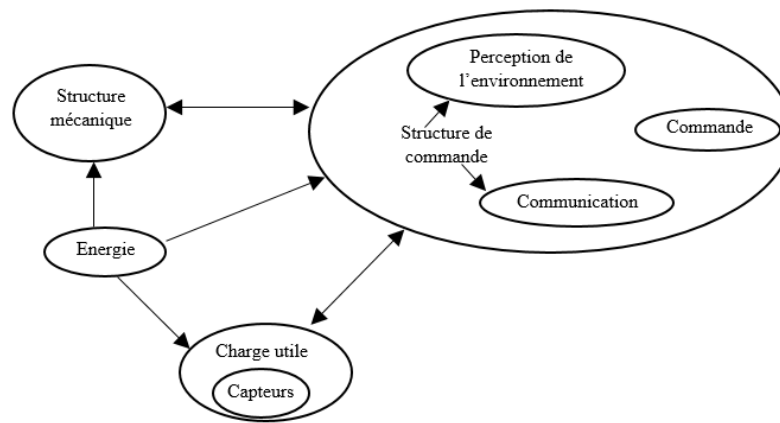


Figure 1.7: Composition d'un robot mobile

a. La structure mécanique

La structure mécanique rassemble les éléments mécaniques et électriques permettant d'assurer la fonction de mobilité. Le mouvement est obtenu en associant la propulsion et l'appui sur le milieu. La propulsion très liée à l'application du robot, est assurée par des techniques variées : électrique, thermique, pneumatique ou hydraulique, en fonction de sa taille, sa masse et de son lieu d'évolution. En robotique, le nombre des roues utilisées est souvent 3, 4 ou 6 pour avoir la bonne stabilité mécanique.

b. La charge utile

La charge utile concerne directement l'application du robot. Le déplacement est dicté par une action à réaliser sur l'environnement.

c. La structure de commande

La structure de commande repose sur trois modules fonctionnant de manière indépendante entre eux. Il s'agit du module de perception de l'environnement, de communication homme machine et du module de commande.

1.2.5. Principe de base de la navigation d'un robot mobile

La décomposition du problème de la mobilité pour les robots mobiles autonomes amène à définir une architecture classique en robotique, organisée suivant un

fonctionnement séquentiel « perception-décision-action ». La prise de décision nécessite donc un conditionnement le plus robuste possible des données sensorielles issues du système de perception. L'étape de décision peut revêtir plusieurs formes suivant la tâche considérée. C'est à partir de la connaissance de son environnement et des buts qu'il doit atteindre qu'un robot mobile pourra élaborer des lois de commande et pourra prendre des décisions d'action adéquates [6].

a. Les systèmes de perception

La fonction perception consiste globalement à saisir un certain nombre d'informations sensorielles dans le but d'acquérir une connaissance et une compréhension du milieu d'évolution. Elle est indispensable aux étapes suivantes qui sont généralement pour un robot mobile les étapes de localisation et de mise à jour de la carte de l'environnement.

La figure 1.8 représente la chaîne fonctionnelle de la navigation, qui constitue en les différentes étapes, pour le processus de localisation d'un robot mobile, de perception à l'action.

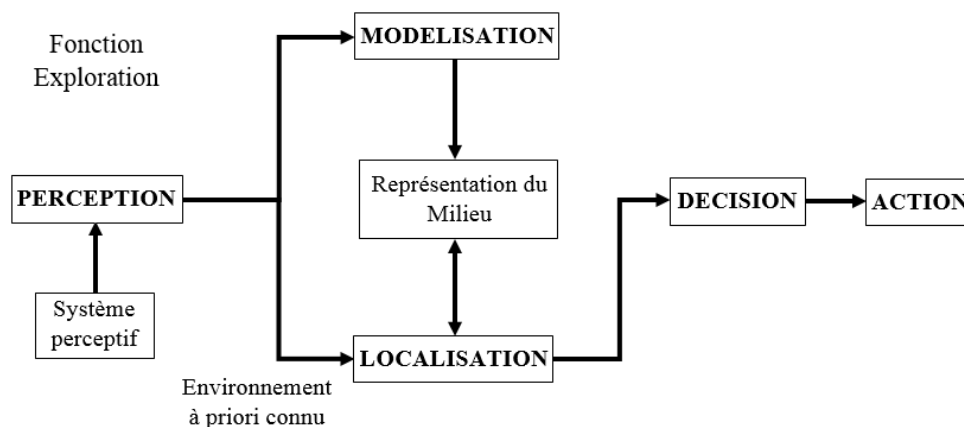


Figure 1.8: Chaîne fonctionnelle d'un système de navigation

Le système de perception d'un robot mobile est basé sur la mise en place des capteurs classifiés en deux catégories :

- **Les capteurs proprioceptifs** qui fournissent des informations propres au comportement interne du robot, c'est-à-dire sur son état à un instant donné,
- **Les capteurs extéroceptifs** qui permettent de fournir certaines connaissances de l'environnement de travail.

Il existe une troisième classe de capteurs non inclus dans les boucles d'asservissement, dont l'intérêt est de veiller sur l'état du robot lui-même que sur l'environnement. Ce sont des

capteurs de sécurité qui se déclenchent en cas de mal fonctionnement ou de panne des boucles internes ou externes. Ce type de capteur génère un ordre d'arrêt du robot lorsque le dépassement de certaines valeurs a lieu [7].

b. Les méthodes de localisation

La problématique de localisation est précédemment explicité comme étant un préalable incontournable de celle de la génération de carte incrémentale de l'environnement. La suite logique de la synthèse faite sur les capteurs portera donc sur les différentes méthodes de localisation. Par analogie avec la classification des capteurs, trois types de méthodes de localisation est distinguée comme suit :

- Les méthodes de localisation relatives, basées sur l'utilisation des capteurs proprioceptifs.
- Les méthodes de localisation absolues, basées sur l'utilisation des capteurs extéroceptifs.
- Les méthodes dites hybrides qui sont basées sur l'utilisation conjointe de deux types de capteurs.

1.3. Conclusion

La parole est un signal sonore, continu et non stationnaire, elle se produit par la vibration de cordes vocales humaines. Elle se caractérise par les sons voisés et ceux non voisés. Le signal de parole peut se mesurer à partir des paramètres tels que la fréquence fondamentale qui varie de 70 Hz à 600 Hz, la hauteur située dans l'intervalle de 16 Hz à 20 kHz, la durée et l'intensité mesurée en dB. L'une de ses utilisations intéressantes est de commander un robot mobile. Les études et les techniques de mise en œuvre de ce dernier ne cessent d'évoluer pour le rendre utile dans la vie quotidienne. De ce fait, la conception est complexe en combinant des notions mécanique, électronique, automatique et informatique.

CHAPITRE 2: RECONNAISSANCE DE LA PAROLE

La reconnaissance de la parole fait partie des techniques de traitement de la parole. Elle est utilisée dans plusieurs domaines. Dans ce projet, elle est appliquée pour commander un robot. Afin d'atteindre cet objectif, il faut d'abord extraire les paramètres caractéristiques en utilisant la méthode cepstrale, puis les grouper selon des classes déterminées par la méthode de Réseau de Neurone Artificiel (RNA).

2.1. Extraction des paramètres

Pour pouvoir utiliser un signal de la parole pour la reconnaissance, il faut d'abord extraire des paramètres qui caractérisent l'information caché derrière ce signal. Les deux méthodes les plus connues pour réaliser cette extraction sont l'analyse spectrale comme par exemple l'application de transformations non-linéaires de l'échelle des fréquences de Mel et l'analyse paramétrique comme par exemple l'utilisation d'une estimation spectrale par prédiction linéaire ou Linear Prediction Coding (LPC). Ces analyses vont produire respectivement, des coefficients cepstraux des fréquences Mel ou Mel Frequency Cepstral Coefficients (MFCC) et des coefficients perceptuels basés sur la prédiction linéaire ou Perceptually-based Linear Prediction (PLP). La figure 2.1 montre les étapes à suivre pour avoir la représentation cepstrale.

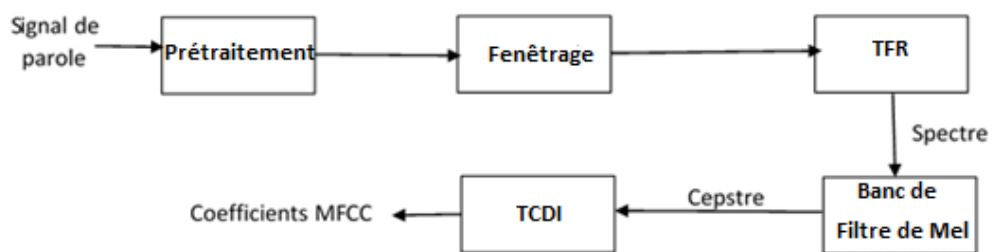


Figure 2.1 : Etape à suivre pour avoir les coefficients MFCC

2.1.1. Acquisition et préaccentuation du signal

L'acquisition du signal de parole constitue la première étape à franchir. Il s'agit de numériser un signal analogique (la parole) pour qu'il soit prêt à des traitements numériques ultérieurs. La figure 2.2 représente les opérations successives permettant de transformer un signal continue $x(t)$ (où t désigne le temps) en un signal numérique $x(n)$ où n correspond à des instants discrets.



Figure 2.2 : Acquisition du signal de la parole

Une fois capté par un microphone, le signal est tout d'abord amplifié, puis filtré, ensuite échantillonné et enfin quantifié. Le filtre est l'outil essentiel pour les traitements des signaux que ce soit analogique ou numérique. Au cours du filtrage, il y a le débruitage de parole qui est un traitement appliqué au signal de parole afin de réduire les effets néfastes d'un environnement bruyant. Il peut s'agir par exemple du bruit ambiant d'un bureau, d'une cafétéria, de la rue, ou de signaux de parole provenant d'autres locuteurs.

2.1.2. Segmentation et fenêtrage

Pour analyser un signal de parole, il faut le segmenter en divisant en plusieurs blocs avec recouvrement. Chaque bloc du signal est contenu dans un terme appelé fenêtre d'analyse. Le choix de la taille de cette fenêtre en nombre de points s'avère très important vis-à-vis de la qualité d'analyse obtenue. Ainsi, une fenêtre de petite taille permettra d'obtenir une bonne analyse dans le domaine temporel, du fait de son étroitesse, mais ne permettra pas d'obtenir une bonne information fréquentielle. À l'inverse, une fenêtre de grande taille permettra d'obtenir une bonne information fréquentielle mais ne permettra pas d'obtenir une bonne information temporelle. La théorie de la Transformée de Fourier (TF) nécessitant la stationnarité, il faut alors réaliser un compromis entre la précision souhaitée et le respect de l'hypothèse de stationnarité [8]. Le fenêtrage signifie donc multiplier l'amplitude d'un signal de parole par l'amplitude de la fenêtre, pour tout instant t . Selon les littératures, Plusieurs types de fenêtres sont utilisés mais les plus courantes sont les fenêtres de Hamming, de Bartlett, rectangulaires, de Blackman et de Kaiser, la fenêtre de Hamming de la Fig. 2.3 est généralement utilisée pour le traitement de signal de parole. Elle est exprimée par l'Eq. 2.1 :

$$w(n) = 0.54 - 0.46 * \cos \frac{2\pi n}{(N - 1)}, \quad 0 < n < N - 1 \quad (2.1)$$

n représente ici l'échantillon considéré dans le signal et N représente la taille de la fenêtre.

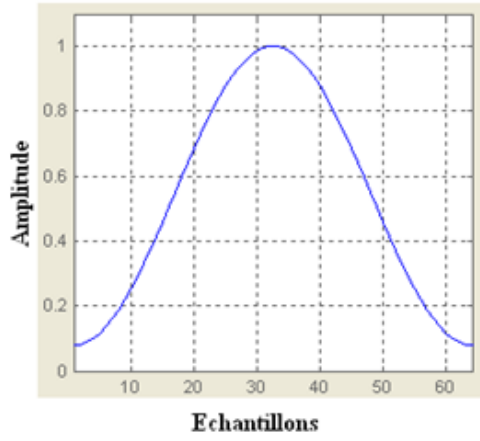


Figure 2.3 : Fenêtre de Hamming

Le signal tronqué y_i à partir d'un bloc du signal $x_i(t)$ est alors :

$$y_i(t) = w(t) * x_i(t), \quad 0 < i < M - 1 \quad (2.2)$$

i est l'indice du signal et M le nombre de bloc du signal.

Les signaux « mandroso » et « mihemotra » de la Fig. 2.3 sont obtenus après segmentation et fenêtrage :

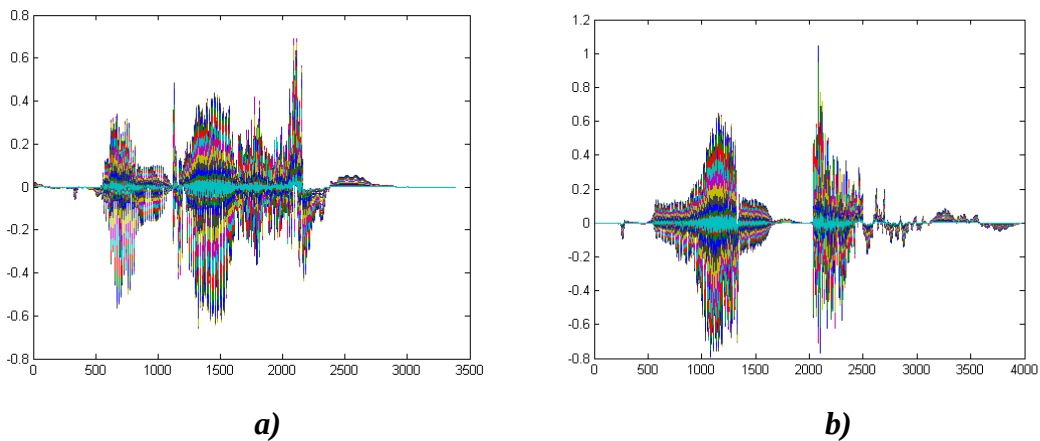


Figure 2.4. : Représentation du signal après la segmentation et le fenêtrage a) mot « mandroso » b) mot « mihemotra »

2.1.3. La transformé de Fourier rapide

La TF d'un signal à temps continu et à temps discret n'est pas sous une forme appropriée pour être obtenu par un calculateur numérique. Mais vue son importance en traitement du signal, il est nécessaire de la mettre sous une forme pratiquement utilisable, cette forme est la Transformé de Fourier Discrète (TFD) qui s'exprime par l'Eq. 2.3 :

$$X_n = \sum_{k=0}^{N-1} x(k) e^{-j2\pi nk/N} \quad (2.3)$$

Le calcul d'une TFD, nécessite une grande quantité d'opérations et devient très long si le nombre d'échantillons est élevé cela implique un retard de réponse, Pour pallier à ce problème il existe une façon de calculer la même chose, c'est la Transformée de Fourier Rapide (TFR). Ce n'est pas une nouvelle transformation, elle n'est rien d'autre qu'un moyen rapide d'obtenir les mêmes résultats que ceux fournis par la TFD. L'algorithme de la TFR utilise le fait que l'opération TF à temps discret peut être décomposée en la TFD de séquences de plus en plus courtes.

Ainsi, la TFD devient [8] :

$$Y(k) = \sum_{k=0}^{M/2-1} y(2k)e^{\frac{-j2\pi(2k)m}{M}} + \sum_{k=0}^{M/2-1} y(2k+1)e^{\frac{-j2\pi(2k+1)m}{M}} \quad (2.4)$$

Une fois la TFR appliquée au signal tronqué, cette transformée permet de passer du domaine temporel au domaine fréquentiel, cette représentation fréquentielle s'appelle spectrogramme. La TFR s'applique au signal tronqué d'Eq. 2.2. Un exemple de spectre est donné à la Fig. 2.4.

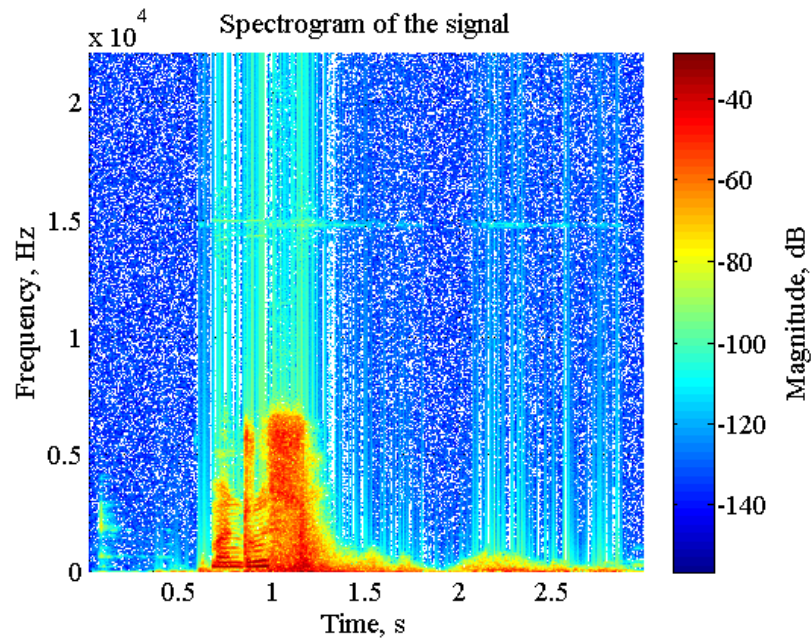


Figure 2.5 : Spectre du mot « mandroso »

2.1.4. Représentation cepstrale

Il y a présence d'intermodulation dans le spectre, ce qui est un problème pour l'analyse. De ce fait, il est nécessaire de supprimer cette intermodulation. Cette dernière est due à la convolution qui est réalisée par le conduit vocal sur la fréquence fondamentale produite par les cordes vocales. La déconvolution de celle-ci peut être effectuée grâce au cepstre. Il est à noter que le nom même de cepstre est défini à partir du mot spectre (cepstre

= (spec)⁻¹tre). De même, la représentation temps-fréquence associée n'est plus qualifiée de fréquentielle mais de quéfrentielle. Le cepstre est alors un outil très semblable au spectrogramme et est donc développé pour l'analyse cepstrale qui, au moyen du cepstre du signal, donne d'autres informations utiles en reconnaissance de la parole [6].

2.1.5. L'échelle Mel

L'énergie du spectre est calculée à travers un banc de filtres numériques couvrant la bande passante, ce qui permet de ne conserver qu'un sous ensemble de ces paramètres. Les filtres triangulaires sont les plus utilisés, Ils sont préférés pour leur simplicité et leur effet de lissage sur le spectre. Dans l'échelle de mesure Mel, la correspondance est approximativement linéaire sur les fréquences au-dessous de 1 kHz et logarithmique sur les fréquences supérieures à celle-ci. La relation entre la fréquence en échelle Hertz et sa correspondance en mels est la suivante [9] :

$$M_{mels} = 2595 \cdot \log \left(1 + \frac{f}{700} \right) \quad (2.5)$$

Où f est la fréquence.

Le travail dans ce processus consiste à convertir le spectre de puissance en spectre de Mel. Les sorties du banc de filtres peuvent être utilisées directement comme paramètres d'un système de reconnaissance. Cependant, dans les systèmes les plus récents, d'autres coefficients dérivés de ces sorties sont le plus souvent utilisés. Il s'agit des coefficients cepstraux (ou Mel Frequency Cepstral Coefficients MFCC). Ces coefficients sont linéairement répartis sur l'échelle Mel (Fig. 2.6) d'où leur nom [9].

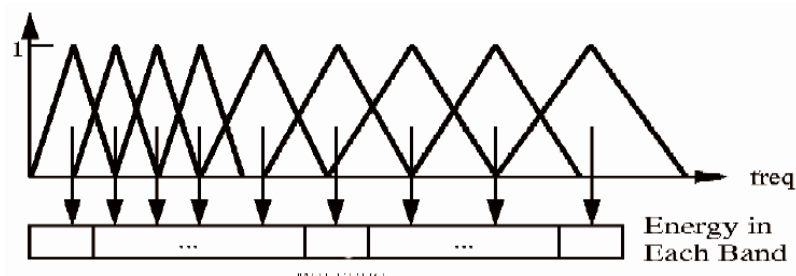


Figure 2.6 : Banc de filtres Mel [9]

2.1.6. Calcul des coefficients MFCC

Dans cette étape, le spectre de Mel est converti en représentation temporelle et le résultat s'appelle MFCC. Le cepstre du signal est donné par la Transformé de Cosinus Discrète Inverse (TCDI) du logarithme de la densité spectrale. La convolution des signaux

dans le domaine temporel se transforme en une addition dans le domaine cepstral. En effet, le signal $x(n)$ tronqué est considéré comme une convolution dans le domaine temporel du signal d'excitation $g(n)$ et de la réponse impulsionnelle du conduit vocal $h(n)$ (Eq. 2.6) :

$$x(n) = g(n) * h(n) \quad (2.6)$$

L'application du logarithme au module de la TF de cette équation donne :

$$\log|X(k)| = \log|G(k)| + \log|H(k)| \quad (2.7)$$

La TCDI de cette quantité donne le cepstre, qui est la somme des contributions de la source et du conduit. La formule d'une Transformé de Cosinus Discrète (TCD) est la suivante [10] :

$$X(u, v) = \alpha_u \alpha_v \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} f(ij) \cos \frac{(2i+1)\pi u}{2n} \cos \frac{(2j+1)\pi v}{2n} \quad (2.8)$$

avec $\alpha_x = \frac{1}{\sqrt{n}}$ si $x = 0$ et $\alpha_x = \frac{2}{\sqrt{n}}$ sinon

En introduisant l'équation 2.8 dans l'équation 2.7, les coefficients cepstraux de Mel sont obtenus :

$$c(n) = \sum_{-\frac{N}{2}}^{\frac{N}{2}} \log|X(u, v)| e^{jf2\pi \frac{n}{N}} \quad \text{où } 0 < n < N \quad (2.9)$$

Le premier coefficient $c[0]$ représente l'énergie moyenne dans la trame de la parole ; $c[1]$ reflète la balance d'énergie entre les basses et hautes fréquences ; pour $i > 1$, $c[i]$ représente des détails spectraux de plus en plus fins. Pour mieux caractériser le signal de parole, les informations sur les changements temporels sont aussi utilisés. Il s'agit des dérivées premières (Δc) et dérivées seconde ($\Delta \Delta c$). Les caractéristiques extraites du signal de parole sont donc [9] :

$$\begin{pmatrix} c_k \\ \Delta c_k \\ \Delta \Delta c_k \end{pmatrix}$$

2.2. La reconnaissance automatique de la parole

2.2.1. Introduction

D'abord, la reconnaissance de la parole ne doit pas être confondue avec la reconnaissance vocale. La reconnaissance automatique de la parole (RAP) est la capacité qu'a une machine de pouvoir analyser et interpréter des signaux de parole, cette capacité de

connaître automatiquement se base par l'apprentissage automatique ; c'est l'intelligence artificielle [11].

Dans cette étape, les caractéristiques extraites précédemment vont être transformées en information discrète afin de tirer des informations nécessaires. Pour ce faire, les signaux seront classifiés selon leurs caractéristiques. La classification est l'élaboration d'une règle de décision qui transforme les attributs caractérisant les formes en appartenance à une classe. Cette décision se fait en deux étapes : l'étape d'apprentissage et l'étape de test.

Globalement, il y'a deux approches de classification pour le signal de la parole : une approche statique et une approche dynamique. Dans la classification statique, le signal est considéré dans sa totalité et une décision unique est prise. La classification dynamique ne considère qu'une fenêtre du signal, et de ce fait prend des décisions locales, qui seront intégrées par la suite dans une décision globale. La figure 2.7 montre les étapes à suivre pour la RAP.

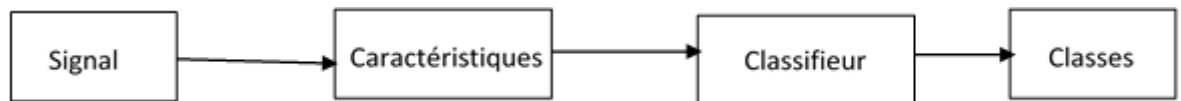


Figure 2.7 : Etapes de reconnaissance automatique de la parole

2.2.2. Les problèmes de la reconnaissance automatique de la parole

La RAP s'avère complexe à cause des variabilités suivantes:

- Variabilité intra-locuteur : Une même personne ne prononce jamais un mot deux fois de façon identique. La vitesse d'élocution peut varier, la durée du signal est alors modifiée.
- Variabilité inter-locuteur : la fréquence fondamentale de voix dépend du genre du locuteur (enfant, femme, homme), des accents régionaux et aussi varie en fonction de sa façon de parler (chanter, murmurer, enrhumé, bégayer).
- Variabilité due à l'environnement : L'acoustique du milieu (milieu protégé ou environnement bruyant), la qualité du microphone ou de la ligne téléphonique, les bruits de la bouche, les hésitations, les mots hors-vocabulaire sont autant d'interférences supplémentaires sur le signal de parole que le système de reconnaissance doit compenser.

2.2.3. L'outil de la reconnaissance automatique de la parole : le réseau de neurone artificiel

Le Modèle de Markov Caché (MMC) est la technologie la plus répandue en RAP, mais le RNA fait aussi son essor ces derniers temps d'où le choix de l'utiliser [12].

a. Le neurone biologique

Le neurone est une cellule composée d'un corps cellulaire et d'un noyau. Le corps cellulaire se ramifie pour former les dendrites. Celles-ci sont parfois si nombreuses et sont nommées alors de chevelure dendritique ou d'arborisation dendritique. C'est par les dendrites que l'information est acheminée de l'extérieur vers le soma, corps du neurone. L'information traitée par le neurone chemine ensuite le long de l'axone (unique) pour être transmise aux autres neurones. La transmission entre deux neurones n'est pas directe. En fait, il existe un espace intercellulaire de quelques dizaines d'Angströms (10^{-9} m) entre l'axone du neurone afférent et les dendrites du neurone efférent. La jonction entre deux neurones est appelée la synapse (Fig. 2.8) [13].

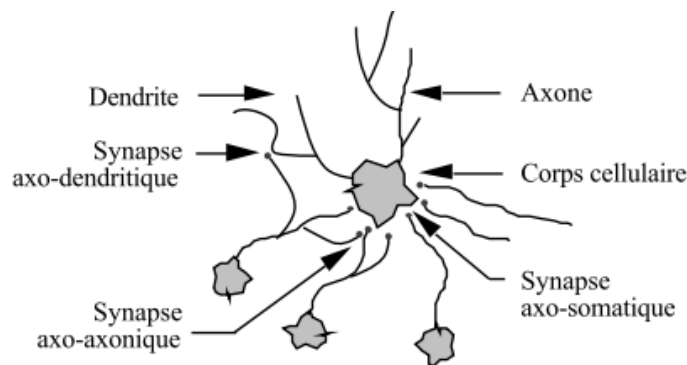


Figure 2.8 : Neurone biologique [13]

b. Le neurone artificiel

Un réseau de neurones artificiel est un modèle mathématique prédictif qui peut être utilisé pour la classification ou la régression. Ce modèle est très schématiquement inspiré du fonctionnement des neurones biologiques, la correspondance entre ces deux neurones est présentée à la Fig. 2.9 [13].

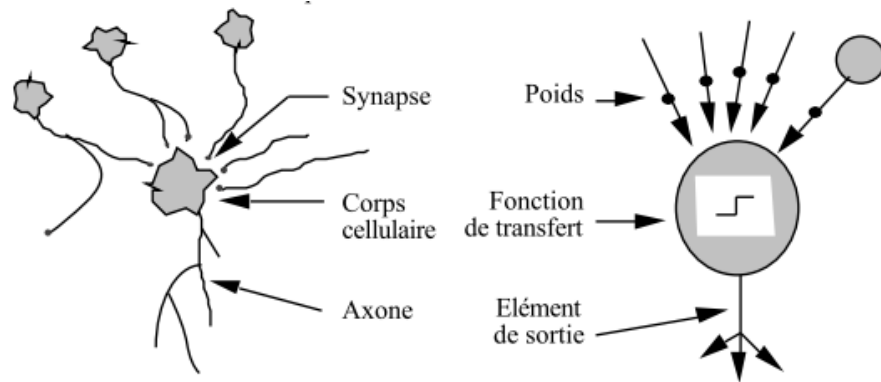


Figure 2.9 : Mise en correspondance neurone biologique/neurone artificiel [13]

c. Présentation d'un réseau de neurone artificiel

Il existe différents types de réseaux RNA utilisés dans le domaine scientifique, ils ont les mêmes tâches soit classification soit régression. Le choix du type de celui-ci dépend alors sur les conditions de son utilisation. C'est la classification lorsque la valeur de la sortie est une valeur discrète. Lorsque la valeur de sortie est continue, il s'agit d'une tâche de régression.

La figure 2.10 montre la représentation générale d'un RNA.

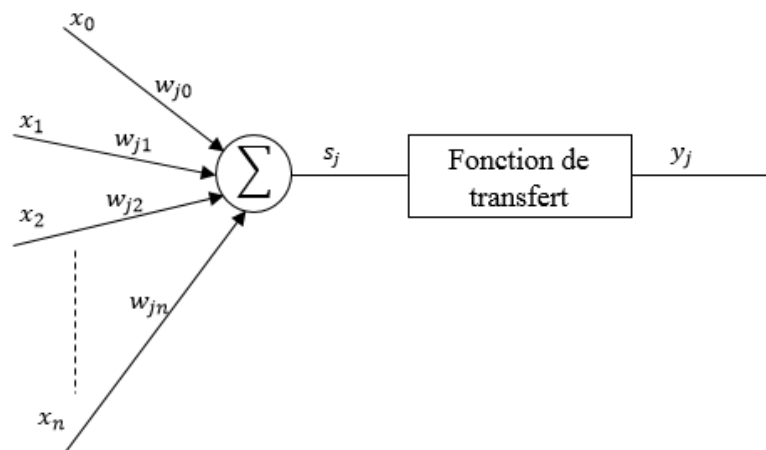


Figure 2.10 : Représentation d'un RNA

Il est constitué de diverses lignes d'entrées (x_0 à x_n) ayant chacune un poids. Le concept de poids est très important car il est directement relié à celui d'apprentissage. En effet, lorsqu'un réseau "apprend", il ajuste des poids associés aux lignes d'entrée de ses neurones de façon à ce que le réseau se comporte de manière adéquate selon diverses excitations. Le neurone artificiel est aussi muni d'un sommateur qui fait la somme de ses entrées. Le sommateur peut avoir à sa sortie une fonction de transfert ou propager

l'information telle quelle. Les poids (w_{j0} à w_{jn}) ainsi que le traitement possible de la sortie par une fonction de transfert y sont représentés.

L'équation de la sortie s_j du sommateur est :

$$s_j = \sum_{i=0}^n x_i w_{ji} \quad (2.10)$$

Où x_i est la sortie des i neurones émetteurs et w_{ji} le poids de la connexion du neurone i vers le neurone j .

Une fois s_j calculée, l'activation y_j du neurone j est calculée en utilisant la fonction de transfert f :

$$y_j = f(v_j) \quad (2.11)$$

La fonction de transfert joue le rôle d'activer les neurones. Pour cela, la fonction sigmoïde d'Eq. 2.12 est souvent utilisée dont la courbe est présentée à la Fig. 2.12 :

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.12)$$

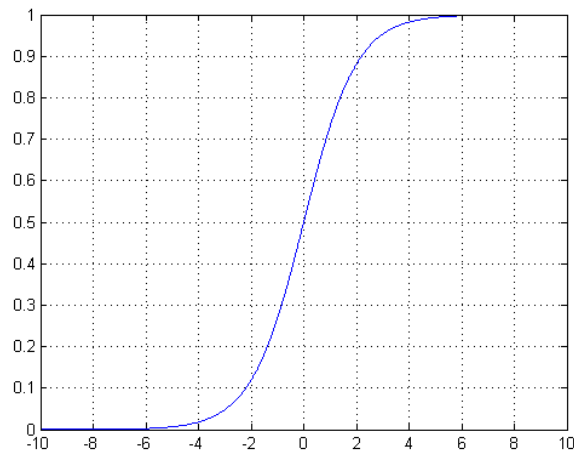


Figure 2.11: Courbe de la fonction sigmoïde

d. Apprentissage

L'apprentissage a comme objectif d'adapter les poids des connexions entre les neurones de sorte que le réseau donne en sortie la classe d'appartenance des formes qui lui sont proposées en entrée. Le réseau entraîné sera utilisé ultérieurement pour produire les résultats attendus. Un RNA effectue ses tâches soit en apprentissage supervisé, soit en apprentissage non supervisé, soit en apprentissage semi-supervisé, le type est choisi en fonction du résultat voulu.

Dans ce style d'apprentissage, la sortie désirée est fournie au réseau pour chaque forme en entrée, ce qui permet de corriger explicitement l'erreur (entre la sortie souhaitée et la sortie obtenue) commise lors de l'activation.

ii. Apprentissage non supervisé

Dans ce cas, les résultats attendus ou désirés ne sont pas présentés au réseau de neurones. Le système apprend par lui-même en découvrant et en s'adaptant selon les caractéristiques des modèles d'entrée.

iii. Apprentissage semi-supervisé

Dans ce type d'apprentissage, les sorties désirées ne sont pas données au réseau mais une évaluation de celles-ci en terme de bonne ou mauvaise approximation.

Des règles ou algorithmes existent pour l'apprentissage tel que la règle de Hebb, la règle de Widrow-Hoff, l'algorithme de rétropropagation.

e. Le perceptron multicouche

Le réseau de Perceptron multicouche (PMC) de la Fig. 2.12 est le plus connu des RNA, il peut résoudre les problèmes non linéairement séparables.

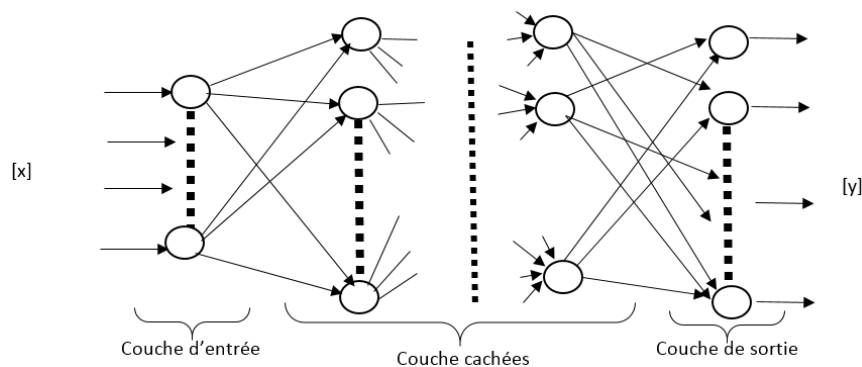


Figure 2.12 : Perceptron multicouche

Il se structure en couches. La première couche du réseau constitue la couche d'entrée. La dernière couche est la couche de sortie qui fournit le résultat du traitement effectué par le réseau. Les couches intermédiaires sont appelées : couches cachées. Les neurones d'une

même couche ne sont pas connectés entre eux, et que chaque couche dans le réseau reçoit les informations de la couche précédente et les transmet à la couche suivante. Le calcul de l'activation dans le réseau se fait en propageant l'activation initiale de la couche d'entrée jusqu'à la couche de sortie. L'erreur est calculée dans le sens inverse. Puisque le réseau ne possède pas de boucles, ces deux calculs peuvent être réalisés chacun en une passe unique dans le réseau, et ne requièrent donc qu'un temps proportionnel au nombre de liens synaptiques.

Toutes les valeurs considérées sont des matrices :

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_k \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ y_n \end{bmatrix}$$

En général, la fonction de transfert utilisée pour les neurones du PMC est la fonction sigmoïde, exponentielle ou tangentielle. Le PMC utilise une règle d'apprentissage qui est la rétropropagation de l'erreur. L'algorithme de rétropropagation est une généralisation de la règle de Widrow-Hoff pour un réseau multicouche, il est largement utilisé lors de l'apprentissage supervisé [14]. Dans cet algorithme la propagation du signal se fait des cellules d'entrée vers la couche de sortie, mais aussi peut-on suivre le chemin inverse pour rétro propager l'erreur commise en sortie vers les couches internes. Elle est une technique de descente de gradient qui minimise la distance entre la sortie du réseau et la sortie désirée.

L'algorithme de la rétropropagation de l'erreur est le suivant [15] :

- Initialiser les poids à des petites valeurs et les seuils du réseau.
- Insérer à l'entrée du réseau un exemple de la base de données en forme de vecteur de caractéristiques, puis calculer sa valeur d'activation et sa fonction d'activation en utilisant les formules (2.11) et (2.12).
- Évaluer le signal d'erreur des sorties du réseau en utilisant l'Eq. 2.13.

$$\delta_j = (d_j - y_j)y_j(1 - y_j) \quad (2.13)$$

Où d la valeur désirée du neurone j

- Corriger les poids en utilisant l'Eq. 2.14.

$$\Delta w_{ij} = \eta \delta_j y_i \quad (2.14)$$

Avec η la constante d'apprentissage ($0 < \eta < 1$)

- Évaluer le signal d'erreur pour chaque couche cachée en utilisant l'Eq. 2.15.

$$\delta_j = y_j(1 - y_j) \sum_k w_{jk} \delta_k \quad (2.15)$$

- Ajuster les poids de la couche cachée en utilisant l'Eq. 2.14.

- Retourner à la seconde étape. et répéter pour l'entrée suivante, jusqu'à ce que l'erreur en sortie soit inférieure à la limite fixée ou que le nombre maximum d'itérations soit atteint.

f. RNA à mémoire associative

Une mémoire associative stocke dans un RNA un certain nombre de formes et le RNA reconstitue l'intégralité de la forme stockée si une partie de la forme stockée lui est fournie. L'information mémorisée ne peut être obtenue qu'à une adresse précise, le seul moyen d'accès est de fournir une information [13][16]. La structure neuronale d'une mémoire associative est similaire à celle d'un ensemble de perceptrons tous alimentés par les mêmes entrées.

La figure 2.11 montre cette architecture où chaque entrée est connectée par des poids modifiables à toutes les sorties. La dimension de la couche d'entrée est de n neurones, celle de sortie de p . Il y a donc $n \times p$ poids dans ce réseau.

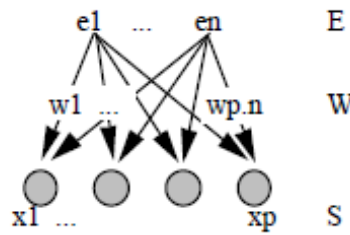


Figure 2.13 : Structure d'une mémoire associative

Le principe de fonctionnement d'une mémoire associative se résume ainsi. Soit $(X_1, X_2, \dots, X_n, \dots)$ un ensemble de vecteurs de R^n . A chaque vecteur X_1 appelé "prototype" de l'espace d'entrée est associé un vecteur de sortie Y_1 . La relation d'association entre X_1 et Y_1 est linéaire. Elle est donnée par l'Eq. 2.16 :

$$Y_1 = W X_1 \quad (2.16)$$

Et l'équation du poids est tirée de l'Eq. 2.16 :

$$W = Y_1 X_1^T \quad (2.17)$$

Où W est la matrice des poids de dimension $p \times n$. C'est une matrice rectangulaire de p lignes et n colonnes.

Le nombre d'apprentissage est obtenu par :

$$M = \sum_{k=1}^q W(k) \quad (2.18)$$

Où q désigne le nombre du modèle d'association entrée/sortie. L'objectif est de faire réaliser à ce réseau des associations entre les vecteurs d'entrées et les vecteurs de sortie désirés c'est-à-dire chaque entrée a une sortie correspondante (X , Y).

L'apprentissage est de type supervisé. La base d'apprentissage est composée de couple de vecteurs d'entrée et des vecteurs de sortie associés. L'algorithme d'apprentissage initial fait appel à la règle de Hebb. Une entrée X_1 est appliquée sur les neurones d'entrée du réseau et en le même temps les valeurs des neurones de sortie sont forcées à Y_1 . Les poids de chaque connexion sont alors modifiés selon la coactivité du neurone entrée et du neurone sortie. Cet algorithme est itéré sur tous les exemples de la base d'apprentissage.

Durant l'apprentissage, la matrice M enregistre la correspondance entre les modèles d'entrées et de sorties suivant la relation suivante :

$$M = YX^T \quad (2.19)$$

Où X^T est la transposée du vecteur X

A la fin du processus d'apprentissage, si la matrice W est initialement nulle ($W = 0$), elle sera de la forme :

$$W = \sum_1^q Y_q X_q^T \quad (2.20)$$

Après cet entraînement, il suffit d'entrer les vecteurs d'entrée pour obtenir le résultat voulu dont le poids utilisé est le poids W obtenu précédent. L'équation de la reconnaissance s'écrit ainsi :

$$Y = WX_i \quad (2.21)$$

La reconnaissance est parfaite si :

$$Y \cong Y_j \quad (2.22)$$

2.3. Conclusion

Le signal de parole est capté par un microphone pour pouvoir le numériser. Il va subir ensuite beaucoup de prétraitements en vue de bien le manipuler pour l'utilisation informatique. En effet, ce signal numérisé constitue beaucoup d'informations qui différencient les uns aux autres. Pour pouvoir faire une RAP, il est nécessaire d'extraire ces caractéristiques dont la méthode cepstrale fournit des résultats prometteurs. La RAP nécessite un moyen d'apprendre automatiquement les significations de ses caractéristiques, le RNA s'avère une bonne méthode pour cette tâche. Le RNA a des différentes variances (forme, algorithme) dont le choix dépend du résultat voulu.

CHAPITRE 3: REALISATION DU PROJET :

COMMANDE D'UN ROBOT MOBILE PAR LA PAROLE

Ce dernier chapitre décrit les différentes étapes de la réalisation du projet. La section 4.1 présente le système de commande qui intègre un ordinateur, un robot mobile et deux utilisateurs qui commandent le robot. Les sections 4.2 et 4.3 illustrent comment se déroule toute la réalisation en décrivant un à un tous les modules électroniques et informatiques composant le système.

3.1. Présentation du projet

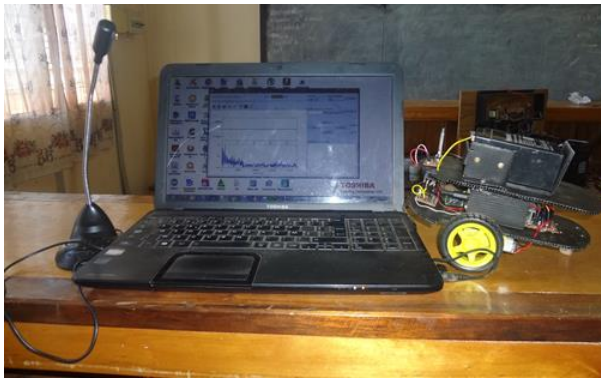
L'objectif du projet est de concevoir un robot mobile commandé, à l'aide des commandes vocales et des commandes par ordinateur.

La commande par ordinateur étant la commande par défaut la plus utilisée jusqu'à ce que la commande vocale ait lieu. C'est donc cette dernière qui priorise la commande et annule l'intervention d'autre commande. La figure 3.1 présente la structure générale de la commande à distance en temps réel.

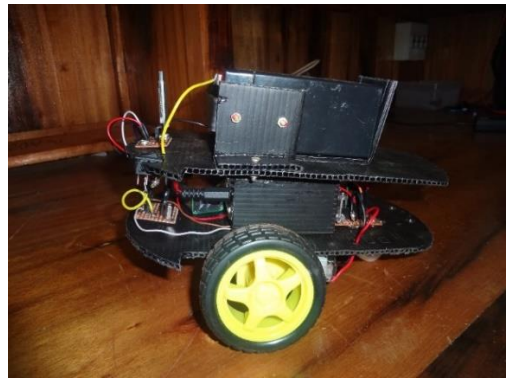


Figure 3.1 : Structure générale de la commande à distance

Le projet est composé d'un ordinateur et un micro dans la partie commande (Fig.3.2.a)) et d'un robot mobile dans la partie à commander (Fig.3.2.b)).



a)



b)

Figure 3.2: Photo du projet a) L'ensemble du système b) Le robot à commander

L'ordinateur permet la communication entre l'utilisateur et le robot à travers l'interface graphique contenant toutes les commandes nécessaires à la mise en marche du robot. Le micro sert à acquérir le signal de la parole.

Le robot mobile contient un circuit de commande et des actionneurs mécaniques alimentés respectivement par une batterie 6 V 4.5 AH et une tension 5 V sortant du port Universal Serial Bus (USB) de l'ordinateur.

3.2. Présentation du côté robot à commander

3.2.1. Schéma du circuit

La figure 3.3 présente le schéma du circuit. Les éléments composant le circuit seront présentés un à un comme la carte Arduino UNO Rev. 3, le driver demi-pont en H L293D, le motoréducteur et le module Bluetooth HC-05.

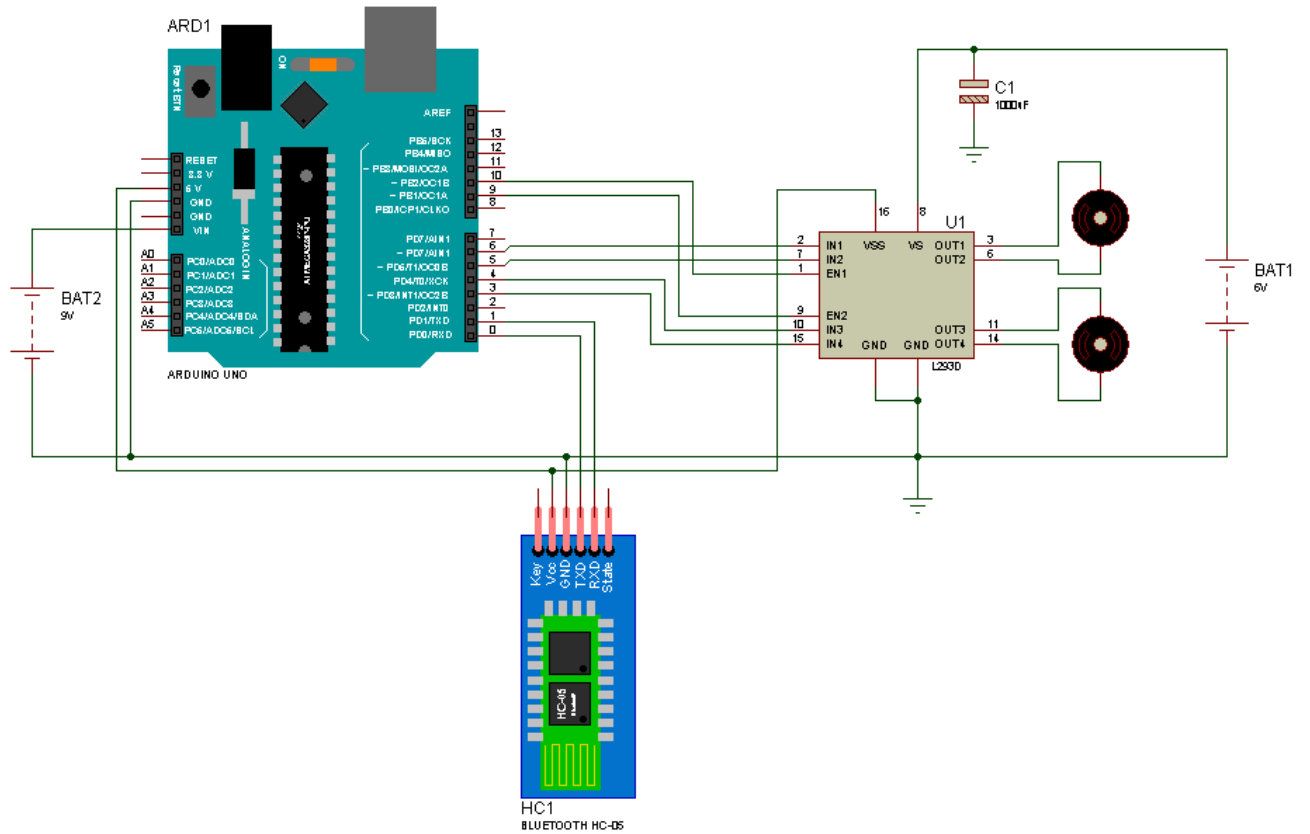


Figure 3.3 : Schéma du circuit

3.2.2. Arduino Uno

a. Présentation de la carte Arduino

Arduino est une plate-forme de prototypage d'objets interactifs à usage créatif constituée d'une carte électronique et d'un environnement de programmation. Arduino est conçu pour être accessible à tous par sa simplicité.

i. Matériel

La carte Arduino utilisée est l'Arduino UNO Révision 3. Cette carte, au format standard c'est-à-dire mesuré environ 52 mm sur 65 mm, est équipée d'un microcontrôleur Atmel de type ATmega328 (Fig. 3.4) cadencé sur la fréquence d'horloge de 16 MHz.

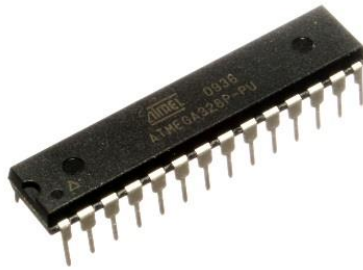


Figure 3.4 : ATmega328P-PU

Le microcontrôleur dispose d'une mémoire Flash 32 KB dont 0.5 KB est utilisée par le Bootloader, d'une mémoire SRAM 2 KB et d'une mémoire EEPROM 1 KB.

L'Arduino est alimentée soit avec une tension d'alimentation interne de 5 V soit avec une tension d'alimentation externe comprise entre 7 V et 12 V par le jack d'alimentation pour une utilisation autonome dont la pile 9 V est utilisée pour l'alimentation. La carte fournit à la sortie de chaque broche un courant max de 40 mA et à la sortie de toutes les broches un courant max cumulé de 200 mA. La carte s'interface à l'ordinateur par l'intermédiaire de sa prise USB ou par d'autres périphériques sans fil comme le module bluetooth. La figure 3.5 [17] montre le schéma décrivant la carte Arduino UNO Révision 3.

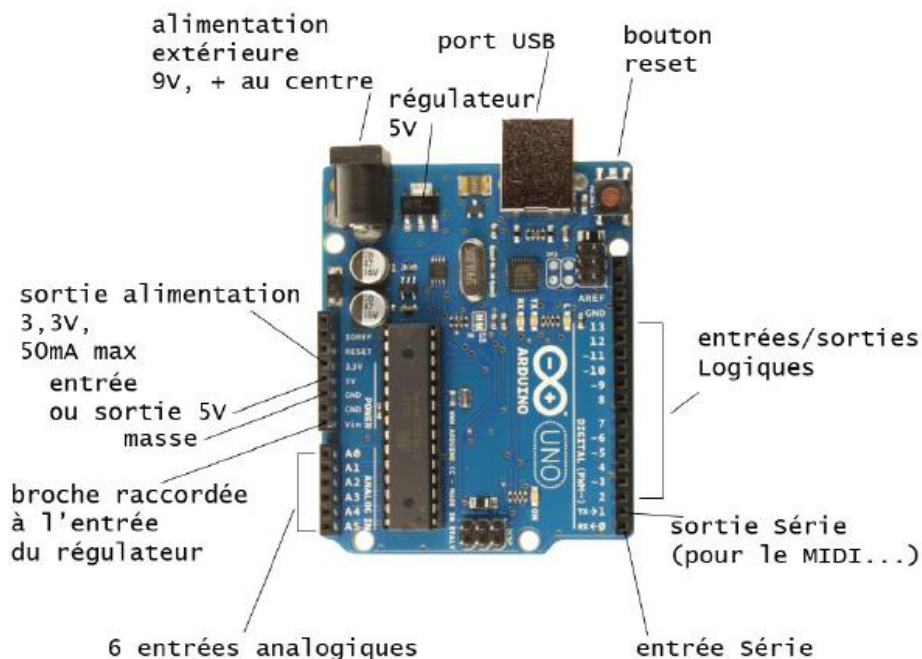


Figure 3.5 : La carte Arduino UNO Révision 3 [17]

ii. Logiciel

L'Environnement de Développement Intégré Arduino (EDI) est une application écrite en Java inspirée du langage Processing. L'EDI permet d'écrire, de modifier un programme

et de le convertir en une série d'instructions compréhensibles pour la carte. La figure 3.6 présente l'interface de l'environnement de programmation Arduino.

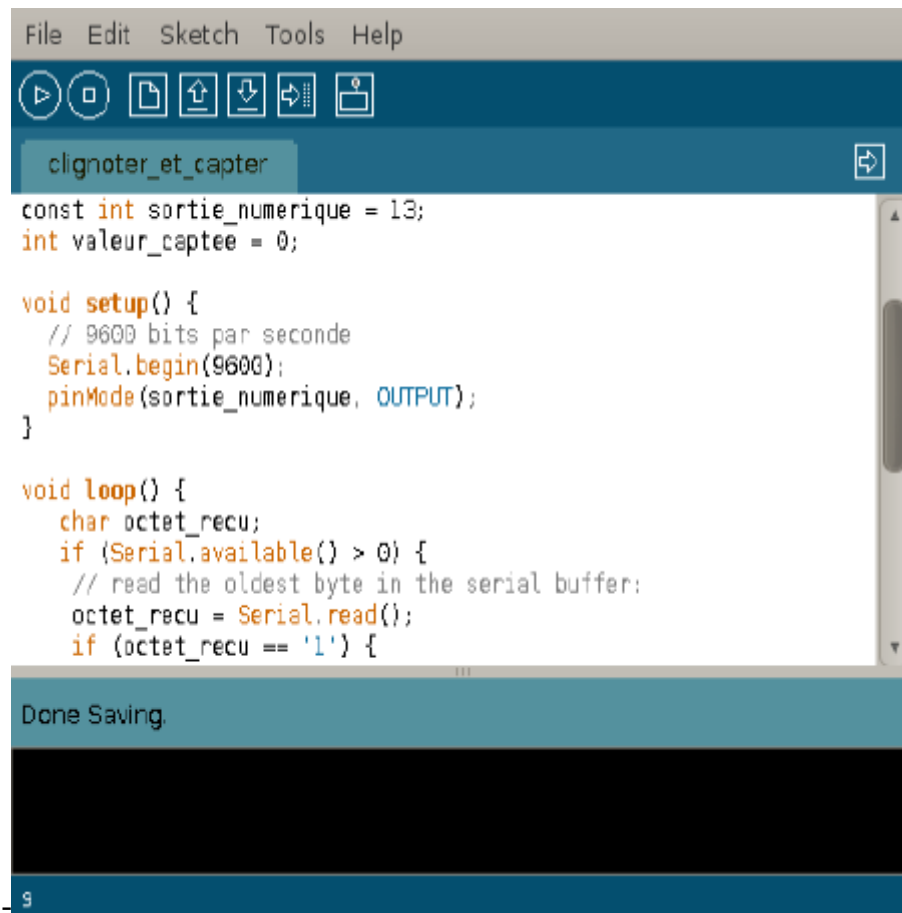


Figure 3.6 : L'interface de l'environnement de développement Arduino

b. La programmation Atmel ATmega328

Le rôle du microcontrôleur est de recevoir et interpréter les consignes issues du programme de commande. La carte électronique se charge de commander chaque moteur. Selon l'organigramme de commande des moteurs de la Fig. 3.7, le microcontrôleur doit être tout d'abord initialisé pour permettre son fonctionnement. Ensuite, des données en provenance de l'ordinateur à travers le port USB sont reçues. Ce sont la commande du sens de rotation du moteur et la vitesse du moteur. Enfin, selon les données reçues, le programme se charge d'envoyer des signaux à la sortie du microcontrôleur pour gérer la direction du moteur, c'est-à-dire sa marche avant, sa marche arrière, rotation ou frein.

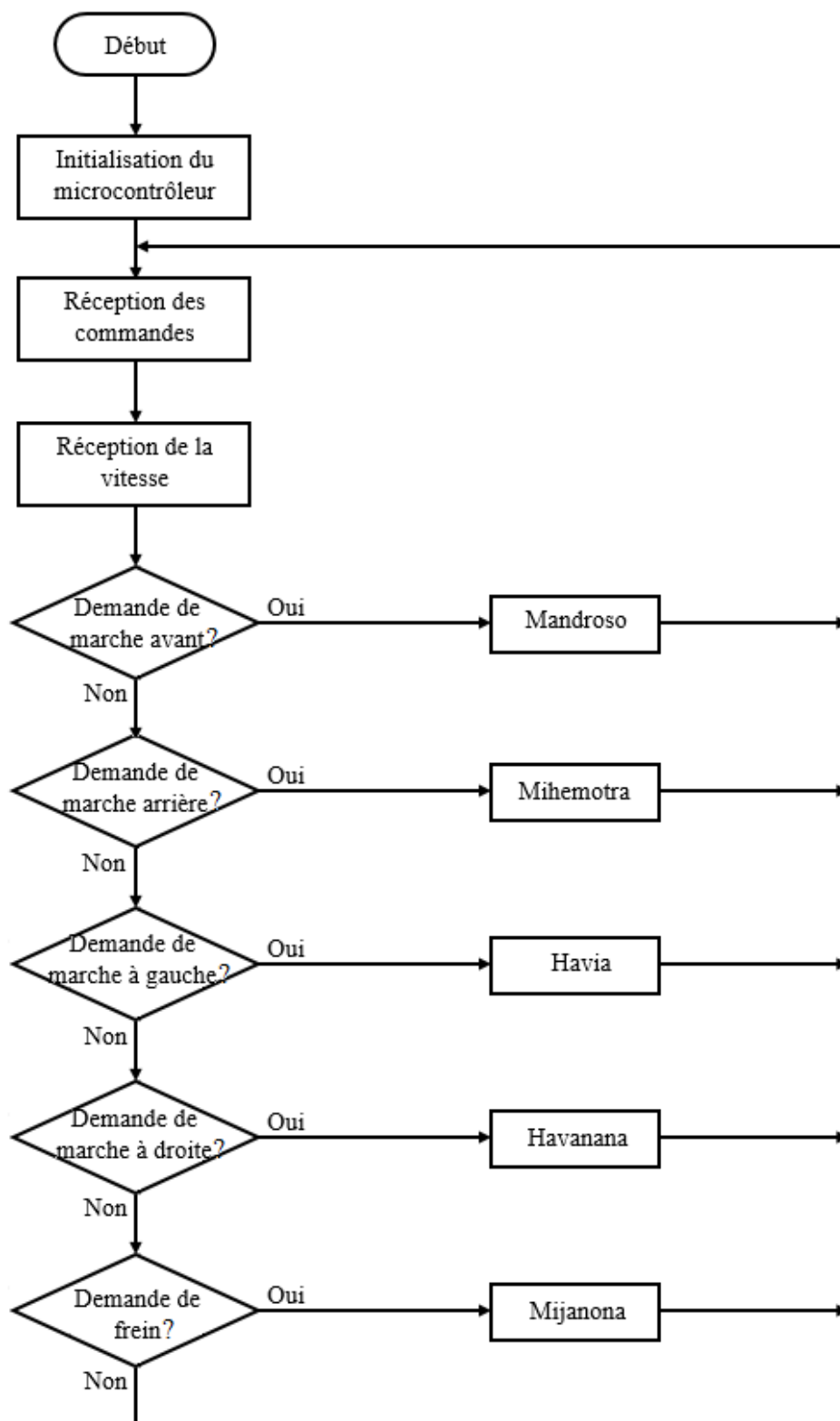


Figure 3.7 : Organigramme de commande du robot

3.2.3. Le driver demi-pont en H L293D

Le L293D est un driver à quadruple demi-pont en H qui peut fournir à sa sortie un courant bidirectionnel de forte intensité jusqu'à 600 mA par canaux sous la tension d'alimentation de 4.5 V à 36 V. Un courant de pic peut être toléré allant jusqu'à 1,2 A par canaux. Le L293D intègre déjà les diodes en parallèle des transistors. Cela implique donc des concessions sur les caractéristiques (le courant max passe à 1A par canaux). Sur la figure 3.8, ce composant a 16 broches et fonctionne selon un système de symétrie assez simple.

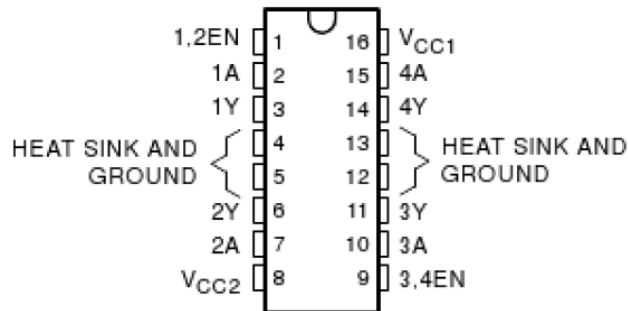


Figure 3.8 : Le driver L293D

Le L293D est conçu pour conduire les charges inductives telles que des relais, les solénoïdes, les moteurs à courant continu et les moteurs pas-à-pas, et aussi bien que d'autres charges à forte intensité/à haute tension dans des applications à alimentation positive.

De chaque côté les broches du milieu (4, 5, 12 et 13) servent à relier la masse mais aussi à dissiper la chaleur. Les entrées d'activation des ponts (Enable) se trouvent sur les broches 1 et 9. Un état HAUT sur ces broches et les ponts seront activés, les transistors pourront s'ouvrir ou se fermer, alors qu'un état BAS désactive les ponts, les transistors restent ouverts. Le tableau I présente les cas possibles de la commande et leurs actions.

Tableau I : Commande et impact sur le moteur

Input 1 (broche 2 et 10)	Input 2 (broche 7 et 15)	Effet
0	1	Tourne dans le sens horaire
1	0	Tourne dans le sens antihoraire
0	0	Frein
1	1	Frein

Ainsi, en utilisant une MLI sur la broche d'activation des ponts, la vitesse du moteur peut être variée. Il ne reste plus qu'à brancher le moteur sur les sorties respectives (2 et 7 ou 11 et 14 selon le pont utilisé) pour le voir tourner.

Le composant possède deux sources d'alimentation : une pour la partie "logique" (contrôle correct des transistors), V_{CC1} et l'autre pour la partie puissance (utile pour alimenter les moteurs à la bonne tension), V_{CC2} . Ces deux entrées respectent les mêmes tensions (4.5V à 36V). La présence du gros condensateur polarisé (1 μ F ou plus selon l'alimentation) est utile pour découpler l'alimentation de puissance du L293D.

3.2.4. Le motoréducteur

Le pilotage du motoréducteur fonctionne en boucle ouverte. Selon le principe de la Fig. 3.9, L'ordinateur va envoyer le signal via Bluetooth pour permettre à l'Arduino de générer le signal MLI selon la commande. L'Arduino va ensuite commander le driver de puissance et ce dernier le motoréducteur :

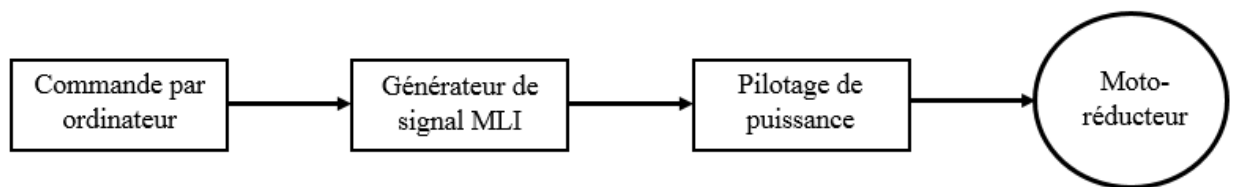


Figure 3.9 : Principe de commande du motoréducteur

La figure 3.10 présente l'ensemble moteur et son réducteur à cinq pignons à engrenages plastiques. Le rapport de réduction vaut en tout 48. Compte tenu de la consommation modérée de la part du moteur, la solution est d'utiliser une batterie rechargeable de 6 V - 4.5 AH.



Figure 3.10 : Photo du motoréducteur

3.2.5. Le module Bluetooth

i. La communication Bluetooth

Le Bluetooth est un protocole de communication sans fil. Il a vu le jour à la fin des années 1990 et n'a vraiment percé que dans les années 2000. Il a subi de nombreuses

révisions et évolutions pour atteindre aujourd'hui la version 4.1 depuis la fin 2013. Il respecte la spécification Institute of Electrical and Electronics Engineers (IEEE) et utilise la gamme de fréquences : 2.4 GHz. C'est une communication bidirectionnelle, deux modules peuvent communiquer ensemble en même temps. Le comportement utilisé est « maître/esclave ». Un esclave pourra parler avec un seul maître, mais un maître pourra dialoguer avec plusieurs esclaves [18].

Pour son utilisation, elle se passe en plusieurs étapes :

- Le maître se met en mode « reconnaissable »
- L'esclave trouve le maître et demande à s'y connecter
- Le maître accepte la connexion
- Les périphériques sont alors appairés (ou associés)
- La communication peut commencer

Ensuite, la communication pourra se faire selon un protocole ou un autre. Ici, cela consistera simplement en une liaison série.

ii. Présentation du module Bluetooth HC-05

Le module intègre la conversion 5V en 3.3V, la régulation de l'alimentation (3.3V) et l'ajout de Diodes ElectroLuminescentes (DELs) de signal. Les broches VCC et GND sont pour l'alimentation (5V) et la masse, Rx/Tx pour la communication. Il y a aussi une broche Key qui servira à envoyer des commandes de configuration au module. Avec cette broche, la voie série (vitesse, parité, etc.) peut être reconfigurée ainsi d'autres options liées au Bluetooth (nom du module, mot de passe d'appairage, mode esclave/maître, etc.). La dernière broche nommée State permet de brancher une DEL pour obtenir un signal sur l'état du module. La figure 3.11 présente le module Bluetooth de type HC-05.

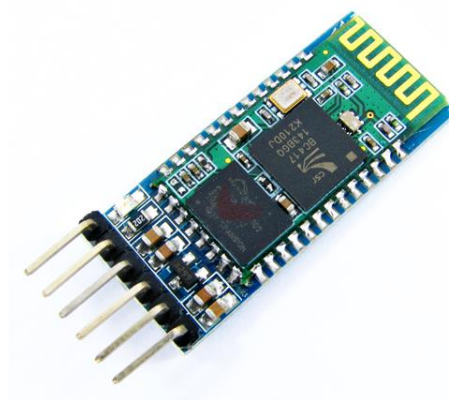


Figure 3.11 : Le module Bluetooth HC-05

3.3. Présentation du côté ordinateur et interface

Le robot est contrôlé à distance via un Bluetooth permettant deux modes de saisie d'ordres. Un premier mode, le mode manuel géré grâce à une interface graphique présente sur un ordinateur, et un second mode le mode vocal qui se fait par l'intermédiaire d'un microphone lié à l'ordinateur.

Le mode vocal nécessite une programmation préalable pour qu'un utilisateur puisse envoyer des ordres. L'utilisateur peut choisir le clavier ou la souris comme mode de saisie de l'interface graphique du mode manuel.

3.3.1. Le langage utilisé

Pour le développement de l'application (traitement de signal de parole, reconnaissance de parole, commande du robot ainsi que l'interface graphique), Python version 3.4 est le langage utilisé.

Python est un langage portable, dynamique, extensible, gratuit, et orientée objet de la programmation. Python est développé depuis 1989 par Guido van Rossum et de nombreux contributeurs bénévoles, il est riche en modules (bibliothèques) qui permettent de réaliser un grand projet [19].

Voici donc les listes de module utilisés pour achever la programmation :

- numpy : calcul numériques
- matplotlib : pour pouvoir tracer les graphes
- scipy : calculs scientifiques
- PyQt4 : interface graphique
- pyaudio : enregistrement audio
- features : extraction de caractéristiques les coefficients MFCC
- pybrain : module apprentissage automatique, bibliothèque de l'intelligence Artificiel et RNA
- scikit-learn : module apprentissage automatique pour visualiser la performance de l'apprentissage
- pyserial : communication série
- auditok : détection du signal utile

3.3.2. La commande par la parole

Pour la RAP, l'outil utilisé est le RNA de type PMC. Le robot respecte à cinq ordres différents : mandroso, mihemotra, havia, havanana, mijanona.

a. Calcul des coefficients MFCC

Les étapes suivies sont décrites en chapitre 2 pour arriver au résultat voulu. Dans la phase de prétraitement, une détection de voix a été faite dans le but d'éliminer le silence avant et après la parole. Puis le signal est segmenté en des fenêtres de 30ms avec 10ms de distance, chaque segment est multiplié par la fenêtre de Hamming avant de le rendre dans le domaine fréquentiel. Après, la TCDI est appliquée au segment pour avoir les MFCC. Le module utilisé est « auditok » pour la détection de l'activité vocale et le module « features » pour l'extraction de caractéristiques MFCC. Voici la syntaxe permettant d'obtenir les coefficients MFCC :

Coefficients = mfcc (signal, fréquence d'échantillonnage)

La variable ***Coefficients*** contient 13 coefficients MFCC de chaque segment (frame) d'un signal. Donc, pour un signal de parole traité, 13*nombre de frames coefficients MFCC sont obtenus. Normalement, ces caractéristiques servent à l'entrée du classificateur. En prenant le nombre de frames à 256, il y a 13*256 coefficients MFCC pour un mot. Cette valeur est très grande pour que la machine puisse créer le classificateur. En d'autre mot, cela attarderait la durée d'apprentissage. Au lieu de faire cela, il est raisonnable de prendre la valeur moyenne de ces 13*256 coefficients pour chaque signal comme nombre d'entrée du classificateur en utilisant la syntaxe suivante :

X = np.mean (Coefficients [int (nombre_segment*0.1):int(nombre_segment*0.9)], axis=0)

b. Le RNA utilisé

1000 mots sont enregistrés pour le corpus d'exemple donné au RNA comme données d'apprentissage qui se répartissent en 200 enregistrements pour chaque classe et 250 enregistrements pour le test (50 chacune). Plus le nombre de base d'apprentissage est élevé plus l'erreur est faible.

Le nombre de neurone de la couche d'entrée est égal à 13, Il est évident que le nombre de neurone de la couche de sortie correspond aux nombres de classe (5 commandes). Une

seule couche cachée suffit pour réaliser le travail dont le nombre de neurone est 10 (nombre favorable après plusieurs apprentissages). Le taux de reconnaissance est 92.33 % pour l'apprentissage et 91.7 % pour le test avec un pas d'apprentissage de 0.01. La figure 3.12 représente la courbe de convergence de l'erreur d'apprentissage du réseau de neurone utilisé. Le taux d'erreur d'apprentissage ainsi que le taux d'erreur de test convergent et ceci veut dire que la machine a bien appris.

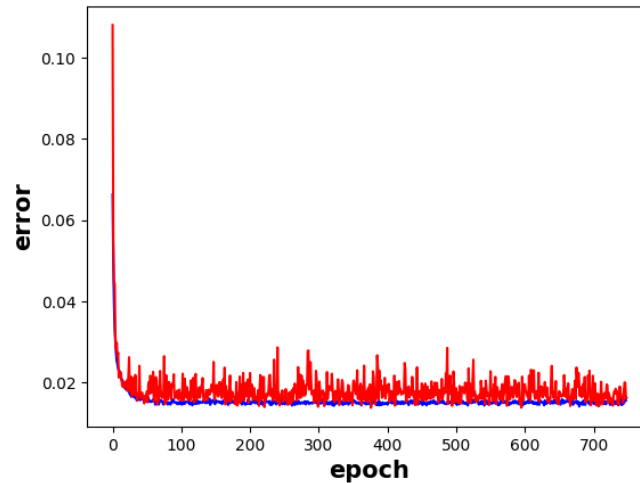


Figure 3.12 : Courbes représentatives d'apprentissage de classeur RNA

La courbe en bleue correspond à l'erreur d'apprentissage, et celle en rouge représente l'erreur du test. La qualité du système de classification est visualisée par l'outil de performance appelé « matrice de confusion » (Fig. 3.13). Le diagonal de cette matrice représente le pourcentage de succès de chaque classe et les autres correspondent au pourcentage des fausses classifications.

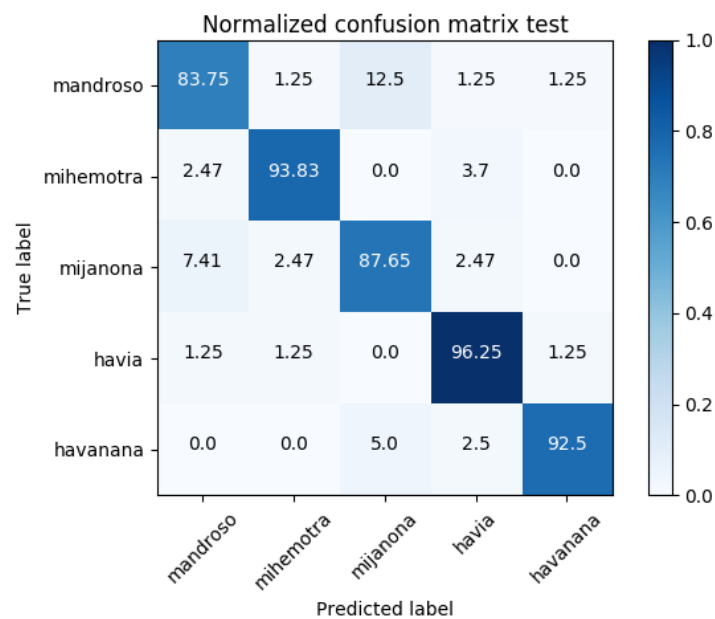


Figure 3.13 : Matrice de confusion

3.3.3. L'interface de commande

L'application développée a une interface utilisateur qui permet de commander le robot. Elle se divise en deux parties dont la première représente une visualisation de spectre lorsqu'une l'utilisateur active la commande vocale, la seconde partie (droite) affiche les différentes caractéristiques de la commande par ordinateur et ses boutons des commandes. Cette interface est présentée à la Fig. 3.14.

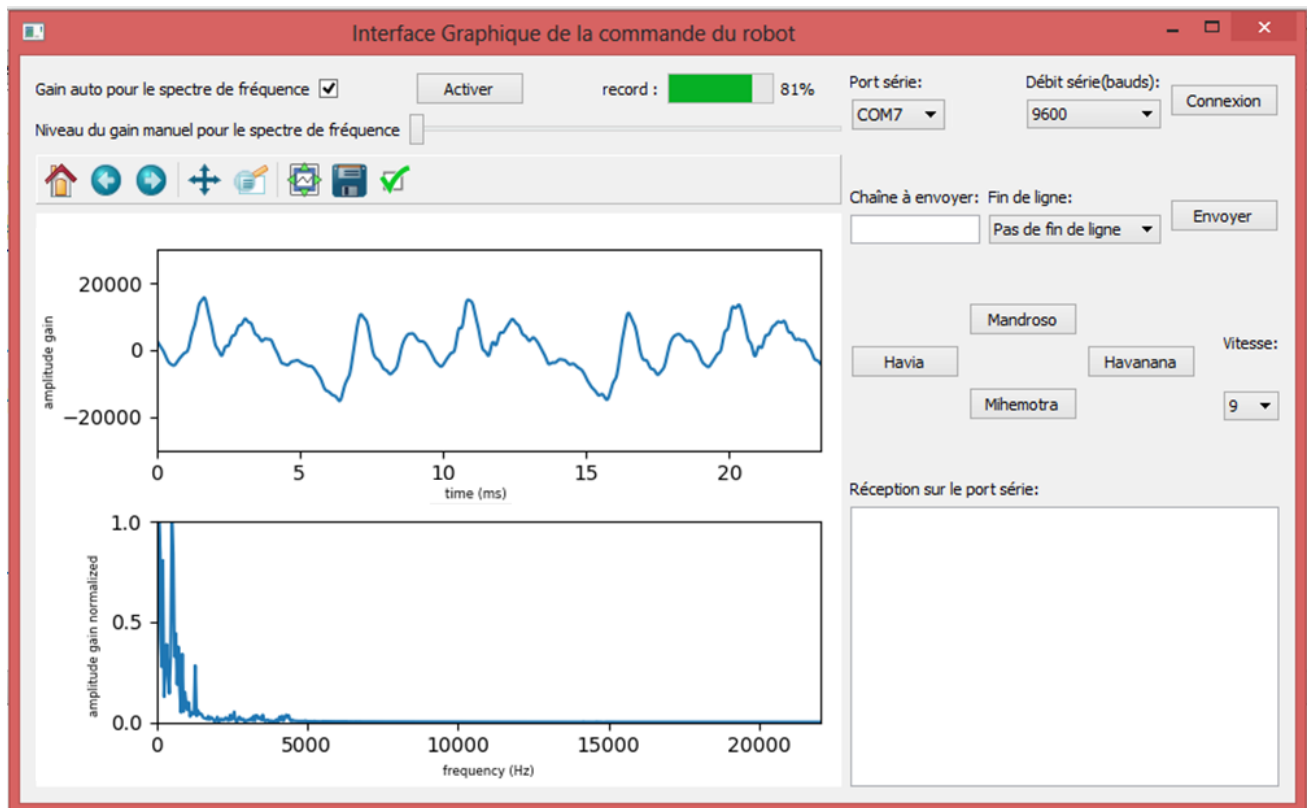


Figure 3.14 : Interface de commande du robot

La commande vocale ne nécessite aucune action directe par l'utilisateur sauf la mise en marche du microphone en appuyant le bouton « Activer ». Voici l'organigramme de la commande vocale du robot présentée par la Fig. 3.15.

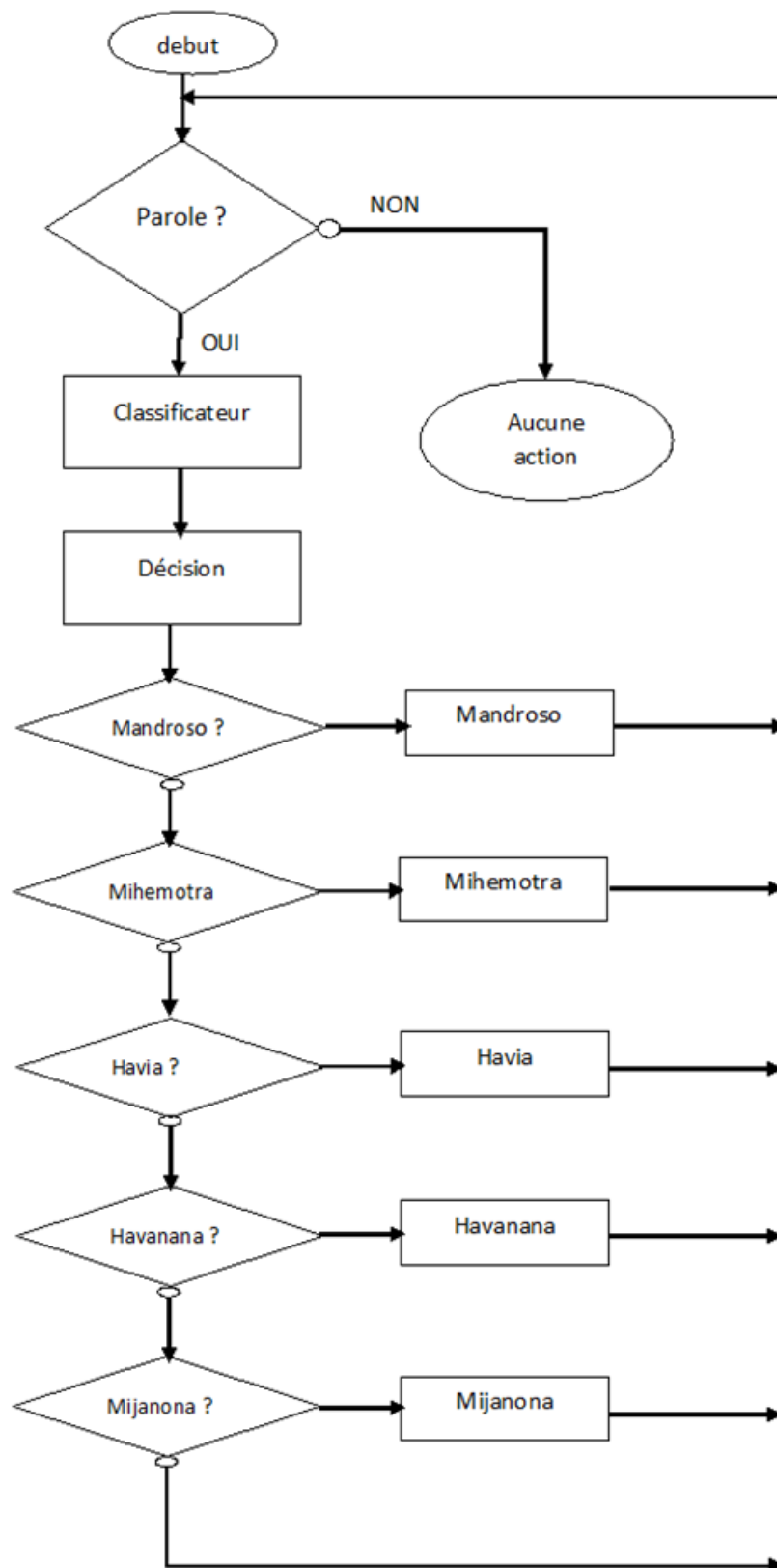


Figure 3.15 : Organigramme de commande vocale

En ce qui concerne la commande par ordinateur, le bouton cliqué correspondant à la commande voulue, Un clic prolongé entraîne une action prolongée ; il suffit de relâcher le bouton pour annuler la commande du robot.

3.4. Conclusion

Une étude approfondie est nécessaire pour réaliser un système de reconnaissance automatique de la parole et de l'utiliser pour commander un robot mobile. Afin de mener à bien la réalisation, il faut tout d'abord connaître les caractéristiques du moteur du robot à utiliser. Le module Arduino est ensuite programmé. Après la carte électronique et la partie mécanique sont conçues en fonction des spécificités du moteur. La performance du système RAP doit être assurée. Enfin pour la commande par la parole et à distance, il faut développer l'interface de commande avec le langage Python et établir la liaison Bluetooth entre l'ordinateur et le robot mobile.

CONCLUSION GENERALE

La parole est un signal sonore, continu et non stationnaire, elle se produit par la vibration de cordes vocales humaines. Elle se caractérise par les sons voisés et ceux non voisés. Le signal de parole peut se mesurer à partir des paramètres tels que la fréquence fondamentale, la hauteur, la durée et l'intensité. L'une de ses utilisations intéressantes est de commander un robot mobile. Les études et les techniques de mise en œuvre de ce dernier ne cessent d'évoluer pour le rendre utile dans la vie quotidienne. De ce fait, la conception est complexe en combinant des notions mécanique, électronique, automatique et informatique.

Pour acquérir le signal de parole, il est capté par un microphone pour pouvoir le numériser. Il a subi de prétraitements en vue de bien le manipuler par l'utilisation informatique. Pour pouvoir faire une RAP, il est nécessaire d'extraire ces caractéristiques par la méthode cepstrale. Le RNA est utilisé pour l'apprentissage des caractéristiques.

Une étude approfondie est nécessaire pour réaliser un système de reconnaissance automatique de la parole et de l'utiliser pour commander un robot mobile. Connaissant les paramètres du moteur et des composants constituant la carte électronique, le système embarqué est conçu selon les besoins fonctionnels. Le développement de l'interface de commande pour la commande à distance du robot permet l'interaction de l'utilisateur avec le milieu qui l'entoure et aussi de protéger les personnes qui travaillent dans un milieu hostile.

Actuellement, la recherche dans le domaine de la RAP est en cours de développement. En effet, le point commun est le taux de reconnaissance qui n'atteint pas le pourcentage à 100% pour une commande parfaite d'un robot mobile. En fait, un autre modèle a déjà progressé dans la RAP avant la venue de la RNA, c'est le MMC. Il est alors envisageable de l'utiliser pour la commande du robot mobile. Est-ce que cette méthode permet d'avoir une précision meilleure sur l'apprentissage ?

ANNEXE 1 : LE VAD

La détection d'activité vocale ou VAD est définie comme le processus qui consiste, à partir d'un signal audio, à différencier les portions qui contiennent de la voix à celles qui n'en contiennent pas [20].

Par convention, la sortie d'un VAD est '1' si l'activité de la voix est présente, ou '0' sinon. Dans la littérature, il y a la région active (région qui contient de la voix) et la région inactive (région sans voix). La figure A.1 montre le résultat de VAD d'une suite de parole.

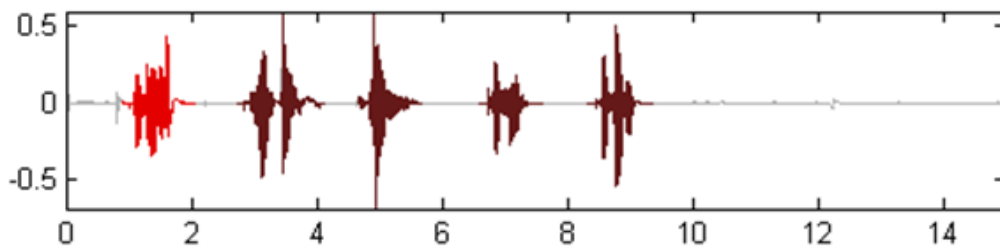


Figure A. 1 : Résultat de VAD d'une suite de parole

La grande majorité des VAD découpent le signal audio d'entrée en trames. Une trame se définit comme une portion de signal de longueur fixe et de l'ordre de quelques millisecondes (habituellement de l'ordre de 10 à 50 ms). Le VAD va décider pour chaque trame si elle est active ou inactive. Il y a plusieurs algorithmes de VAD mais ils suivent de différentes manières les mêmes étapes suivantes (Fig. A.2) :

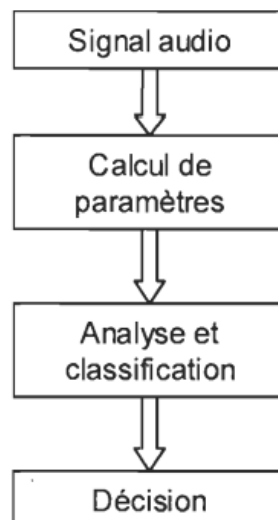


Figure A. 2 : Etapes de l'algorithme VAD

Soit un signal discret de m échantillons, les paramètres peuvent être calculés dans le domaine temporel (l'énergie du signal $E_s(m)$, la puissance du signal $P_s(m)$ et le rapport de

passage par zéro S_m .) ou à partir du domaine spectral (TFD, analyse de forme spectrale, ondelettes, etc.). À partir de ces paramètres, une déduction est tirée si la sortie vaut '0' ou '1' (respectivement inactivité ou activité vocale). Pour un signal de parole $s(n)$, les mesures de ces paramètres dans le domaine temporel sont calculées en utilisant les Eq. A.1, A.2, A.3 :

Pour l'énergie du signal :

$$E_s(m) = \sum_{n=m-L+1}^m s^2(n) \quad (A.1)$$

Avec L la largeur des frames

Pour la puissance du signal :

$$P_s(m) = \frac{1}{L} E_s(m) \quad (A.2)$$

Pour le rapport de passage par zéro :

$$S_m = \frac{1}{L} \sum_{n=m-L+1}^m |sgn(s(n)) - sgn(s(n-1))| \quad (A.3)$$

Où sgn est la fonction signe telle que :

$$sgn(s(n)) = \begin{cases} +1 & \text{si } s(n) \geq 0 \\ -1 & \text{si } s(n) < 0 \end{cases} \quad (A.4)$$

Il est à noter que l'index pour ces fonctions est m et pas n , car ces mesures ne sont pas calculées pour chaque échantillon. L'énergie s'accroît quand le signal $s(n)$ contient de la parole et c'est le cas aussi pour la puissance. Le rapport de passage par zéro donne une mesure du nombre de fois où le signal $s(n)$ change de signe. Ce rapport est en général plus grand dans les régions non voisées [21].

ANNEXE 2 : DATASHEET DU MOTEUR

La figure A.3 montre le datasheet d'un MCC (Moteur à Courant Continu) FC-130RA/SA provenant de Mabuchi Motor. Le moteur choisi est de type FC-130SA-14150 de tension nominale 6V et de rendement 42 % [28].



FC-130RA/SA

OUTPUT: APPROX 0.2W-10W

Carbon-brush motors

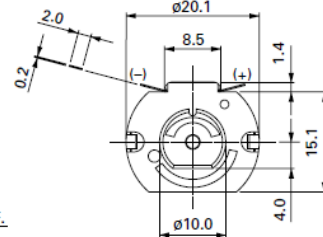
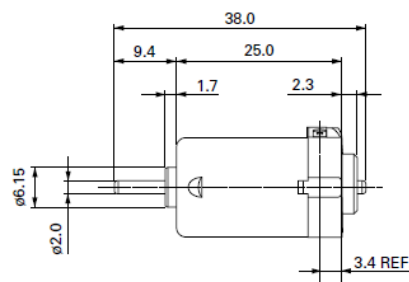
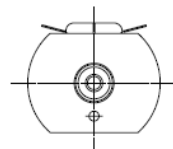
Typical Applications Household Appliances : Styling Brush
 Toys and Models : Motorized Toy / Radio Control Model



MODEL	VOLTAGE		NO LOAD		AT MAXIMUM EFFICIENCY					STALL		
	OPERATING RANGE	NOMINAL	SPEED r/min	CURRENT A	SPEED r/min	CURRENT A	TORQUE mN·m	TORQUE g·cm	OUTPUT W	TORQUE mN·m	TORQUE g·cm	CURRENT A
FC-130RA-14150	4.5 ~ 6.0	4.5V CONSTANT	13200	0.14	9890	0.42	0.74	7.5	0.76	2.94	30	1.25
FC-130RA-10300	4.5 ~ 12.0	12V CONSTANT	18000	0.080	13770	0.26	0.90	9.1	1.29	3.82	39	0.85
FC-130SA-14150	4.5 ~ 6.0	6V CONSTANT	12400	0.14	9560	0.47	1.19	12.1	1.19	5.19	53	1.59
FC-130SA-2270	3.0 ~ 4.5	3V CONSTANT	13500	0.27	10740	1.05	1.37	14.0	1.54	6.71	69	4.10

UNIT: MILLIMETERS

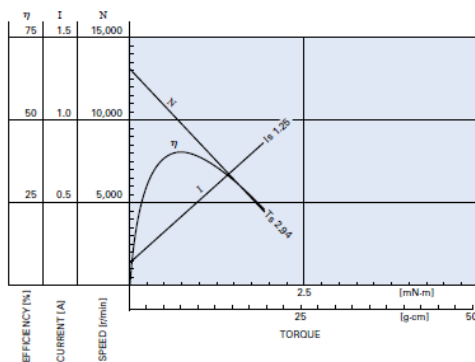
DIRECTION OF ROTATION



WEIGHT: 17g (APPROX)

FC-130RA-14150

4.5V



FC-130SA-14150

6.0V

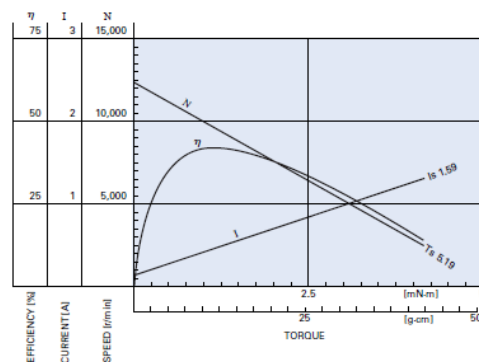


Figure A. 3 : Datasheet du MCC FC-130RA/SA [22]

ANNEXE 3 : LE BLUETOOTH

Le Bluetooth est une technologie permettant de connecter n'importe quels périphériques sans utiliser de fil, en utilisant les ondes radios avec une plage de fréquences variant entre 2.4 et 2.483 GHz [23].

– Origine du nom

Le nom Bluetooth, qui signifie en anglais "Dent bleue", provient du roi danois Harald 1^{er}, qui est connu pour avoir unifié les pays scandinaves. Son surnom était "Harald à la dent bleue".

– Histoire

Le Bluetooth a été inventé par la société Ericsson, en 1994, à Lund en Suède.

En 1998, la Bluetooth Special Interest Group (SIG) est fondé. Il s'agit de la collaboration entre Ericsson et plusieurs autres grandes entreprises comme IBM, Intel, Nokia et encore Toshiba. Cette association va créer le Bluetooth 1.0 en juillet 1999.

En mars 2006, la SIG, qui s'est auparavant agrandi après que Motorola, Microsoft, Lucent et 3Com les aient rejoint en 1999, crée le Bluetooth 2.0.

La SIG compte désormais environ 1800 entreprises, et aujourd'hui, la quatrième version du Bluetooth est apparue.

– Evolution du Bluetooth

- Versions et normes du Bluetooth

Le Bluetooth en est au jour d'aujourd'hui à la version 4.0. Il y eut plusieurs versions antérieures apportant chacune une amélioration à la technologie.

Norme IEEE 802.15.1 :

- Bluetooth 1.0 et v1.0B
- Bluetooth v1.1 parut en 2002, appelée IEEE 802.15.1-2002

- Bluetooth v1.2 parut en 2005, appelée IEEE 802.15.1-2005

Norme IEEE 802.15.2 :

- Bluetooth v2.0 + Enhanced Data Rate (EDR) parut en 2004
- Bluetooth v2.1 + Enhanced Data Rate (EDR) parut en 2007

Norme IEEE 802.15.3 :

- Bluetooth v3.0 + High Speed (HS) parut en 2009

Norme IEEE 802.15.4 :

- Bluetooth v4.0 parut en 2010

- Différence entre les normes

Cette différence est illustrée dans le Tab. A.1 :

Tableau A. 1 : Différence entre les normes

Norme	Vitesse	Portée
IEEE 802.15.1	1Mbits/s	10 mètres
IEEE 802.15.2	2Mbits/s	10 mètres
IEEE 802.15.3	20Mbits/s	10 mètres
IEEE 802.15.4	1Mbits/s	50-60 mètres

Dans la norme IEEE 802.45.2, la version 2.1 a subit une amélioration de la sécurité grâce à la fonction Encryption Pause and Resume. La version 3.0 du Bluetooth utilise la fréquence radio 802.11 appartenant au Wi-Fi. La version 4.0 permet d'augmenter la portée pour les bas débits. Sur la vitesse, chaque version du Bluetooth apporte des progrès sur l'économie d'énergie et l'autonomie des appareils.

- **Les classes Bluetooth**

La portée des appareils utilisant la technologie Bluetooth dépend de la puissance en émission. Plus la puissance est élevée, plus la portée sera grande. Ces appareils sont répartis

en trois classes, en fonction de la puissance émise. La puissance maximale est de 100 mW pour une portée d'environ 100 mètres sans obstacle.

Le tableau A.2 décrit les classes Bluetooth :

Tableau A. 2 : Les classes Bluetooth

Classe	Puissance	Portée
1	100 mW	~100 mètres
2	2.5 mW	10 à 20 mètres
3	1 mW	< 10 mètres

La classe 1 est souvent utilisée pour les dongles (adaptateurs pour ordinateur). La classe 2 est la plus commune, la majorité des appareils utilisant la technologie Bluetooth sont de classe 2. La classe 3 est la plus faible, elle est utilisée en majorité par les téléphones portables.

– **Fonctionnement**

La technologie Bluetooth utilise la bande de fréquence 2.4 GHz - 2.483 GHz. Cette technologie utilise la technique Etalement de Spectre par Saut de Fréquence (ESSF). La technique ESSF permet de découper la bande de fréquence en 79 canaux, ayant chacune une largeur de 1 MHz. En changeant plus de 1600 fois par seconde de canal, le standard Bluetooth évite l'interférence avec les signaux des autres modules radios.

- **Méthode Maître/Esclave**

Le standard Bluetooth se connecte aux périphériques grâce à la méthode maître/esclaves. Un maître peut se connecter au maximum à 7 esclaves "actifs", ou 255 esclaves en mode "parked".

Le réseau formé par un périphérique Bluetooth et tous les autres périphériques Bluetooth à sa portée est appelé picoréseau. Un picoréseau possède un unique maître. Le périphérique maître ne peut pas se connecter à tous les périphériques esclaves en mêmes temps, il commute très rapidement d'un esclave à un autres. Toutefois, un périphérique esclave peut avoir plusieurs maîtres, cela permet de relier plusieurs picoréseau entre eux. Cela s'appelle un réseau chaîné.

- Connexion entre périphériques

La procédure de connexion entre deux périphériques Bluetooth, suit les étapes suivantes :

- Mode passif : Tous les périphériques sont passifs, c'est à dire qu'il n'est connecté à aucun autre périphérique.
- Phase d'inquisition : Durant cette phase, le périphérique maître envoie une requête à tous les autres périphériques à sa portée. Les autres périphériques lui répondent en envoyant leur adresse.
- Paging : Il s'agit d'une technique utilisée par le périphérique maître pour se synchroniser au périphérique esclave à l'adresse choisit. A travers cette méthode le périphérique maître synchronise son horloge et sa fréquence avec le périphérique esclave.
- Découverte des services : Le périphérique maître analyse les services du périphérique esclave grâce au protocole Service Discovery Protocol (SDP).
- Création de canal : Durant cette étape, le périphérique maître crée un canal de communication avec le périphérique esclave grâce au protocole Logical Link Control and Adaptation Protocol (L2CAP).
- Pairage : Il s'agit du système de sécurité du périphérique esclave. Cela permet de limiter l'accès aux autres périphériques. Le pairage se fait grâce à une clé de chiffrement appelé code Personnel Identification Number (PIN).

Une fois ces étapes réalisées, la connexion entre le périphérique maître et le périphérique esclave est accomplie.

ANNEXE 4 : LE LANGAGE PYTHON

– L’EDI PyCharm

L’EDI pour Python utilisé est le JetBrains PyCharm Community v2016.1.2. Il supporte le langage Python de version supérieur à 2.4 et il est multiplateforme.

Pour créer un projet, il suffit de cliquer sur le bouton « Create New Project » puis de choisir un interpréteur pour le langage Python. Un fichier de programme en langage Python porte l’extension « .py ». L’importation de module est faite via le mot clé « import ». L’interface de l’EDI PyCharm est présentée à la Fig. A.4 :

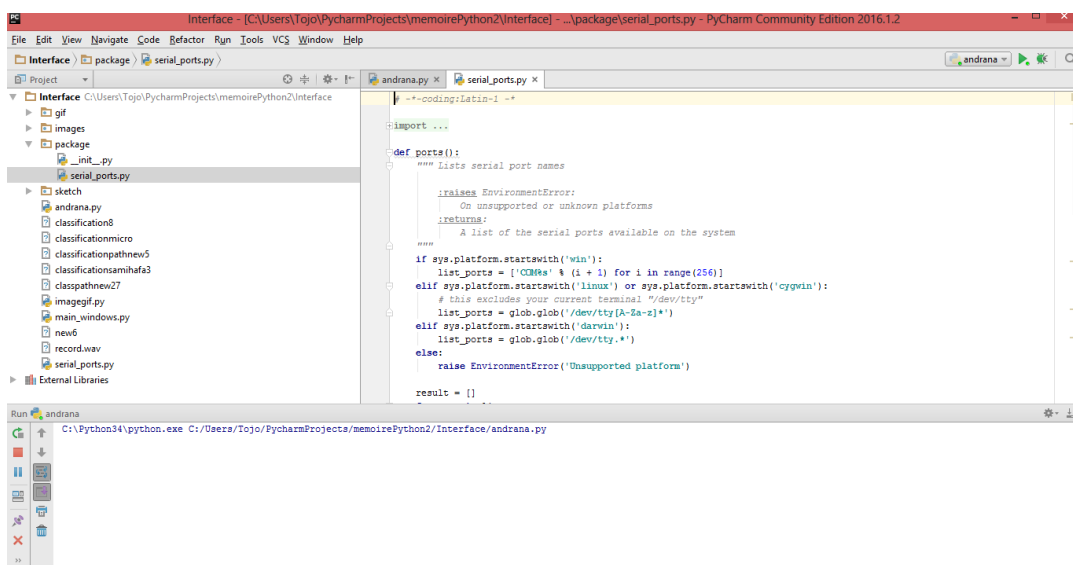


Figure A. 4 : L’interface de l’environnement de développement PyCharm

– Extraction des caractéristiques du signal de la parole

```
from features import mfcc
import scipy.io.wavfile as wav

(freq_echantillonnage, signal_echantilloné) = wav.read(signal)
mfcc_feat = mfcc(signal_echantilloné, freq_echantillonnage)
#pondération par coefficient
x =
np.mean(mfcc_feat[int(len(mfcc_feat)*0.1):int(len(mfcc_feat)*0.9)], axis=0)
```

– Apprentissage de RNA

```
import pickle
import matplotlib.pyplot as plt
import numpy as np
```

```

import itertools
from pybrain.utilities import percentError
from pybrain.tools.shortcuts import buildNetwork
from pybrain.supervised.trainers import BackpropTrainer
from pybrain.structure.modules import SoftmaxLayer
from pybrain.structure.modules import SigmoidLayer
from sklearn.metrics import confusion_matrix

#import des corpus d'exemple pour train and test
print ("Loading datatrain from file")
fileObject = open('datasettrain', 'rb')
datasettrain = pickle.load(fileObject)

print ("Loading datatest from file")
fileObject1 = open('datasettest', 'rb')
datasettest = pickle.load(fileObject1)

#normaliser classes
dataset._convertToOneOfMany(bounds=[0, 1])
datasettest._convertToOneOfMany(bounds=[0, 1])

#create network
fnn = buildNetwork(nbr_neurone_entrée,\
                   nbr_neurone_caché \
                   nbr_neurone_sortie,\
                   hiddenclass=SigmoidLayer,\
                   outclass=SoftmaxLayer)

#algo d'apprentissage
trainer = BackpropTrainer(fnn, dataset=dataset, learningrate=0.01,
momentum=0.1)

#entraîner ANN
a,b = trainer.trainUntilConvergence(verbose=True,
                                   trainingData=datasettrain,
                                   validationData=datasettest,
                                   maxEpochs= nbr_de_repetition)

#taux d'erreur
report_error(trainer, datasettrain, datasettest)
out = fnn.activateOnDataset(datasettrain).argmax ( axis =1)
out1 = fnn.activateOnDataset(datasettest).argmax ( axis =1)

#sauver classeur
print('saving neural network to file ...')
fileObject = open('classifieur', 'wb')
pickle.dump(fnn, fileObject)
fileObject.close()
print("saved")

#-----matrice de confusion test-----
cnf_matrix = confusion_matrix(targetst, out1)
np.set_printoptions(precision=2)
# Plot confusion matrix
plt.figure()

plot_confusion_matrix(cnf_matrix1, classes=classes_name,
                     title='Confusion matrix test')

plt.show()

#afficher allure d'apprentissage
plt.plot(a, 'b', b, 'r')
plt.show()

```

REFERENCE

- [1] RANDRIAMANANTOANINA Jacky Aimé ; Synthèse de la parole ; Mémoire d'ingénieur ; Département Electronique ; Informatique Appliquée ; 2010
- [2] T. Dutoit ; Introduction au Traitement Automatique de la Parole ; Copyright © 2000 Faculté Polytechnique de Mons
- [3] Wikipédia ; <https://fr.wikipedia.org/wiki/Robotique>; 06 Septembre 2016
- [4] Robotique ; Jean Louis Boimond ; ISTIA ; Université Angers
- [5] Réalisation d'un robot mobile : Le cerveau ; <http://roboticus.org/la-robotique/92-realisation-dun-robot-mobile-le-cerveau.html> ; 06 Septembre 2016
- [6] Cyril DROCOURT ; Localisation et modélisation de l'environnement d'un robot mobile par coopération de deux capteurs omnidirectionnels ; thèse de Doctorat ; Université De Technologie De Compiègne ; 2002
- [7] Belkhadria Khemisti ; Commande d'un robot mobile par réseaux de neurones artificiels ; Magister en Electronique ; Département d'électronique ; Université El Hadj Lakhdar Batna
- [8] E531 ; Traitement Numérique Du Signal ; Cours M2 ; Mention Électronique ; ESPA ; 2014-2015
- [9] Lindasalwa Muda, Mumtaj Begam, I. Elamvazuthi ; Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques ; Article ; 3, 2010
- [10] BENAMMAR Ryadh ; Traitement Automatique De La Parole Arabe Par Les HMMs: Calculatrice Vocale ; 2012
- [11] E554 IA ; Intelligence Artificielle ; Cours M2 ; Mention Electronique ; ESPA ; 2014-2015
- [12] Syed Ali Khayam ; The Discret Cosine Transform (DCT) : Theory and Application ; Département Electrique et Informatique ; Université d'Etat Michigan
- [13] Claude TOUZET ; Les réseaux de neurones artificiels, introduction au connexionnisme cours, exercices et travaux pratiques ; 1992

- [14] Halima ABIDET-BAHI ; NESSR: Un système neuro-expert pour la reconnaissance de la parole ; Thèse de Doctorat d'Etat ; Département d'informatique ; Université de BADJI MOHKTAR-ANNABA ALGERIE ; 2005
- [15] Nadia Benahmed ; Optimisation de réseaux de neurones pour la reconnaissance de chiffres manuscrits isolés: sélection et pondération des primitives par algorithmes génétiques, Maîtrise ; Génie de la Production Automatisée ; Université du Québec ; 2002
- [16] Les réseaux de neurone ; <http://www.alliot.fr/fneurone.html.fr>; 07 Septembre 2016
- [17] Initiation Arduino ; <http://oli.lab.perso.sfr.fr/cours%20Arduino.pdf>
- [18] Utiliser un module Bluetooth HC-05 avec Arduino ; <http://eskimon.fr/2498-arduino-annexes-g-utiliser-module-bluetooth-hc-05>; 06 Septembre 2016
- [19] Apprendre à programmer avec python 3 ; python.developpez.com/cours/apprendre-python3 ; 09 Septembre 2016
- [20] ROBERTO CHIODI ; Détection d'Activité Vocale basée sur la transformée en ondelettes; Maîtrise ; Génie Électrique ; Université Du Québec À Trois-Rivières ; 2010
- [21] BENCHENIEF Abderezek ; Reconnaissance Vocale Basée Sur Les SVM ; mémoire d'ingénieur ; Département Génie Electrique; Option Signaux et Communications., Université Mohamed Khider – Biskra ; 2011
- [22] Mabuchi Motor ; datasheet Mabuchi Motors Complete ; 2007
- [23] Bluetooth – troyesGEII ; <http://www.iut-troyes.univ-reims.fr/wikigeii/index.php/Bluetooth>; 06 Septembre 2016

Auteur : RAFIDIMALALA Tojoniaina Daniella
Titre : Reconnaissance Automatique de la Parole: Utilisation à la commande d'un robot mobile
Nombre de page : 54
Nombre de figure : 40

RESUME:

Ce présent travail porte sur la reconnaissance automatique de la parole : utilisation à la commande d'un robot mobile. Dans cette optique, nous avons créé un algorithme pour extraire les caractéristiques de la parole par la méthode cepstrale et de les classifier par l'apprentissage automatique qui est le RNA. Cela permet donc de générer la commande vocale du robot. Ensuite, nous avons créé un robot mobile en utilisant des MCC. Enfin, nous avons conçu une interface graphique sous Python permettant de commander le robot par ordinateur et de visualiser le signal de la parole et son spectre. L'envoi des commandes est fait via la liaison Bluetooth. Le but de ce projet est de fournir une Interface Homme Machine (IHM) permettant à l'utilisateur d'établir une communication vocale avec la machine pour simplifier sa vie que ce soit dans un milieu hostile ou dans un milieu domestique.

Mots clés: robot mobile, méthode cepstrale, RNA, commande vocale,

Rapporteur: Mr RATSIMBA Mamy Nirina

Contact des auteurs: tojodaniella@gmail.com (+261 34 79 451 29)