

N° d'ordre :  
N° de série :

**RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE**  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**



**UNIVERSITE ECHAHID HAMMA LAKHDAR - EL OUED**  
**FACULTÉ DES SCIENCES EXACTES**  
**Département D'Informatique**



**Mémoire de Fin D'étude**  
**Présenté pour l'obtention du Diplôme de**

## **MASTER ACADEMIQUE**

Domaine : **Mathématique et Informatique**  
Filière : **Informatique**  
Spécialité :  **Systèmes Distribués et Intelligence Artificielle**

Présenté par :

- **GUEDIRI Oussama**
- **GHEZAL Mohammed**

### **Thème**

**Proposition d'une méthode structurelle à base de  
zoom-vue pour la description de la parole arabe.**

Soutenue le 29 – 05 - 2017 Devant le jury:

M.	KHEBBACHE Mohib eddine	MAB	Président
Mme.	GUIA Sana Sahar	MAA	Rapporteur
M.	ZAIZ Faouzi	MAA	Encadreur

**Année Universitaire: 2016-2017**

Proposition d'une méthode structurelle à base de zoom-vue pour  
la description de la parole arabe.

GUEDIRI Oussama  
GHEZAL Mohammed

29/05/2017

## Dédicace

*Avant tous, je remercie dieu le tout puissant de m'avoir donné le courage et la patience pour réaliser ce travail malgré toutes les difficultés rencontrées.*

*Je dédie ce modeste travail À celui qui m'a orienté et m'a pris les secrets de la vie : « **mon Père** »,*

*À celle qui m'a ouvert les portails et m'a donné la tendresse et le courage.*

*À celle qui endeuillée pour me rendre heureuse.*

*À celle qui attend chaleureusement ce jour : "**ma chère Mère**".*

*À Mes frères et Mes sœurs.*

*À tout mes oncles, tantes, leurs conjoints ainsi que leurs enfants.*

*À mes meilleurs amis.*

*À mon binôme .*

*À tous les étudiants de la 2ème année Master promotion 2017....*

*À tous les habitants d' EL OUED.*

## Remerciements

*A Dieu, le tout puissant, nous rendons grâce pour nous avoir donné santé, patience, volonté et surtout raison.*

*En premier lieu, je tiens à remercier "Mr. ZAIZ Faouzi" notre promoteur pour sa serviabilité, sa disponibilité, ses remarques et ses orientations constructives qui nous ont été utiles tout au long de notre projet..*

*Je remercie également les membres qui nous ont fait l'honneur de participer, au jury de ce mémoire.*

*Enfin, je remercie tous ceux qui ont, de près ou de loin, contribué à la concrétisation de ce travail trouvent ici ma gratitude et reconnaissance.*

# Résumé

Un signal de la parole enregistré sur machine n'est pas reconnu que à partir d'une séquence de valeurs numériques. Alors ,pas d'information sur l'ordre des mots ou des caractères.

De ce fait, malgré toutes les méthodes proposées ,il est difficile de segmenter un signal de parole en segments de base.

Dans ce travail,nous proposons une méthode qui permet de segmenter un signal de parole arabe en syllabes,en se basant sur un modèle de zoom vue du signal traité .

La méthode proposé est testée et comparée avec d'autres méthodes . Elle permet d'avoir un bon taux de segmentation dans un temps raisonable.

*Mots Clés:* Un signal de la parole, segments de base, syllabes,parole arabe, modèle de zoom vue, taux de segmentation.

# Abstract

A computer can't recognize a recorded signal speech only if it had been transferred into number values ,so it can't be understood that informations in a type of words or characters .

Because of this reason ,although all the proposed tools and methods, the speech segmentation into basic parts stays a very difficult thing

In this work , we propose a segmentation method of the arabic speech to syllables that based on the zoom view of the speech signal.

Our method had been tested and compared with other methods, that shown here efficiency in the segmentation with an accepted treatment time.

*Keywords:* signal speech, syllables, arabic speech, efficient in the segmentation , zoom view

# الملخص

الحاسوب لا يمكنه التعرف على إشارة الكلام المسجلة إلا عبر تحويلها إلى قيم عددية بإذن هو لا يستطيع فهم معلومات على شكل كلمات أو حروف .

ولهذا السبب ، وبالرغم من كل الوسائل والطرق المقترحة يبقى تقسيم الكلام إلى أجزاء و قطع أساسية أمر في غاية الصعوبة .

في هذا العمل ، اقترحنا طريقة تسمح بتقسيم إشارة كلام عربي إلى مقاطع صوتية ( Syllables ) ، وذلك اعتمادا على العرض المكبر ( Zoom vue ) لإشارة الكلام المعالج.

الطريقة التي إقترحناها مجربة وتمت مقارنتها بعدة طرق أخرى ، حيث أثبتت هذه الطريقة نجاعة كبيرة في التقسيم وذلك في زمن معالجة مقبول .

**كلمات مفتاحية:** إشارة الكلام ، Syllables مقاطع صوتية بكلام عربي، نجاعة كبيرة في التقسيم، Zoom vue العرض المكبر .

# Table des matières

Dédicace	ii
Résumé	iv
Table des matières	vii
Liste des algorithmes	xii
Acronymes	xiii
Introduction générale	1
<b>1 La reconnaissance automatique de paroles</b>	<b>2</b>
1.1 Introduction	2
Introduction	2
1.2 Historique	2
1.2.1 Le Départ	2
1.2.2 Les premiers résultats	2
1.2.3 Le début du progrès et de la réussite	2
1.2.4 Le changement de l'orientation vers les langues connues	2
1.3 Le son naturel	3
1.3.1 Définition	3
1.3.2 Les éléments nécessaires	3
1.3.3 Fonctionnement de l'oreille	3
1.3.4 Perception du son	4
1.3.5 Les types de son	4
1.4 Outils d'acquisition	5
1.4.1 Microphones	5
1.4.2 Cartes son	6
1.5 La langue arabe	8
1.5.1 Introduction à la langue arabe	8
1.5.2 Les spécificités de la langue arabe	8
1.6 système de reconnaissance automatique de paroles SRAP	10
1.6.1 Définition	10
1.6.2 Les avantages de la reconnaissance automatique de la parole	10
1.6.3 Applications de la reconnaissance de la parole	10
1.6.4 Approche de la reconnaissance	11
1.6.5 Comment évaluer notre système	14
1.7 Conclusion	14
<b>2 Traitement des paroles</b>	<b>15</b>
2.1 Introduction	15
Introduction	15
2.2 Caractéristiques du signal de parole	15
2.2.1 La fréquence fondamentale	15
2.2.2 L'intensité	16
2.2.3 Le Pitch	16

2.2.4	Le timbre . . . . .	17
2.3	Les opérations de traitement . . . . .	18
2.3.1	La numérisation . . . . .	18
2.3.2	La segmentation . . . . .	20
2.3.3	La paramétrisation et l'Extraction de caractéristiques . . . . .	21
2.3.4	La classification . . . . .	27
2.4	Conclusion . . . . .	31
<b>3</b>	<b>Conception et mise en œuvre</b>	<b>32</b>
3.1	Introduction . . . . .	32
3.2	Mise en œuvre du système . . . . .	32
3.2.1	Acquisition : . . . . .	34
3.2.2	Méthode de segmentation proposée : . . . . .	35
3.2.3	Extraction des caractéristiques : . . . . .	42
3.2.4	Classification : . . . . .	44
3.3	Conclusion . . . . .	44
<b>4</b>	<b>Résultats et discussion</b>	<b>45</b>
4.1	Introduction . . . . .	45
4.2	Choix du Language de programmation . . . . .	45
4.2.1	Pourquoi JAVA ? . . . . .	45
4.3	Interface du système . . . . .	46
4.4	Fonctionnalité du système . . . . .	46
4.5	Test et bilan . . . . .	47
4.6	Comparaison des résultats . . . . .	50
4.6.1	Première différence : . . . . .	50
4.6.2	Seconde différence : . . . . .	51
4.6.3	Troisième différence : . . . . .	52
4.7	Conclusion . . . . .	52
	<b>Conclusion générale</b>	<b>53</b>

# Liste des figures

1.1	Fonctionnement de l'oreille . . . . .	4
1.2	Perception et analyse du son par l'être humain [6]. . . . .	4
1.3	Le signal de son analogique vs signal de son numérique [15]. . . . .	5
1.4	Processus d'acquisition d'un signal. [4]. . . . .	5
1.5	Fonctionnement d'un microphone dynamique. . . . .	5
1.6	Rôle d'une carte son. . . . .	7
1.7	Les sorties de prononciation. . . . .	8
1.8	Classification des consonnes tenant compte des contraintes de la transcription [19]. . . . .	8
1.9	Système de reconnaissance automatique de parole de mots isolés [12]. . . . .	13
1.10	Schéma de principe d'un système de reconnaissance de la parole continue [2]. . . . .	13
2.1	La fréquence fondamentale de son. . . . .	16
2.2	Intensité de son. . . . .	16
2.3	Le Pitch de Son. . . . .	17
2.4	Le timbre de son. . . . .	17
2.5	Enregistrement numérique d'un signal .[5] . . . . .	18
2.6	La convertisseur analogique numérique . . . . .	18
2.7	un signal échantillonné.[6] . . . . .	19
2.8	un signal quantifié.[6] . . . . .	19
2.9	Exemples de segmentation de parole continue.[4] . . . . .	20
2.10	Étapes de calcul d'un vecteur caractéristique de type MFCC [6]. . . . .	22
2.11	Les fonctions de fenêtrage [6]. . . . .	23
2.12	L'extraction des paramètres vocaux par LPC. [13]. . . . .	25
2.13	Analyse PLP [12]. . . . .	25
2.14	Comparaison des spectres dérivés d'une analyse PLP et LPC [12]. . . . .	26
2.15	Les phases de la classification. [10]. . . . .	27
2.16	Séquence cachée et observée de la HMM [4]. . . . .	29
2.17	Classification neuronale. . . . .	30
2.18	Les couches du classificateur FLC. . . . .	31
3.1	Architecture générale de notre système . . . . .	33
3.2	Segmentation en mots VS segmentation en syllabes . . . . .	35
3.3	Algorithme général de la segmentation . . . . .	36
3.4	Exemple de valeur maximale . . . . .	37
3.5	Exemple de période de phase de la valeur maximale. . . . .	37
3.6	Exemple de période de phase de la forme répétée . . . . .	39
3.7	Valeurs maximales dans les zones gauche et droite . . . . .	40
3.8	Exemple de forme répétée . . . . .	42
3.9	la schéma de l'extraction des caractéristiques . . . . .	42
3.10	La normalisation . . . . .	43
3.11	Classification par FLC . . . . .	44
4.1	L'interface utilisateur globale du système . . . . .	46
4.2	L'enregistrement de parole . . . . .	46
4.3	Segmentation de parole . . . . .	47
4.4	Exemple de segmentation (1) . . . . .	48

4.5	Exemple de segmentation (2) . . . . .	49
4.6	la comparaison de segmentation (2) . . . . .	50
4.7	Comparaison de segmentation (1) . . . . .	51

# Liste des tableaux

1.1	Tableau résumé des caractéristiques des microphones [7]	6
1.2	Exemples d'applications de la reconnaissance de la parole	11
1.3	La comparaison entre l'approche globale et l'approche analytique	12
4.1	Résultats de segmentation.	50
4.2	Comparison du temps de calcul .	52

# Liste des algorithmes

2.1	Algorithm	29
3.1	Algorithm de Capture.	34
3.2	Algorithme de la valeur maximale.	37
3.3	Algorithme de frontière droite de la phase.	38
3.4	Algorithme de frontière gauche de la phase.	38
3.5	Algorithme de la période de la phase.	38
3.6	Algorithme de la période de phase de la forme répétée.	39
3.7	Condition d'arrêt.	40
3.8	la classe syllabe.	41
3.9	Post-traitement.	41
3.10	Algorithme de la normalisation.	43

# Acronymes

RAP	Reconnaissance Automatique de Parole
MIC	Modulation d'Impulsion Codée
DSP	Digital Signal Processor
DAC	Digital to Analog Converter
MIDI	Musical Instrument Digital Interface
MFCC	Mel-frequency cepstral coefficients
FFT	Fast Fourier transform
DFT	Discrete Fourier transform
LPC	linear Predictive Coding
PLP	Perceptually Based Linear Prediction
K-PPV	K Plus Proches Voisins
HMM	Hidden Markov Model
SRAP	Système de Reconnaissance Automatique de Parole
FLC	Fast Logic Classifier
SP	Speech Processing

# Introduction générale

La reconnaissance vocale ou la reconnaissance automatique de parole (**RAP**) est une technologie informatique permettant à un logiciel d'interpréter la parole humaine pour faciliter l'interaction homme-machine. Cette technologie utilise des méthodes informatiques pour l'accès à de nombreuses applications comme : Saisie des données grâce à une interface vocale, aide aux handicapés, les applications téléphoniques, utilisation de la RAP dans les jeux électroniques, etc. Vu l'importance de la RAP, plusieurs systèmes ont été développés pour la reconnaissance vocale, parmi les plus connus : IBM Via voice, Microsoft SAPI et d'autres. Aussi, il y a des Open Sources comme HTK (Hidden Markov Model Toolkit) , ISIP , AVCSR et CMU Sphinx .

La parole est représentée dans la machine par une séquence de valeurs numériques. Cette séquence ne contient pas d'informations qui indique le début et la fin des mots ou caractères. Pour cela, malgré toutes les tentatives de réaliser une méthode de segmentation efficace, les résultats restent loin de répondre à nos ambitions, ce qui laisse cette phase comme un très grand défi.

Dans cette étude nous proposons une contribution qui se présente par une méthode de segmentation du signal de la parole en syllabe en se basant sur un modèle de zoom-vue. La méthode est testée et comparée avec d'autres méthodes déjà réalisées, et elle permet d'avoir un très bon taux de segmentation en syllabes.

Le reste du travail est organisé comme suit :

Le premier chapitre présente un état de l'art sur les systèmes de reconnaissance de la parole, dans lequel nous allons donner un aperçu sur l'architecture d'un système de reconnaissance de la parole.

Le deuxième chapitre présente un panorama sur la segmentation du signal de la parole, en présentant les différentes approches et stratégies de segmentation.

Le troisième chapitre est consacré à la conception et la mise en œuvre du système et décrit en détail la méthode de segmentation proposée.

Le chapitre quatrième présente les résultats obtenus par notre méthode ainsi qu'une comparaison, discussion et évaluation de la méthode.

Enfin, nous terminons le travail par une conclusion sur les résultats obtenus par la méthode proposée, et des perspectives de ce travail.

# La reconnaissance automatique de paroles

## 1.1 Introduction

Le système de reconnaissance automatique de parole (**SRAP**) est un système qui permet à une machine d'extraire le message oral contenu dans un signal de parole.

Nous nous sommes intéressés dans ce chapitre à un système RAP pour traiter les paroles arabes . Dans un premier temps, nous allons essayer de donner un historique et une idée générale sur la reconnaissance automatique de parole , de montrer les avantages de cette dernière , ainsi que les méthodes utilisées pour créer un système **SRAP** efficace.

## 1.2 Historique

Cette partie reflète brièvement l'évolution de la reconnaissance automatique de parole dès le début à nos jours .

### 1.2.1 Le Départ

Les premiers essais ont consisté en la mise en place des machines capables de comprendre des mots humains à la fin de l'année 40 .C'était au ministère de la Défense aux Etats-Unis afin d'interpréter les lettres russes qui ont intercepté.

En dépit de tous les grands efforts et le coût élevé , les résultats obtenus ne sont pas ceux attendus.

### 1.2.2 Les premiers résultats

Dans les années soixante , Les chercheurs ont porté leur attention sur la reconnaissance des mots isolés, Et ont essayé d'utiliser cette reconnaissance dans les mini-applications telles que :

- la commande vocale
- la dictée vocale

Ces applications ont été à l'origine de cette technique. [2]

### 1.2.3 Le début du progrès et de la réussite

Nous voyons que la première réalisation commerciale de l'entreprise avec la reconnaissance vocale est développée par **J.J.W. Glenn** et **M.H. Hitchcok** sous le nom d'«**le Voice Command system**» :Dispositif fiable qui reconnaît 24 mots différents Après cinq séances d'apprentissage sur le même parlant. Ce fut dans les années soixante-dix ans. [2]

### 1.2.4 Le changement de l'orientation vers les langues connues

À l'heure actuelle,la plupart des chercheurs tentent de guider leurs applications vers la compréhension des langues parlées et des phrases complexes. Pour cette tendance, Il faut trouver des solutions pour les problèmes linguistiques face à l'application.

Enfin , nous voyons que l'accès à une application puissante de la reconnaissance semblable à celle de l'être humain est devenu possible, avec le développement rapide de cette technologie.

## 1.3 Le son naturel

### 1.3.1 Définition

Dans la vie courante, nous percevons différentes voix, que ce soit sans le vouloir tels que :

- la voix du vent.
- Le bruit des avions et divers moyens de transport.
- le chant des oiseaux.
- le bruit de l'eau ruisselant

ou des voix que nous entendons délibérément et intentionnellement comme :

- la voix des personnes qui nous entourent.
- la musique et la télévision.

Toutes ces voix différentes sont considérées dans le concept de **son naturel**.

Alors, quel est ce concept ?

Il s'agit des vibrations qui se propagent dans toutes les directions ,dans une plage de vitesses différentes, selon l'environnement et ses obstacles.

Mais comment résultent ces vibrations ?

Elles sont dues à plusieurs raisons différentes, mais nous allons expliquer la production de la voix humaine.[13]

### 1.3.2 Les éléments nécessaires

- **une source** : qui produit le son

Le son vient à la suite de vibrations résultant d'un corps solide, liquide ou gazeux... etc. Ce son varie en fonction de son origine. Le son vient de plusieurs natures différentes tels que : Frottement , friction, pression,... etc

- **un milieu** : qui transmet la vibration

Les vibrations produites par la source sonore se propagent dans l'air et dans tout environnement Irrégulièrement sans le mouvement du corps qui les produit. Seule l'énergie se déplace d'un endroit à un autre endroit.

Dans certains cas, Il y a des obstacles au mouvement des vibrations sonores qui conduisent ainsi à changer ce son.

- **un récepteur** : l'oreille et l'ouïe

L'oreille humaine est un système complexe qui reçoit des voix différentes et de les analyse . Elle est capable de détecter les sons et les voix venantes de longues distances, selon les différents degrés de fréquences.[14]

### 1.3.3 Fonctionnement de l'oreille

1. Les ondes sonores atteignent l'oreille, ces ondes fonctionnent par des vibrations au niveau de **la tympan** .
2. Ces vibrations transférées à la **fenêtre ovale** par **osselets** .
3. après ça,les ondes transmises aux **cellules ciliées** situés dans **l'endolymphe** et **la membrane basillaire**.

Ces cellules sont divisées en deux parties : cellules internes et externes , Et représentent un centre de sensibilité aux différentes fréquences .

- Les cellules qui possèdent l'unité de réception des sons les plus aigus sont les cellules les plus proches de la base de **la cochlée**.
- Les cellules qui possèdent l'unité de réception des sons les plus graves sont les cellules les plus proches de l'extrémité de **la cochlée**.

4. Le son reçu par l'oreille humaine est le son qui sa fréquence comprise entre 20 Hz et 20 kHz.[8]

La figure ci-des sous illustre le fonctionnement de l'oreille.

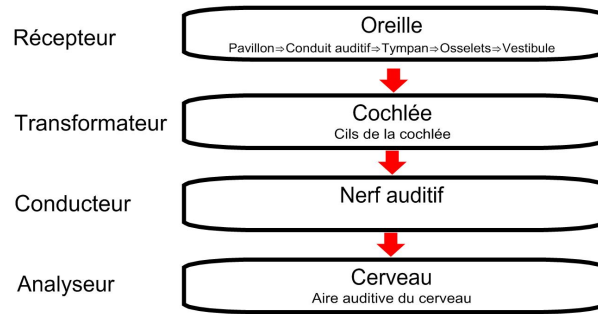


Figure 1.1: Fonctionnement de l'oreille .

### 1.3.4 Perception du son

1. L'oreille humaine reçoit des vibrations sonores.
2. Ces vibrations transformées en messages électriques au niveau des cellules nerveuses
3. Le cerveau fonctionne à interpréter et à comprendre ces messages.[6] voir figure suivante

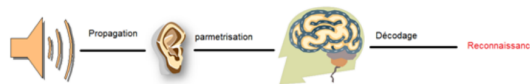


Figure 1.2: Perception et analyse du son par l'être humain [6].

#### Donc :

Le son est un phénomène psychologique qui est essentiellement la perception. Les vibrations acoustiques sont transférées avant d'arriver au centre sensoriel dans le cerveau . Cela signifie qu'elles ne parviennent pas de façon brute au cerveau.

#### Discussion :

Nous concluons que le son réside à l'intérieur du cerveau humain, et non à l'extérieur, parce que le processus de compréhension du son est à l'intérieur du cerveau.

### 1.3.5 Les types de son

Nous allons maintenant parler des types de sons : le son analogique et le son numérique.

#### 1.3.5.1 Le son analogique

Le son analogique est un signal électrique continu de sorte qu'il existe une corrélation positive entre la variation de vibrations acoustiques et la valeur de tension électrique, la transmission de courant électrique à travers le fil et la modification des vibrations électriques dans le son sont identiques. Cette symétrie est particulièrement dans la fréquence. Donc, ce genre de son n'est pas manipulable par un ordinateur qui ne connaît que 0 et 1 . [6]

#### 1.3.5.2 Le son numérique

Ce type est représenté par un ensemble de zéro et un .Ceci est le type utilisé dans le CD audio . Le passage du premier type vers le deuxième type est appelé **échantillonnage**. [6] voir figure la figure

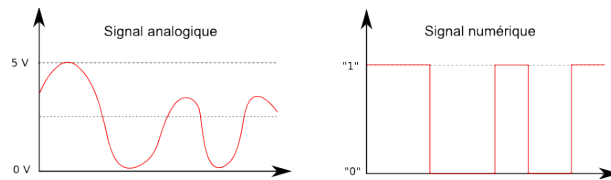


Figure 1.3: Le signal de son analogique vs signal de son numérique [15].

## 1.4 Outils d'acquisition

Le système d'acquisition d'informations représente l'intermédiaire entre le capteur et l'ordinateur. Il est divisé en deux parties : partie matérielle et partie logicielle. Ce système recueille diverses informations du capteur, puis les met dans les différents moyens de stockage par exemple (disque dur...), afin de les gérer plus tard. figure 1.4.



Figure 1.4: Processus d'acquisition d'un signal. [4].

### 1.4.1 Microphones

Le microphone est un appareil qui capture les modifications des vibrations sonores dans l'air et les transforme en une tension électrique.

Donc, le travail de base du microphone est la conversion de diverses vibrations acoustiques en impulsions électriques, et leur enregistrement dans l'ordinateur. [21] voir la figure

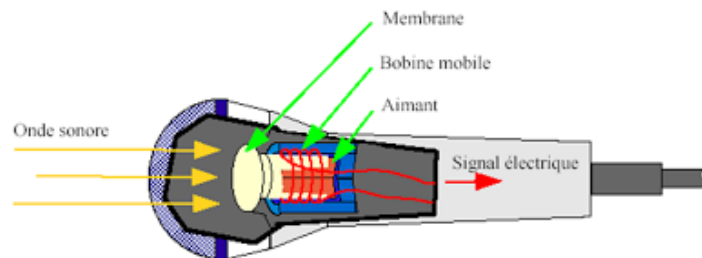


Figure 1.5: Fonctionnement d'un microphone dynamique.

#### 1.4.1.1 Type des microphones

Microphone contient plusieurs types :

- à condensateur
- à magnétostriction
- électronique
- ionique ...etc . voir table 1.1

Nous essayons de prendre l'exemple de Microphone à condensateur, Parmi ses propriétés :

- petite taille.
- construction facile.
- l'efficacité.

Tableau 1.1: Tableau résumé des caractéristiques des microphones [7]

Type	Utilisations	Avantages	Inconvénients	Qualité	Prix
Charbon	Téléphone	Forte puissance de sortie	Souffle, Bande passante réduite	Médiocre	peu coûteux
Dynamique	Toutes	Sensibilité élevée Robustesse, Directivités variés, Courbe de réponse étendue		De médiocre à excellent	peu coûteux à très coûteux
Ruban	Studio et intérieur	Très grande fidélité, Sensibilité moyenne	Usage réservé à l'intérieur très fragile	assez bon à excellent	moyen à coûteux
Statique	Toutes	Haute fidélité, nécessité de forte amplification	Nécessite une source de tension de polarisation	Excellent	très coûteux
Électret	Toutes	Comme statique, mais sans polarisation	Craint parfois la chaleur et l'humidité	excellent	coûteux à très coûteux
Piézo	Surtout public-adress	Bonne sensibilité, assez fidèle, tension de sortie élevée	Comme électret	de médiocre à assez bon	peu coûteux

### 1.4.2 Cartes son

La carte son représente un périphérique d'un ordinateur. Sa travail principal est de gérer toutes les entrées-sorties audio .Et cela se fait par l'envoi des sons émis au haut-parleur et la réception des sons reçus par l'ordinateur .

Elle doit être connecté à l'ordinateur sur bus PCI,USB...etc car elle est une extension externe.

#### 1.4.2.1 Les types de cartes son

Il y a trois types de cartes son :

- une puce intégrée à la carte mère
- une carte interne
- une carte externe

#### 1.4.2.2 Rôle d'une carte son

On peut résumer le rôle de la carte son par les étapes suivantes :

1. Le signal analogique arrive à l'**entrée MIC** de la carte via une micorphone .
2. ce signal est filtré et nettoyé par un **circuit de conditionnement**.
3. Maintenant, la carte son fait la conversion analogique numérique afin de rendre le traitement de l'information facile par l'ordinateur.
4. Cette conversion comprend les opérations suivantes : l'échantillonnage, la quantification et le codage.
5. Enfin, la carte a stockée ces informations sous la forme de valeurs numériques . [4]

LA figure suivante présente le rôle de le carte son.



Figure 1.6: Rôle d'une carte son.

#### 1.4.2.3 Les principaux éléments d'une carte

1. **Le processeur spécialisé DSP (digital signal processor) :** Il fait tous les traitements numériques du son.
2. **Le convertisseur digital-analogique DAC (digital to analog converter) :** Il fait la conversion des données sonores de l'ordinateur en signal analogique .
3. **Le convertisseur analogique-numérique appelé ADC**
4. **Les connecteurs d'entrées-sorties externes :** parmi eux :
  - Une entrée microphone (notée parfois Mic)
  - Une interface MIDI : permettant de connecter des instruments de musique .
5. **Les connecteurs d'entrées-sorties internes :** parmi eux :
  - Connecteur CD-ROM / DVD-ROM.
  - Connecteur pour répondeur téléphonique. [18]

## 1.5 La langue arabe

### 1.5.1 Introduction à la langue arabe

La langue arabe est la langue classée sixième dans le monde en terme de locuteurs, dont le nombre atteint environ 250 millions .

Le nombre de programmes qui sont intéressés à cette langue est en une augmentation continue. Mais le problème que se pose toujours est la difficulté de traitement automatique de cette langue en raison de ses **spécificités morphologiques et syntaxiques** .

L'alphabet arabe contient 28 caractères, dont 6 voyelles et les restes sont des consonnes. On peut diviser les consonnes selon plusieurs critères :

1. Les sorties de prononciation comme maitre la figure ci-dessous.



Figure 1.7: Les sorties de prononciation. .

2. Les caractères solaires et lunaires ,selon le « ال » au début du mot :  
les 28 caractères arabes ont été divisées en deux groupes :
  - 14 caractères solaires .
  - 14 caractères lunaires .La figure montre cette classification.

Solaires	Lunaires
ت د ذ ر ز س ش ض ط ظ ل ن	أ ب ج ح خ ع غ ف ق ك هـ م و ي

Figure 1.8: Classification des consonnes tenant compte des contraintes de la transcription [19].

3. La vibration lors de la prononciation de la caractères...etc

### 1.5.2 Les spécificités de la langue arabe

Parmi les caractéristiques de cette langue ce qui rend le traitement difficile :

- Le mot arabe peut être égal au sens d'une phrase complète en français et en anglais.  
« أ رأيت » ( ara ayta),  
l'équivalent en français de  
« Est-ce que vous avez vu ? ».

- L'ordre des mots dans la phrase arabe n'a aucune importance, à l'inverse de la langue française, dont l'ordre est toujours : (sujet, verbe, complément) .  
Ceci conduit à une grande difficulté à traiter avec la langue arabe.
- Certains caractères arabes sont liés dans le processus de reconnaissance de la parole avec la qualité du microphone , parce que ces façons de prononciation sont difficiles, comme :  
خ (kh : prononciation Kha , ), ض (d : prononciation Dad), ذ (d : prononciation Thal), ظ (z : prononciation Zah),
- Certaines lettres n'ont pas d'équivalent dans d'autres langues.  
comme par exemple :  
غ (gh : prononciation Ghayn) .
- Certains lettres arabes ont une façon de prononciation identique, et de ce fait nous ne pouvons pas les différencier.  
comme par exemple :  
ظ et ض .[22]

## 1.6 système de reconnaissance automatique de paroles SRAP

### 1.6.1 Définition

Dans l'informatique Il existe deux domaines du traitement automatique de la parole. Le premier est la reconnaissance automatique de la parole , Le second est synthèse vocale.

La reconnaissance automatique donne la possibilité à la machine de comprendre et de traiter les informations fournies oralement.

à la différence de le synthèse vocale qui consiste à transformer l'entrée du texte par l'utilisateur humain en parties sonores prononcés.

L'étude de ces sujets nécessite le passage sur plusieurs sciences ,parmi elles :

- le traitement du signal
- l'informatique
- la linguistique
- les statistiques...etc

Maintenant, nous voulons faire la différence entre ces deux sujets : dans la synthèse vocale , le système peut bien fonctionner même sans l'utilisation de la reconnaissance automatique. Contrairement à d'autres systemes.

Le traitement automatique de la parole à l'heure actuelle ouvre de grandes perspectives, Surtout dans le domaine de l'interaction entre l'utilisateur et la machine. Lorsque cette interaction a commencé à prendre l'indépendance progressivement en évitant divers périphériques connus comme : claviers, souris...etc,ce traitement automatique facilite le dialogue personne/machine de manière significative.

Dans les logiciels de reconnaissance vocale,En dépit de la grande vitesse accordée par l'entrée vocale par à rapport l'entrée par le clavier. Le clavier reste dans ces logiciels une chose essentiel pour corriger les erreurs qui se trouvent dans l'entrée vocale. [2]

### 1.6.2 Les avantages de la reconnaissance automatique de la parole

Les avantages qui résultent de l'utilisation de la reconnaissance automatique de la parole sont illimités, parmi eux :

- L'indépendance totale de l'utilisation de la main et du clavier.
- Enorme vitesse par rapport à l'usage du clavier et d'autres moyens d'entrée .
- Facilité de la parole par contre les erreurs d'écriture.

Ces avantages donnent de nombreuses solutions Afin d'éliminer plusieurs limitations existantes dans les anciens programmes :

- Saisie vocale de données ;
- la conduite vocale des moyens de transport .
- La commande vocale de robots ...etc.

### 1.6.3 Applications de la reconnaissance de la parole

Les applications de la reconnaissance automatique de parole sont nombreuses. Elles ont été construites pour remplacer ou compléter la transaction avec la machine dans les applications existantes, Lorsque l'utilisateur devient indépendant totalement de divers périphériques informatiques. par exemple, pour accéder à un service , La parole représente parfois le seul mode de communication Disponible à utiliser .

Le tableau 1.2 présente quelques exemples d'applications de la reconnaissance automatique de parole.[12]

Tableau 1.2: Exemples d'applications de la reconnaissance de la parole

Domaine	Applications
Téléphonie	Automatisation de transactions téléphoniques (ex : opérations bancaires), selfservice téléphonique pour l'accès à des services d'information (ex : consultation des bulletins météorologiques), etc.
Automobile	Contrôle mains-libres des équipements tels que la radio, le conditionnement dans le système de navigation, le téléphone sans fil (ex : voice dialing), les systèmes télématiques, etc.
Multimédia	Logiciels de dictée vocale, interaction vocale dans les logiciels pédagogiques (ex : apprentissage des langues) et ludiques (ex : jeux vidéo), etc.
Médical	Aide aux personnes handicapées.
Industriel	Contrôle vocal de machines, application pour la gestion de stocks, etc.

#### 1.6.4 Approche de la reconnaissance

Dans la reconnaissance automatique de la parole, il existe deux approches différentes :

- l'approche globale
- l'approche analytique

Ces deux approches différentes de l'unité de base utilisé ( mots - phonèmes - syllabes...) constituante du signal de parole.

##### 1.6.4.1 Approche globale

Dans l'approche globale Le mot est considéré comme l'unité de base, et ne peut être divisé. L'idée de cette approche est de donner des images décrivent tous les mots utilisés dans le programme. Cette opération se fait dans la phase d'apprentissage. Lorsque l'utilisateur répète la prononciation du même mot plusieurs fois pour faire l'apprendre à la machine.

##### L'avantage :

Elle est utile contre le changement possible qui se produit au milieu de la prononciation du mot, ou les changements limités.

##### L'inconvénient :

Elle est limitée aux petits vocabulaires prononcés par un petit nombre de locuteurs .[17]

##### 1.6.4.2 Approche analytique

Cette approche est plus générale que la première , car elle peut fonctionner avec de grands vocabulaires . Cette approche fait que le programme n'apprend pas des mots complets, mais il apprend des parties de mots représentés par (phonèmes, syllabes, ...).

Donc, l'unité de base de cette approche est un partie de mot (phonèmes, syllabes, ...).

##### L'avantage :

Dans l'approche analytique. les mots sont traités en tant que suite de phonèmes. Contrairement à l'approche globale , où la machine a besoin d'un grand volume de mémoire, afin d'enregistrer tous les mots utilisés.

##### L'inconvénient :

le grand changement du phonème en fonction du contexte (effets de la coarticulation) .[17]

Tableau 1.3: La comparaison entre l'approche globale et l'approche analytique

Caractéristique \ Approche	Approche Globale	Approche Analytique
Unité de base	mot	phonème - syllabes ...
L'effet de coarticulation	petit	grand
Vocabulaire traité	petit	grand
Taille de mémoire nécessaire	grande	petite

### 1.6.4.3 Principe général de ces deux méthodes

Ce qui distingue ces méthodes est l'unité de base pour la reconnaissance :

- Approche globale : mot
- Approche analytique : partie de mot (phonèmes, syllabes, ...)

Le même principe pour ces deux approches, s'appuient sur deux phases :

#### — La phase d'apprentissage

Un locuteur prononce l'ensemble des termes (les unités de base), et fait répéter plusieurs fois, pour créer une base de données des signaux acoustiques (dictionnaire) dans la machine.

Le locuteur doit prononcer l'ensemble du vocabulaire dans plusieurs situations différentes (Voix forte - voix faible,,), Ceci afin d'obtenir de meilleurs résultats.

#### — La phase de reconnaissance

Un locuteur prononce un mot ou une phrase du vocabulaire. Ensuite, la reconnaissance compare ce mot avec tous les mots dans le dictionnaire et sélectionne le mot le plus proche .

les trois parties principales de cette phase :

- Un capteur physique (par exemple : **un microphone**) .
- Une partie de paramétrisation des mots (unité de base) (par exemple un analyseur **LPC** ou **MFCC**) .
- Une partie de décision pour lier un mot inconnu avec le mot le plus proche dans le dictionnaire (par exemple :**reseau de neurones**) . [17]

#### — Les types de reconnaissance :

On peut diviser les types de reconnaissance selon le mode d'élocution :

##### 1. Reconnaissance de mots isolés

Le locuteur prononce les mots séparément, lorsque il prend un petit silence entre deux mots. Dans les systèmes de reconnaissance de parole de ce type , un signal de parole est analysé afin d'extraire son sens sous la forme d'une séquence de mots.

Cette méthode est utilisée principalement dans l'approche globale voir la figure 1.9.

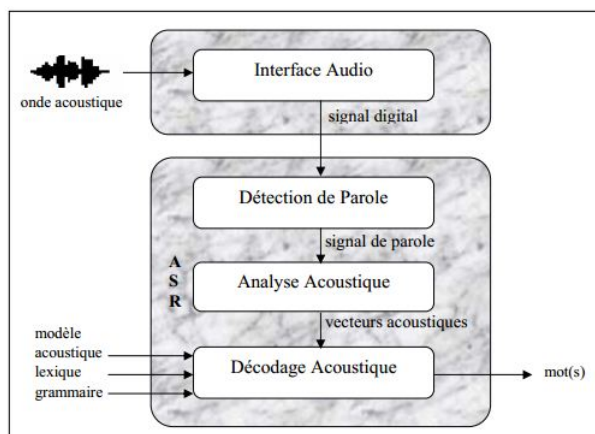


Figure 1.9: Système de reconnaissance automatique de parole de mots isolés [12].

## 2. Reconnaissance de la parole continue

Le locuteur dit un discours de façon normale, sans l'intercaler de silence. Le programme analyse et sépare ce discours afin de le traiter.

Cette méthode est utilisée principalement dans l'approche analytique .

Les systèmes de reconnaissance de parole de cet type reposent sur une succession de modules :

### le module acoustique :

il est utilisé pour extraire les caractéristiques physiques du signal capturé.

### le module phonétique :

qui reconnaît les parties de mot (phonèmes-syllabe ...) prononcés .

### le module lexical et le module phonologique :

ils sont utilisés pour reconnaître les mots a partir des (phonèmes-syllabes ...) résultants .

### Les modules syntaxiques et sémantiques :

permettent de reconnaître le sens de la phrase.[17] et[2]

Voir la figure suivante :

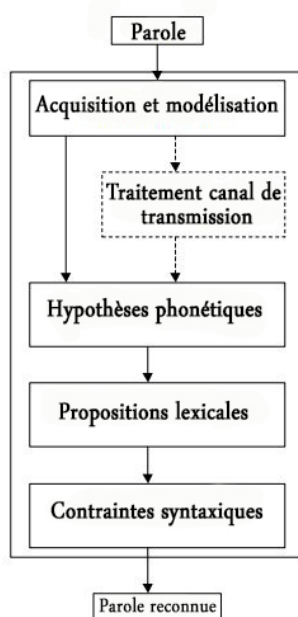


Figure 1.10: Schéma de principe d'un système de reconnaissance de la parole continue [2].

### 1.6.5 Comment évaluer notre système

Pour évaluer notre système, Nous devons répondre aux questions suivantes :

#### **un utilisateur unique ou plusieurs utilisateurs ?**

les systèmes dépendants d'un seul locuteur sont plus faciles à développer que les systèmes indépendants .

#### **des mots isolés ou paroles continues ?**

La complexité varie d'un cas à un autre. Il est plus simple de reconnaître des mots isolés qu'une parole continue.

#### **L'efficacité du système ?**

Est-ce-que le système dans des conditions difficiles fonctionne correctement ? par exemple l'influence :

- Du bruit lors de l'enregistrement de la parole
- La Qualité du microphone .
- La Prononciation non claire ...etc

#### **Discussion :**

En général, Un système qui fonctionne avec plusieurs utilisateurs qui supporte la parole continue et qui peut travailler dans des conditions difficiles , représente le système le plus performant et le plus désiré.

## 1.7 Conclusion

Nous concluons à la fin de ce chapitre que les applications de reconnaissance vocale sont conçues spécifiquement pour aider ou remplacer les moyens de communication existants (homme-machine) , Et dans certains cas, ils deviennent le seul moyen de communication.

Comme nous savons aussi qu'on peut classer les systèmes de reconnaissance automatique selon plusieurs critères (l'objectif demandé...)et que ces applications sont très sensibles aux influences extérieures.

# Traitement des paroles

## 2.1 Introduction

En informatique ,La science moderne s'est intéressée à améliorer la communication entre l'homme et la machine. Elle tend à remplacer la communication ancienne (Clavier...) par une communication plus directe (Une parole directe de l'utilisateur). Cela permet une communication analogique et continue contrairement à la méthode ancienne.

Pour cette raison, Le traitement de la parole **SP** (Speech processing) est devenu comme une partie du domaine informatique pour améliorer la communication homme- machine.

On va voir dans ce chapitre Les étapes nécessaires dans le traitement de la parole et les moyens les plus utilisés dans chaque étape.

## 2.2 Caractéristiques du signal de parole

Ce sont les caractéristiques qui nous permettent de distinguer le son, et nous avons besoin d'eux dans le processus du traitement de la parole plus tard[13] [6] :

- La fréquence fondamentale
- L'Intensité
- Le timbre
- Le pitch

Il existe un lien significatif parmi ces caractéristiques, de telle sorte que chacune affecte l'autre.

### 2.2.1 La fréquence fondamentale

Appelée aussi **la hauteur** du son , qui nous permet de l'évaluer , elle est une caractéristique distinctive du signal périodique, qui représente la plus petite fréquence.[13] figure2.1

- La **fréquence** d'un signal périodique est l'inverse de sa période.
- L'unité de mesure de la fréquence est le **:hertz Hz**.
- Un **hertz** est équivalent à un événement par seconde (s-1 ou 1/s).
- '**Les harmoniques** sont de nombres multiples de la fréquence fondamentale.
- tous les signaux périodiques ont une fréquence fondamentale.

#### Exemple :

La période = 20 s

La fréquence fondamentale = 1/20 Hz

Les harmonique =2/20 Hz,3/20 Hz,...

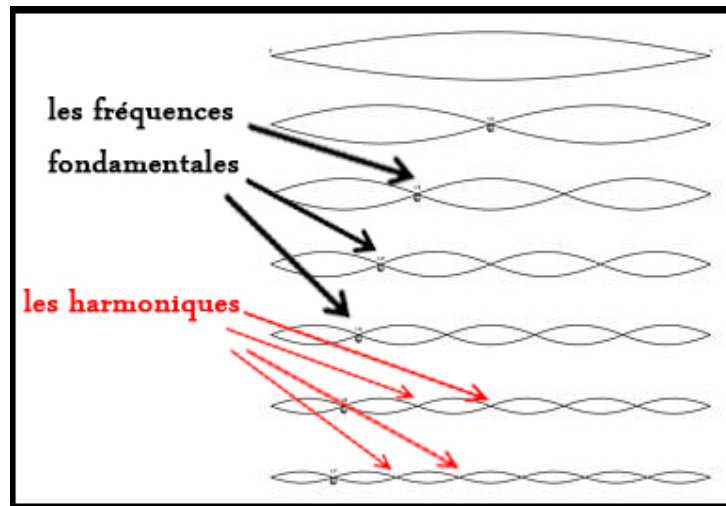


Figure 2.1: La fréquence fondamentale de son.

### 2.2.2 L'intensité

Le **volume** du son, qui permet la comparaison entre la puissance et la faiblesse du son. Il est lié directement avec l'amplitude de l'onde, et a une incidence sur la puissance sonore, de sorte que, le son le plus fort a l'intensité la plus grande et l'amplitude de l'onde la plus grande, voir la figure 2.2 L'unité de mesure de l'intensité est : **décibels (dB)**.

#### Exemples :

120 dB : L'avion au moment du décollage.

35 dB : maison dans quartier calme.

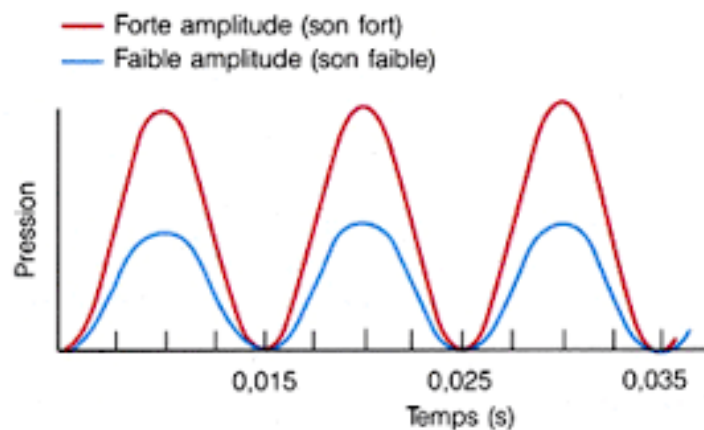


Figure 2.2: Intensité de son.

### 2.2.3 Le Pitch

Le Pitch est une propriété du son qui représente le changement de fréquence fondamentale. cette propriété donne une idée claire sur la hauteur du son.

Il établit une distinction entre le son grave et son aigu. [13]

- le pitch permet de distinguer aussi **la vitesse** du son.
- **la vitesse** du son est la vitesse à laquelle se déplacent les ondes sonores.

Voir la figure 2.3.

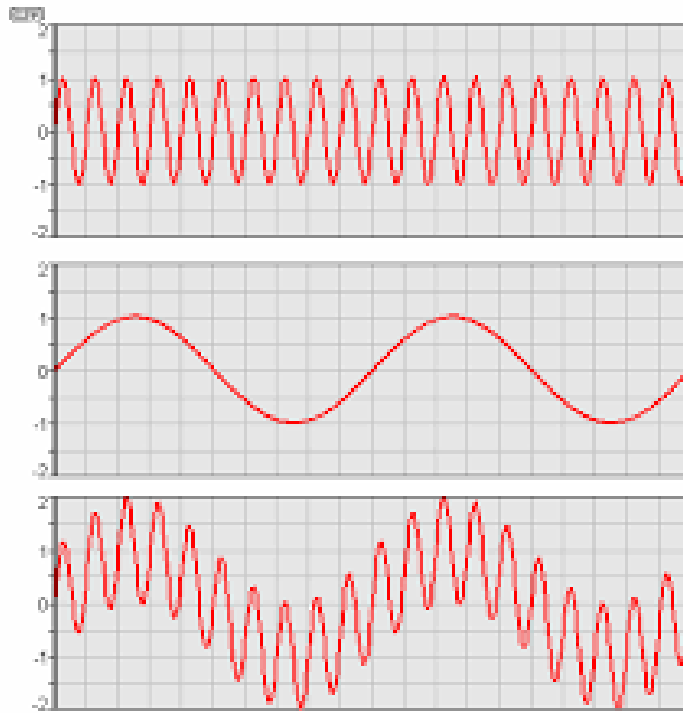


Figure 2.3: Le Pitch de Son.

### 2.2.4 Le timbre

Le timbre est un ensemble de fonctionnalités et de caractéristiques qui permettent la différenciation entre les sons.

Il est lié à des vibrations et à des résonances sortant du nez et de la gorge...etc pendant la prononciation, et qui déterminent le timbre de chaque son. [6]

L'objectif principal de cette caractéristique est la distinction entre deux sons de même hauteur et de même intensité.

Voir la figure suivante.

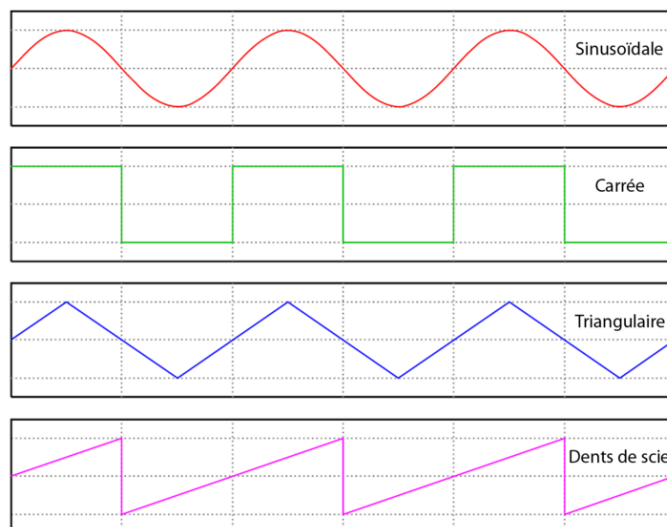


Figure 2.4: Le timbre de son.

## 2.3 Les opérations de traitement

Parmi les types d'opérations de traitement possibles et successifs sur le signal de parole, Nous voulons parler de [5]

1. opérations de traitement de bas niveau : (**La numérisation**) :

- L'échantillonnage
- La quantification
- Le codage

la figure 2.5 représente une explication schématisée de ce type de traitement.

2. opérations de traitement de haut niveau :

- La segmentation
- La paramétrisation
- La classification

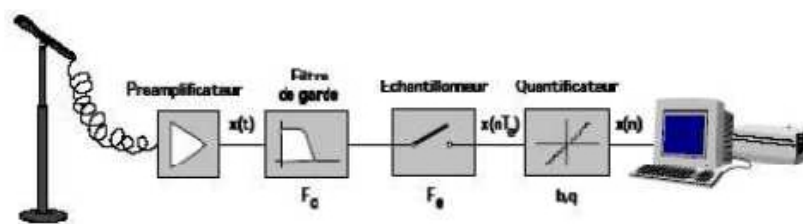


Figure 2.5: Enregistrement numérique d'un signal .[5]

### 2.3.1 La numérisation

La numérisation est une opération du traitement, dont le rôle est la conversion des informations de type complexe :

- texte
- image
- audio et vidéo
- signal électrique...etc

les informations de type numérique sont traitables par l'ordinateur.

Le convertisseur analogique- numérique ( CAN ) est conçu pour cette conversion, voir la figure 2.6.

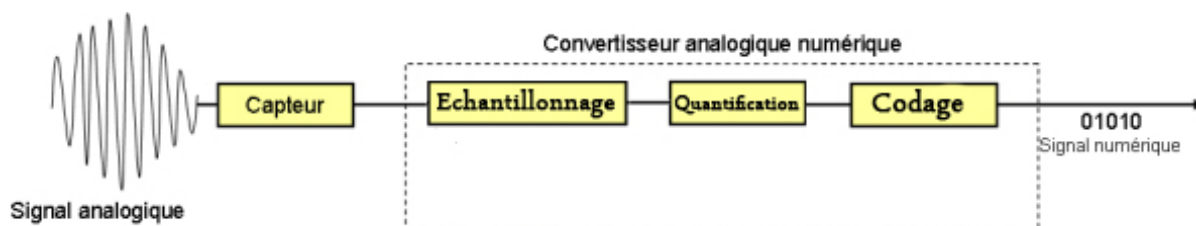


Figure 2.6: La convertisseur analogique numérique

#### 2.3.1.1 L'échantillonnage

L'échantillonnage consiste à choisir une partie de chaque ensemble d'échantillons à étudier... Il prend les valeurs du signal à intervalles réguliers, et d'une façon systématique. Afin de produire une série de valeurs discrètes.

La fréquence d'échantillonnage est d'une extrême importance dans ce processus, Lors de la grande fréquence il donne de bons résultats, Et pendant, la petite il donne des résultats inexacts.

En mathématiques, L'échantillonnage est utilisé pour changer une fonction  $f(x)$  à valeurs continues en une fonction  $f^*(x)$  discrète constituée par l'ensemble de valeurs  $a(x)$  aux instants d'échantillonnage  $= kx$  avec  $k$  est un constant.[6]

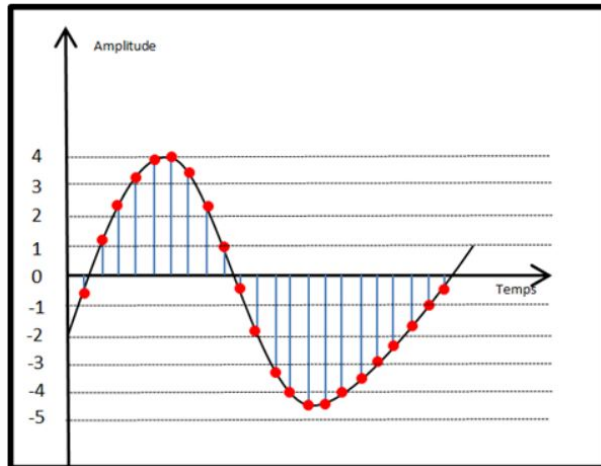


Figure 2.7: un signal échantillonné.[6]

### 2.3.1.2 La Quantification

La quantification est le processus de détermination d'une valeur spécifique à Un ensemble limité de valeurs, Afin de faciliter le traitement par ordinateur. Cette étape est de donner des valeurs rapprochées aux vraies valeurs des échantillons.

L'échelle de quantification est la plage qui fait cette approximation. Le bruit de quantification est l'erreur systématique résultant du rapprochement de ces valeurs à la valeur spécifique Grâce au travail de la quantification.

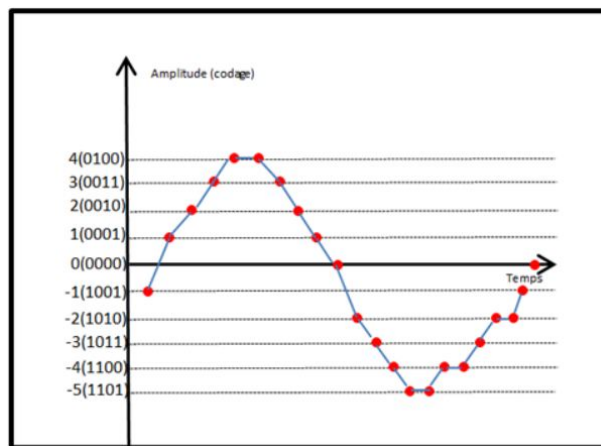


Figure 2.8: un signal quantifié.[6]

### 2.3.1.3 Le Codage

C'est la représentation binaire des valeurs quantifiées qui permet le traitement du signal sur machine. De façon générale un codage permet de passer d'une représentation des données vers une autre.

## 2.3.2 La segmentation

C'est l'opération de division d'un signal de paroles, en petites tranches appelées **segments**. Chaque tranche a ses propres caractéristiques qui permettent de le distinguer des autres. On peut identifier plusieurs types de segmentation [13] :

- la segmentation en mots ;
- la segmentation en phonèmes ;
- la segmentation en syllabes ;
- la segmentation en voisé/non-voisé ;
- la segmentation en groupes inter-pausaux ;
- la segmentation en tours de parole.

La choix de type de segmentation est fait selon l'approche utilisée et l'objectif à obtenir. voir la figure suivante

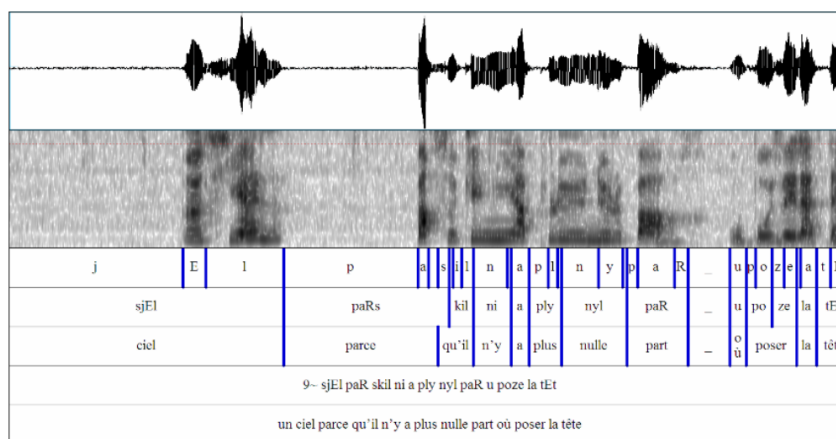


Figure 2.9: Exemples de segmentation de parole continue.[4]

### 2.3.2.1 Les methodes de segmentation

De façon générale, les méthodes de segmentation de parole peuvent être divisées en deux classes :

#### 1. les méthodes de segmentation sans connaissance :

Ce sont les méthodes de segmentation du signal de paroles sans connaissance préalable du contenu de ce signal. Où elles travaillent sur la division d'une manière générale et sans étude du contenu linguistique ou autre .

#### 2. les méthodes de segmentation avec connaissance :

Ce sont les méthodes de segmentation qui sont basées sur l'utilisation de certaines contraintes linguistiques et cognitifs, Afin de diviser le signal de paroles.[16] on va voir des exemples de ces méthodes dans la phase de classification car ces deux phases sont Interconnectées et interdépendantes

### 2.3.2.2 Les problèmes liés à la segmentation

On peut les diviser en 3 groupes principaux :

#### 1. Les problèmes liés au locuteur :

- **la variabilité inter locuteur :**
  - Age et sexe...etc
  - type d'élocution .
- **la variabilité intra locuteur :**
  - Les conditions psychologiques (stress, émotion).

— Les conditions physiques (fatigue, rhume).

## 2. Les problèmes liés au contexte :

Discours non clair.  
La vitesse d'élocution.

## 3. Les problèmes liés à l'environnement :

le bruit .  
le mouvement des choses .

### 2.3.3 La paramétrisation et l'Extraction de caractéristiques

L'objectif de cette étape est de trouver des coefficients représentatifs qui décrivent des signaux de paroles .

Nous avons besoin de convertir le signal de parole en une série de vecteurs de paramètres.

Ces paramètres doivent représenter au mieux, ce qui signifie l'extraire des informations les plus utiles pour la reconnaissance.

Cette Extraction est faite pour déterminer le contenu linguistique du signal de parole, Parce qu'il est inutile et trop variable pour être utilisé directement dans les système de reconnaissance vocale.

Ces système sont souvent utilisés un modèle Pour créer des vecteurs de paramètres acoustiques , il faut rappeler qu'il existe plusieurs modèles bien connus. [14]et[6]

#### 2.3.3.1 Les types de paramétrisations les plus connues

- Les paramètres MFCC
- Les paramètres LPCC
- Les paramètres PLP

Les raisons de leur utilisation en abondance :

- Il y a des algorithmes rapides pour les implementer
- Ces méthodes utilisent La FFT qui représente la première étape d'une analyse spectrale.
- Les bons résultats des expériences précédentes

#### 2.3.3.2 Des notions importantes :

##### Energie du signal :

Le premier caractéristique qui définit le signal de parole c'est l'énergie .

Cette énergie représente la puissance du signal.

Elle est évalué sur des parties de signaux successifs Pour pouvoir mettre en évidence les changements.

Il existe d'autres façons différentes de ce calcul ,Parmi eux :

- La première formule qui permet de calcul l'énergie est :

$$E(\text{fenêtre}) = \sum_{n \in \text{fenêtre}} |n|^2 \quad (2.1)$$

- On peut faire une modification sur la première, tel que Nous collectons les valeurs absolues des des amplitudes des échantillons afin de réduire la charge de calculations ,Et les différences sont les mêmes.

- L'un des plus largement utilisé :**celle de Taboada et al :**

Elle représente une méthode de calcul en utilisant une modification ,Où nous accordons une attention au bruit ambiant.

$$E(\text{fenêtre}) = \left( \sum_{n \in \text{fenêtre}} |n|^2 R \right) \quad (2.2)$$

**Notice :**

**R :** est la valeur moyenne de l'énergie du bruit.

Si il n'y a que le bruit de fond,alors Le résultat limite est 0.

Il y a un problème face à cette méthode ,c'est l'estimation de **R**. [12].

### 2.3.3.3 Les paramètres MFCC

**Mel-frequency cepstral coefficients** est une méthode de paramétrisation du signal de parole permet la description de ce signal à travers un nombre limité de paramètres.

Maintenant, nous décrivons chacune des étapes de **MFCC**, représentées par la Figure 2.10

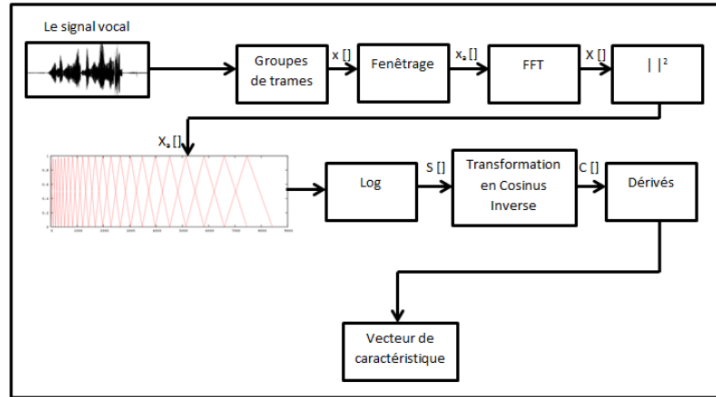


Figure 2.10: Étapes de calcul d'un vecteur caractéristique de type MFCC [6].

#### 1. Groupement en trames (Frame blocking)

Le signal de parole brut est segmenté en tranches appelés **trames** de  $N$  échantillons, avec un pas d'avancement de  $M$  trames ( $M < N$ ).

Souvent, Nous utilisons Les valeurs 10 et 20 pour  $M$  et  $N$ .

#### 2. Fenêtrage

Pour utiliser la transformation de Fourier discrète il faut transformer le signal périodique au fenêtrage du signal.

Les paramètres :

- $w(n)$  représente une fenêtre où  $0 < n < N - 1$ .
- $N$  représente le nombre d'échantillons utilisé dans chacune des trames.
- $x_a$  représente le résultat du fenêtrage du signal.

la formule utilisé; 2.3

$$x_a = x(n)w(n), 0 < n < N - 1 \quad (2.3)$$

Les fenêtres les plus utilisées sont :

- Fenêtre de Hamming : 2.4

$$w(n) = \begin{cases} 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) & 0 \leq n \leq N - 1 \\ 0 & \text{sinon} \end{cases} \quad (2.4)$$

- Fenêtre rectangulaire : 2.5

$$w(n) = \begin{cases} 1 & 0 \leq n \leq N - 1 \\ 0 & \text{sinon} \end{cases} \quad (2.5)$$

- Fenêtre triangulaire : 2.6

$$w(n) = \begin{cases} \frac{2n}{N-1} & \text{si } 0 \leq n \leq \frac{N-1}{2} \\ \frac{2(N-n-1)}{N-1} & \text{si } \frac{N-1}{2} < n < N - 1 \\ 0 & \text{sinon} \end{cases} \quad (2.6)$$

— Fenêtre de Hann : 2.7

$$w(n) = \begin{cases} 0.5 - 0.5 \cos\left(\frac{2\pi n}{N-1}\right) & \text{si } 0 \leq n \leq N-1 \\ 0 & \text{sinon} \end{cases} \quad (2.7)$$

— Fenêtre de Blackman : 2.8

$$w(n) = \begin{cases} 0.42 - 0.5 \cos\left(\frac{2\pi n}{N-1}\right) + 0.08 \cos\left(\frac{4\pi n}{N-1}\right) & \text{si } 0 \leq n \leq N-1 \\ 0 & \text{sinon} \end{cases} \quad (2.8)$$

La figure (2.11) illustre la formes de les fonctions définies ci-dessus

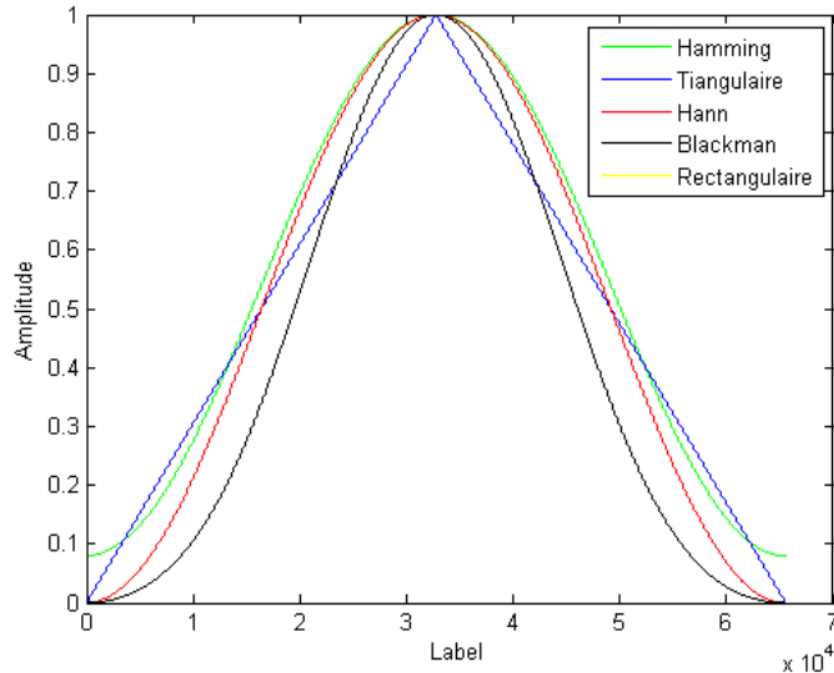


Figure 2.11: Les fonctions de fenêtrage [6].

### 3. Calcul de la transformée de Fourier rapide (Fast Fourier Transform, FFT)

Cet algorithme est habituellement utilisé dans le traitement de signal numérique. Pour convertir des données séparées du domaine temporel au domaine de fréquentiel. Donc L'objectif principal de la transformée de Fourier FFT est de convertir du domaine temporel au domaine fréquentiel. Il y a aussi un moyen facile qui permet d'inverser la conversion.

La **FFT** est un algorithme rapide qui permet le calcul de transformée de Fourier discrete (**DFT**), cette algorithme peut être implémenté en utilisant la formule suivante 2.9 :

$$x[k] = \sum_{n=0}^{N-1} x_a[n] \frac{-2j\pi}{N} kn; 0 \leq k \leq N-1 \quad (2.9)$$

**Notice :**

$X[k]$  : sont des valeurs absolues .[4]

### 4. Filtrage sur l'échelle Mel

Le spectre d'amplitude est pondéré par un banc de  $M$  filtres triangulaires espacés selon l'échelle Mel. Dans l'échelle de mesure Mel, la correspondance est approximativement linéaire sur les fréquences au-dessous de  $1kHz$  et logarithmique sur les fréquences supérieures à celle-ci. Cette relation est donnée par la formule 2.13

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \quad (2.10)$$

Le logarithme de l'énergie de chaque filtre est calculé selon l'équation 2.15 :

$$S[m] = \ln\left[\sum_{n=0}^{N-1} X_a[K]H_m[k]\right], \quad 0 \leq m \leq M \quad (2.11)$$

## 5. Calcul du cepstre sur l'échelle Mel

Le cepstre sur l'échelle de fréquence Mel est obtenu par le calcul de la transformée en cosinus discrète (equation (2.12)) du logarithme de la sortie des  $M$  filtres (reconversion du log-Mel-spectre vers le domaine temporel).

$$c[n] = \sum S[n] \cos \pi n \left(m - \frac{1}{2}\right) / M, \quad 0 \leq n \leq M \quad (2.12)$$

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 [6].

## 6. Calcul des caractéristiques dynamiques des MFCC

Les variations temporelles dans le cepstre ( $c$ ) représente un rôle très important dans la reconnaissance de parole et c'est à partir des dérivées des coefficients ( $\Delta c$ , coefficients delta ou vélocité) et des dérivées secondes ( $\Delta\Delta c$ , coefficients delta du second ordre ou accélération) des MFCC statiques que nous pouvons calculer ces variations.

En général, un système de parole utilise un échantillonnage avec une fréquence de 16 kHz pour extraire les traits suivants

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

Où :

- $c_k$  est le vecteur MFCC de la  $k^{i\text{ème}}$  trame
- $\Delta c_k = c_{k+2} - c_{k-2}$ , dérivée première des MFCCs calculée à partir des vecteurs MFCC de la  $k^{i\text{ème}} + 2$  trames et  $k^{i\text{ème}} - 2$
- $\Delta\Delta c_k = \Delta c_{k-1} - \Delta c_{k+1}$ , seconde dérivée des MFCCs. [6]

### 2.3.3.4 Les paramètres LPC

(linear Predictive Coding)Le codage prédictif linéaire représente une méthode de paramétrisation de la parole . Elle se base sur l'hypothèse que la parole peut être modélisée en utilisant un processus linéaire.

L'objectif de cette méthode est de trouver un nombre limité de paramètres pour la description du signal de parole

La raison de la célébrité de Cette méthode est que les tests ont prouvé que les systèmes de reconnaissance qui utilisent le codage prédictif linéaire ont obtenu des résultats meilleurs que les autres systèmes.

Il y a 5 phases pour l'extraction de caractéristiques du signal de parole par la méthode LPC :

- Le filtrage,
- l'échantillonnage,
- le fenêtrage,
- l'auto corrélation,
- le calcul des coefficients

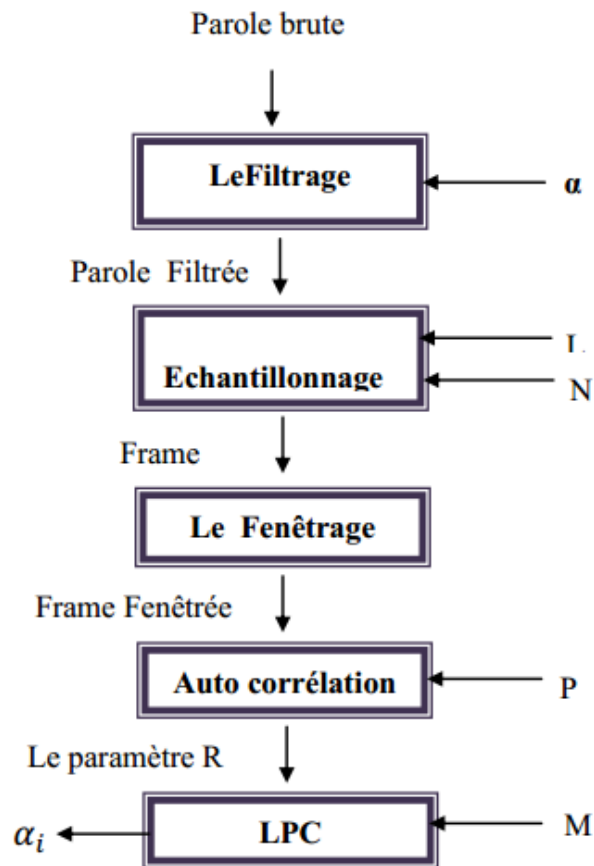


Figure 2.12: L'extraction des paramètres vocaux par LPC. [13].

### 2.3.3.5 Les paramètres PLP

La méthode PLP (Perceptually based Linear Prediction), Perceptual Linear Prediction, est une méthode adaptée à la **prédiction linéaire**.

La prédiction linéaire représente l'origine et la base de tous les types de traitement du signal de parole. La méthode PLP combine le principe du prédiction linéaire avec la représentation du signal qui suit l'échelle humaine de l'audition.

Le schéma suivant représente la fonctionnement de la méthode PLP. figure 2.13

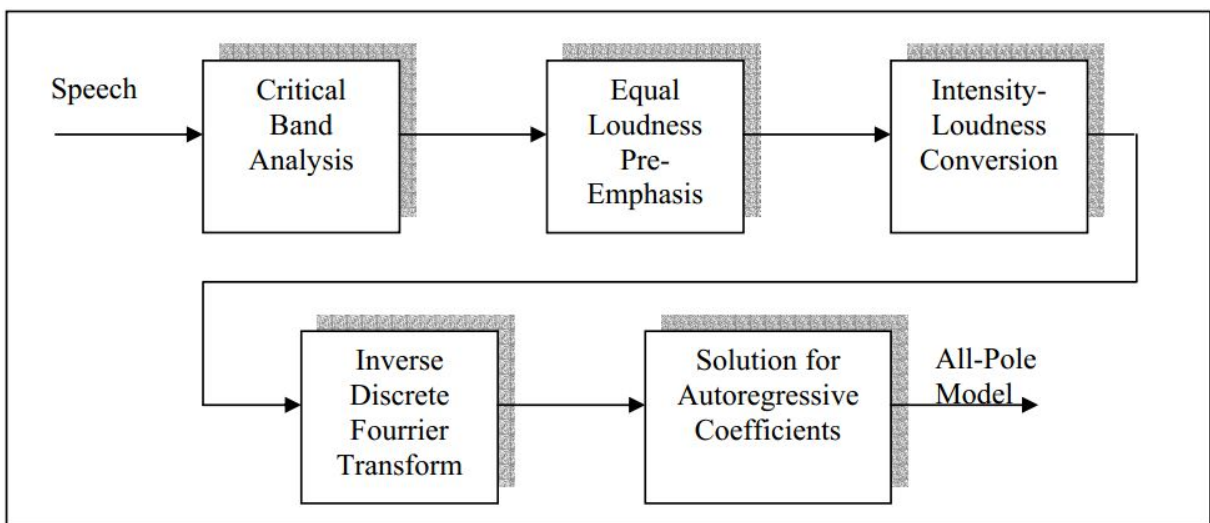


Figure 2.13: Analyse PLP [12].

Cette méthode peut être divisée en 3 étapes :

1. On analyse le signal de parole Pour obtenir un spectre suivant une échelle d'audition :
  - On fait une analyse en bandes critiques selon une échelle Bark par un banc de filtres ;
  - On fait Un prétraitement des valeurs obtenues comme les traitements effectués par l'oreille : accentuation des basses fréquences et atténuation des hautes fréquences ;
  - On applique la méthode de Steven.
2. On ajuste ce spectre à travers une transformée de fourier inverse, et puis ce signal fait un passage sur un filtre pour l'améliorer et réduire sa dimension.
  - On fait un changement des sorties des filtres du banc afin d'obtenir un spectre sur une échelle fréquentielle auditive ;
  - On applique une transformée de Fourier inverse pour retourner au domaine temporel ;
  - On applique un ensemble d'équations linéaires pour obtenir les paramètres désirés.
3. On utilise le filtrage inverse et des autres méthodes pour reconstruire le signal de parole.  
(cette étape n'est pas nécessaire.)

L'avantage de cette méthode est que les paramètres PLP résultants sont résistants au bruit que les coefficients cepstraux classiques, car ils utilisent de l'échelle fréquentielle perceptible par l'oreille. voir la figure 2.15

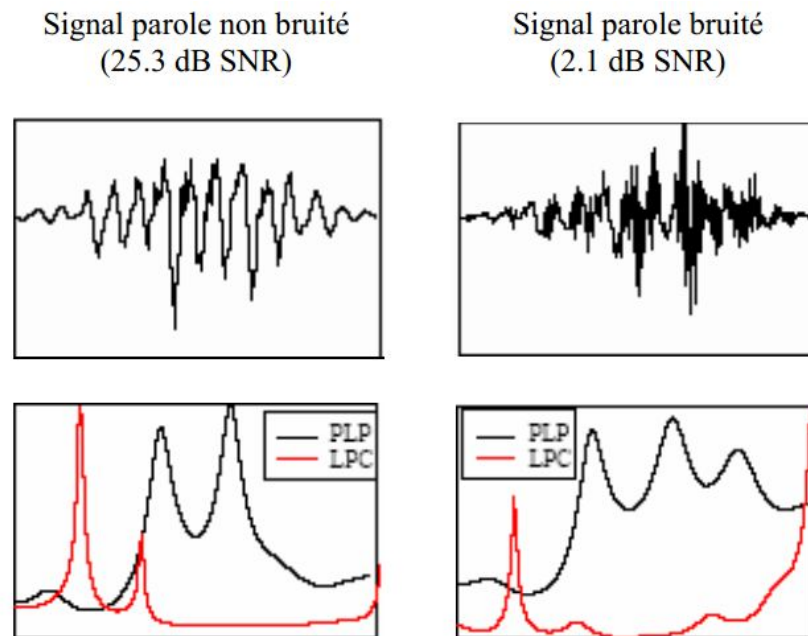


Figure 2.14: Comparaison des spectres dérivés d'une analyse PLP et LPC [12].

### 2.3.4 La classification

le traitement de la parole est un domaine de recherche scientifique renouvelé, un groupe de chercheurs ont effectué plusieurs études pour apprendre les meilleures méthodes qui donnent une meilleure classification.

La classification est un processus d'intelligence artificielle qui travaille pour que le comportement de la machine soit plus intelligent.

Chaque classifieur a besoin :

- de la définition des classes.
- des caractéristiques (attributs).
- d'un algorithme de décision.

Souvent, certains classifieurs ont un moyen pour calculer leurs performances. [6]

#### 2.3.4.1 Les phases de la classification

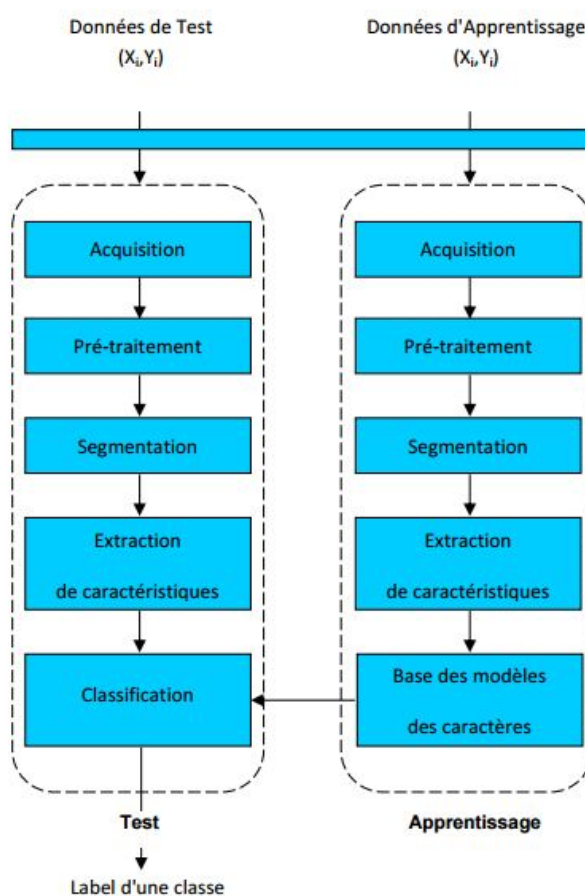


Figure 2.15: Les phases de la classification. [10].

Donc Comme nous avons noté dans la précédente figure, Il y a deux phases principales dans le processus de la classification qui :

1. **L'apprentissage :** La phase d'apprentissage est la phase la plus importante dans la reconnaissance, elle est intéressée de la création du vocabulaire (dictionnaire linguistiques).  
Les deux types d'apprentissage :
  - (1) **l'apprentissage orienté :** Dans ce cas la phase d'apprentissage est supervisée par un guide
  - (2) **l'apprentissage non orienté :** Dans ce cas la phase l'apprentissage se fait automatiquement.
2. **Test ou Décision :** La décision est la deuxième étape de reconnaissance, Lorsque la machine fonctionne pour Déterminer le modèle le plus proche dans le dictionnaire à La nouvelle entrée. cette estimation doit être en temps court que possible.

A de partir cette estimation on a la possibilité de trouver 3 états résultantes :

- Le cas d'une sortie unique : Cela est le cas des succès.
- Le cas de plusieurs sorties : la résultat est multiple.
- Le cas de non sortie : S'il n'y a pas de résultats, Tel est le cas de l'échec.

Comme nous avons dit précédemment Il y un facteur appelé le taux de classification qui permet de mesurer la performance de la classification. [13]

### 2.3.4.2 Exemples de methode de classification et segmentation

#### 1. Classification Bayésienne

Les réseaux bayésiens sont des réseaux qui nous permettent de trouver de nouveaux états de decision probabiliste à partir de connaissances incertaines. Ils resultent de la combinaison entres des domaiones scientifiques : **La probabilistes** et **la théorie de graphes**.

**Théorème de Bayes :**

$$P(B|A, c) = \frac{(P(B|c)P(A|B, c))}{(P(A|c))} \quad (2.13)$$

tels que :

- c est un contexte
- A,B deux événements
- $P(B|A,c)$  est la probabilité de B si A est vrai .
- $P(B|c)$  est la probabilité a priori de l'événement B
- $P(A|B,c)$  est la probabilité de A si B est vrai
- $P(A|c)$  est la normalisation

**Généralisation :**

Soit  $x = x_1, x_2, \dots, x_N$  un ensemble de N observations.  
 $M$  classes ( $C_1, C_2, \dots, C_M$ ) avec leur probabilité a priori.  
 $P(C_i)$ , probabilité d'avoir la classe  $C_i$ , mesuré à partir de controle les nombres d'occurrence des modèles de cette classe.

L'objectif de la classification Bayésienne est de trouver un moyen d'une décision avec moins d'erreurs Et le plus tôt possible. Dans notre cas, Le système fonctionne à trouver pour tous  $x$  ,la classe  $C_i$  le plus proche en utiliant  $P(x/C_i)$ .

**Applications des réseaux bayésiens**

- Aide au diagnostic médical.
  - Prévisions météo.
  - détection de fraudes pour les factures de téléphone...etc
- [12] Et[9]

#### 2. méthode k-ppv (K-Plus Proches Voisins)

Cette méthode de classification est basée de façon globale sur l'approximation .  
de sorte qu'elle vise à trouver pour Élément d'entrée l'élément le plus proche dans le dictionnaire .

Cette recherche est réalisée par la mesure de l'échelle de similarité plusieurs fois pour décider sur l'élément le plus proche.

Il y a beaucoup de méthodes pour la mesure de l'échelle de similarité et la méthode la plus connue est celle de la distance euclidienne.

**L'algorithme kppv :**

SetAlgoLined

---

**Algorithme 2.1:** Algorithm .

---

**Début**

**Pour** ( $exemple(x', c) \in L$ ) **Faire**  
  | Calcule la Distance  $D(x, x')$ ;  
**Pour** ( $x' \in kppv(x)$ ) **Faire**  
  | Copmte le nombre d'occurences de chaque classe  
  | Attribuer à  $x$  la classe la plus fréquence

---

tels que :

L l 'ensemble d'apprentissage.

x l'élément d'entrée. [12]et [3]

**3. méthode Markov cachés**

la première utilisation du modèle de Markov cachés HMM (Hidden Markov Model sou HMMs) dans la méthode de segmentations date de soixante ans par Baum et ses collaboratrices.

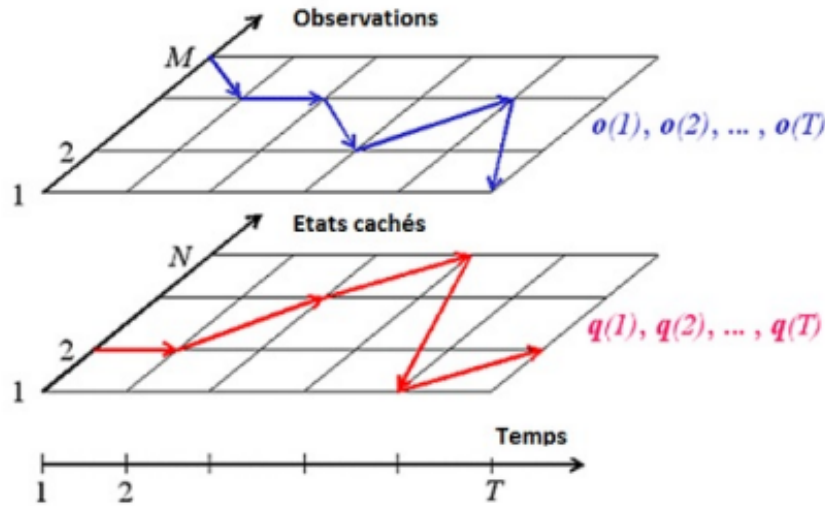


Figure 2.16: Séquence cachée et observée de la HMM [4].

 $\lambda = (S, \delta, T, G, \pi)$  ou

- $S$  : est un ensemble de  $N$  états,
- $\delta$  : est un alphabet de  $M$  symboles,
- $T = S \times S \rightarrow [0, 1]$  : est la matrice de transition, indiquant les probabilités de transition d'un état à l'autre on note  $P(s \rightarrow s_0)$  la probabilité de transition de l'état  $s$  vers l'état  $s_0$
- $G = S \times \delta \rightarrow [0, 1]$  est la matrice de génération, indiquant les probabilités de génération associées aux états on note  $P(o|s)$  la probabilité de générer le symbole  $o$  appartenant à  $\delta$  à partir de l'état  $s \in S$ .
- $\pi : S \rightarrow [0, 1]$  est un vecteur de probabilités initiales de visite

Il n'y a pas de règle stricte pour choisir l'architecture du HMM, par conséquent nous trouvons des travaux sur l'apprentissage dynamique du nombre d'états d'un Modèle de Markov Caché à des observations continues au traitement de signal et au traitement d'images. La procédure de génération d'une séquence  $o_1 \dots o_T$  de symboles à l'aide d'un HMM consiste à partir d'un état  $s$  en suivant la distribution  $\pi$ , de se déplacer d'état en état suivant les probabilités de transition, et générer un symbole sur chaque état rencontré en utilisant la distribution de probabilité de génération associée à l'état. Lorsqu'un symbole a été généré, on choisit une transition sortante suivant la distribution de probabilité de transition associée à l'état courant, et la procédure est répétée jusqu'à la  $T^{ième}$  génération de symbole.[4]

#### 4. Classification neuronale

La classification neuronale est une méthode de décision dérivée du domaine de la biologie, Elle a été découverte en surveillant le travail des neurones dans le corps humain.

Dans l'informatique, les réseaux de neurones ont plusieurs utilisations différentes En particulier, dans la reconnaissance des formes Cela est dû à plusieurs raisons comme :

- La grande efficacité
- La vitesse de recherche
- La robustesse dans les états difficiles

Cela n'empêche pas l'existence des limites de cette méthode tels que :

- Temps d'apprentissage plus long
- Le problème de la formation de Neurons : les couches de Neurons, la stratégie de connectivité [13]

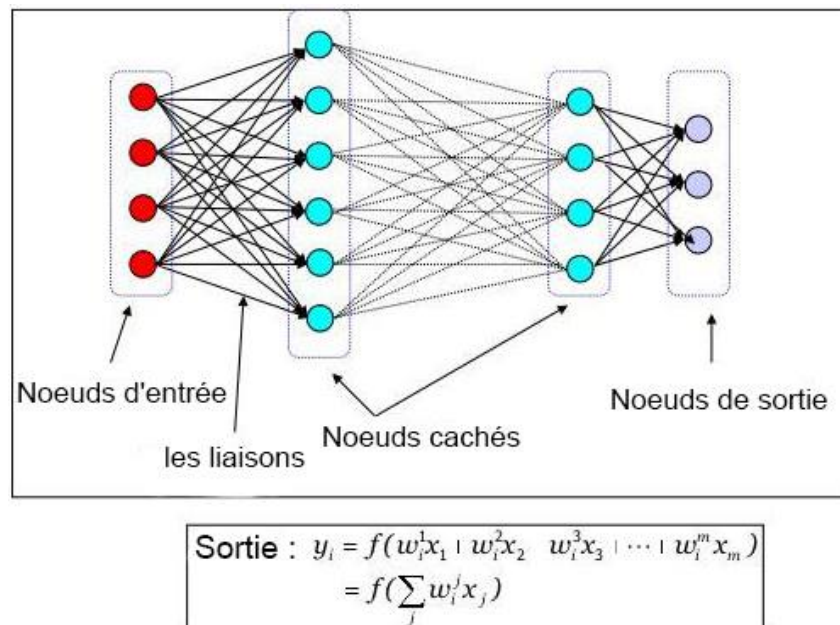


Figure 2.17: Classification neuronale.

#### 5. classificateur FLC (Fast Logic Classifier)

L'objectif principal de ce classificateur est la ressemblance fonctionnelle à la pensée humaine. Nous distinguons 2 cas :

- (1) **Cas de doute** : S'il est hésitant sur la validité de la décision.
- (2) **Cas de confirmation** : S'il est sûr de la validité de la décision.

Ce classificateur est basée principalement sur le concept de couches :

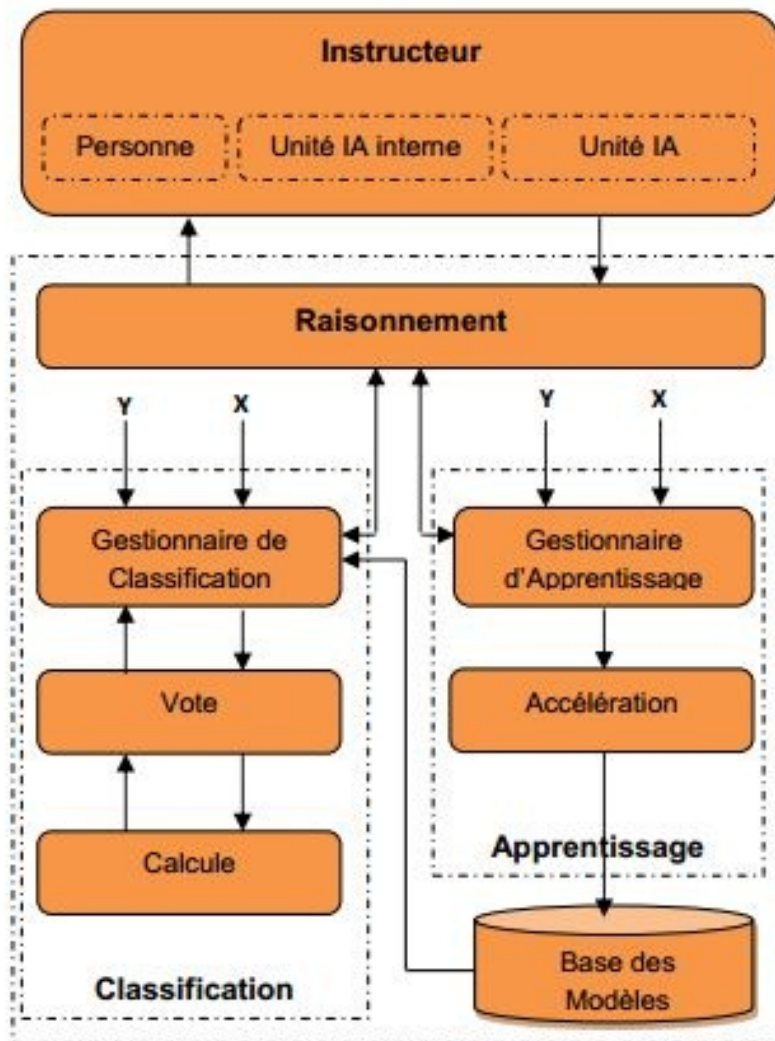


Figure 2.18: Les couches du classificateur FLC.

[10]

- **La couche de Raisonnement** Elle a tous les éléments nécessaires (classes ,propriétés,algorithmes..) afin de gérer la classification.
- **La couche d'apprentissage** : Elle met en place la base des modèles
- **La couche d'accélération** : Elle accélère le processus d'apprentissage , En utilisant la compression des données sauvegardées.
- **La couche de classification ou test** : Elle Fait avancer le processus de prise de décision . pour obtenir le résultat.
- **La couche de calcul** : EILE estime la différence de distance entre l'entrée et les modèles enregistrés.
- **La couche de vote** : Elle Choisit et nomme le plus proche parmi les modèles. [10]

## 2.4 Conclusion

Nous avons fourni dans ce chapitre un résumé des différentes méthodes les plus célèbres et largement utilisées dans le traitement de parole.

On peut dire que depuis une décennie, les techniques de traitement de la parole ont connu plusieurs grandes révolutions.

Dans un proche avenir, nos ordinateurs peuvent nous parler et reconnaître ce que nous leurs disons.

## Conception et mise en œuvre

### 3.1 Introduction

Dans les chapitres précédents nous avons vu les différentes étapes nécessaires pour construire un SRAP, ainsi qu'un panorama sur les différentes approches et méthodes de segmentation.

Dans ce chapitre, nous allons présenter une conception par affinement successif du système en donnant son architecture générale, puis nous le détaillons en étudiant séparément chacun de ses composants où nous allons voir clairement les détails de notre contribution qui se présente en un algorithme de segmentation en syllabes.

### 3.2 Mise en œuvre du système

Notre travail vise à développer une application capable de faire l'Acquisition d'un signal de parole arabe enregistré par l'utilisateur, ensuite faire un ensemble de traitements afin de segmenter en syllabes cette parole, pour faciliter l'extraction des vecteurs acoustiques, et classifier les syllabes.

Le système commence par l'acquisition et le prétraitement du signal de la parole, (éventuellement réalisées par la bibliothèque Javax.sound), ensuite, à l'application l'algorithme de segmentation proposé afin d'extraire les différentes syllabes. Par la suite il extrait un vecteur de caractéristiques de chaque syllabe. Finalement, ces vecteurs sont passés à un classificateur **FLC** afin de construire la base des modèles dans la phase d'apprentissage, ou pour connaître la classe de chaque syllabe dans la phase de décision.  
Figure : 3.1

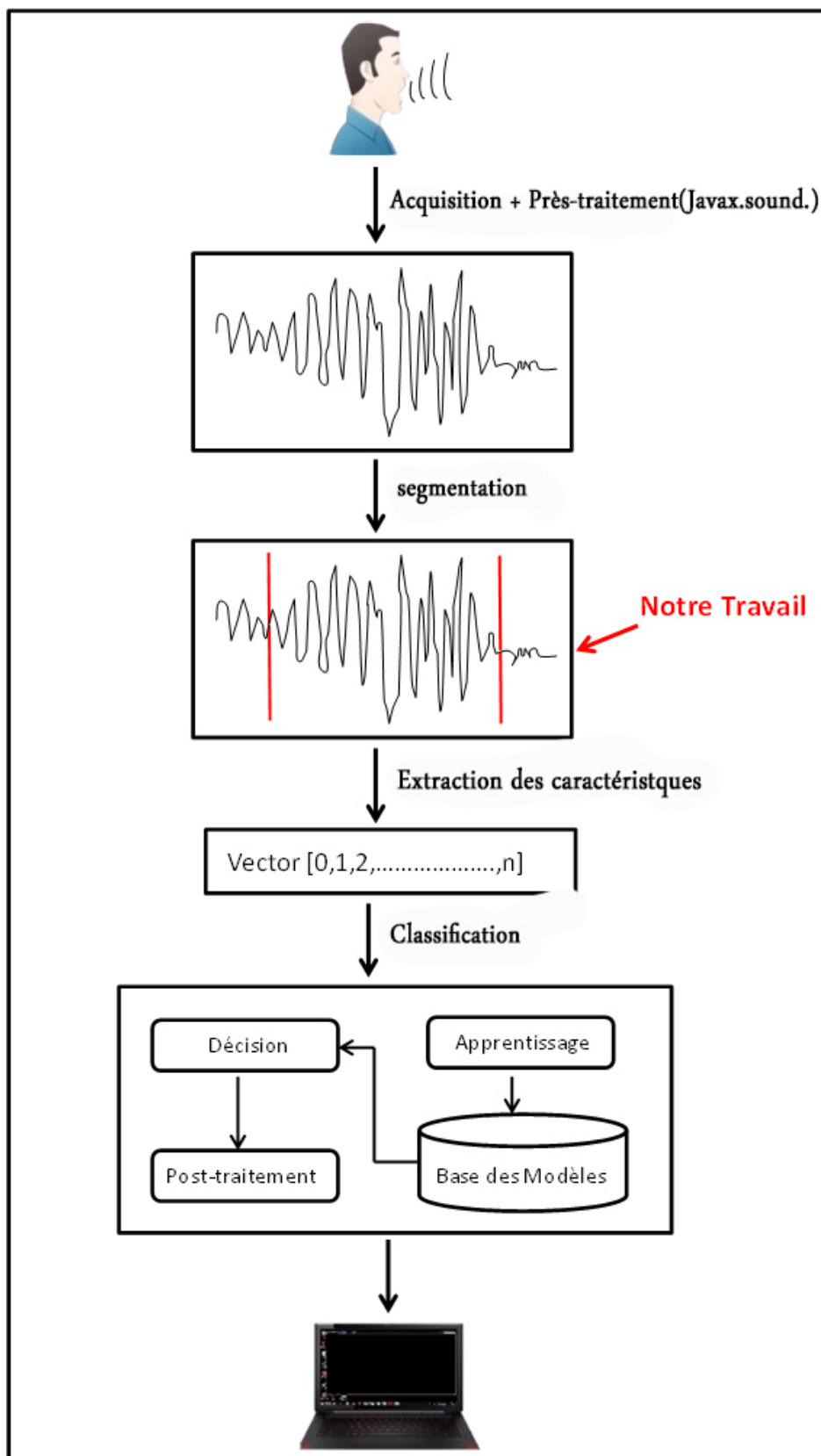


Figure 3.1: Architecture générale de notre système .

### 3.2.1 Acquisition :

La première phase de notre système ,c'est l'acquisition de signal vocal enregistré par l'utilisateur via un microphone ,ainsi la carte de micro-ordinateur permet de réaliser les étapes de traitement comme la numérisation l'échantillonnage et la quantification.

ces étapes peuvent être réalisées par l'utilisation de la bibliothèque nommé **javax.sound** , cette bibliothèque peut être appelée à travers la fonction **import javax.sound.\***.

Pour commencer l'aquisition nous avons besoin de lancer un thread appelé **capture** , L'algorithme suivant représente la méthode **run()** de ce thread :

---

**Algorithme 3.1:** Algorithm de Capture.

---

```
milliseconds : entier;
numBytesRead : entier;
frameSizeInBytes, bufferLengthInFrames, bufferLengthInBytes : entier;
data : tableau[bufferLengthInBytes] de byte;
Début
    audioInputStream ← nil
    info ← DataLine(TargetDataLine.class, format);
    line ← (TragetLine)Audiosystem.getLine(info);
    line.open(format, line.getBufferSize());
    out ← ByteArrayOutputStream();
    frameSizeInBytes ← format.getFrameSize();
    bufferLengthInFrames ← (line.getBufferSize())/8;
    bufferLengthInBytes ← (bufferLengthInFrames * frameSizeInByte);
    line_start();
    numBytesRead ← line_read(data, 0, bufferLengthInBytes);
    while (thread <> nil) do
        Si (numBytesRead >= 0) Alors
            | out_write(data, 0, numBytesRead);
        FinSi
    end
    /* fermer la ligne */
    line_stop();
    line_close;
    line ← nil;
    out_flush();
    out_close();
    audioBytes ← out;
    bais ← ByteArrayInputStream(audioBytes);
    audioInputStream ←
        AudioInputStream(bais, format, audioBytes.length/frameSizeInBytes);
    milliseconds ← ((audioInputStream.getFrameLength() * 1000)/format.getFrameRate())
    /* pour reproduire le son */
    audioInputStream_reste();
    /* pour Afficher la graphe */
    afficher();
Fin
```

---

### 3.2.2 Méthode de segmentation proposée :

Cette phase a pour but d'extraire les segments acoustiques nécessaires pour la phase de décision.

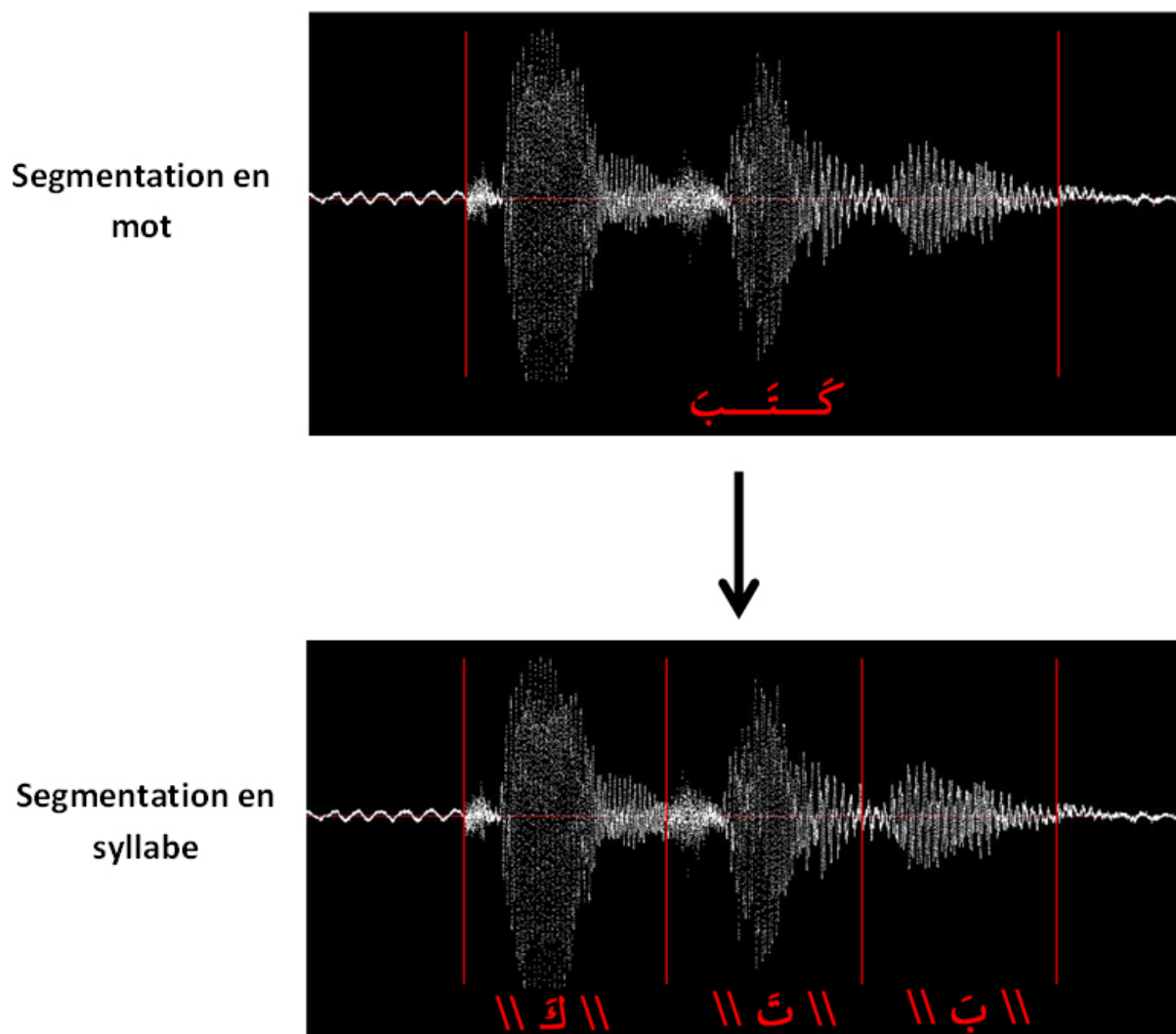


Figure 3.2: Segmentation en mots VS segmentation en syllabes .

#### 3.2.2.1 Description de l'algorithme proposé :

A ce niveau, nous avons proposé un nouveau algorithme qui permet d'identifier le début et la fin de chaque syllabe du signal de la parole

Les principales étapes de cet algorithme sont :

1. Identification de la valeur maximale d'intensité (volume) dans le tableau du signal capté.
2. Calcul la période de phase de cette valeur ,après on cherche les valeurs maximales Proches des autres phases droites dans une période spécifiée scientifiquement par une relation spéciale.  
Ce processus est pour déterminer plus tard la valeur maximale de la région droite, et on calcule encore la période entre la valeur maximale totale et la valeur maximale de la région pour suivre l'évolution des valeurs maximales à partir de ce periode jusqu'à la condition d'arrêt.
3. Enfin on répète le processus jusqu'à déterminer les limites de la syllabe, ainsi on fait le post-traitement et on remplit les valeurs des syllabes dans la table par zéros .  
il faut répéter la fonction plusieurs fois pour déterminer tous les syllabes.

Nous résumons notre algorithme de la segmentation de la parole arabe en syllabes par la figure :3.11

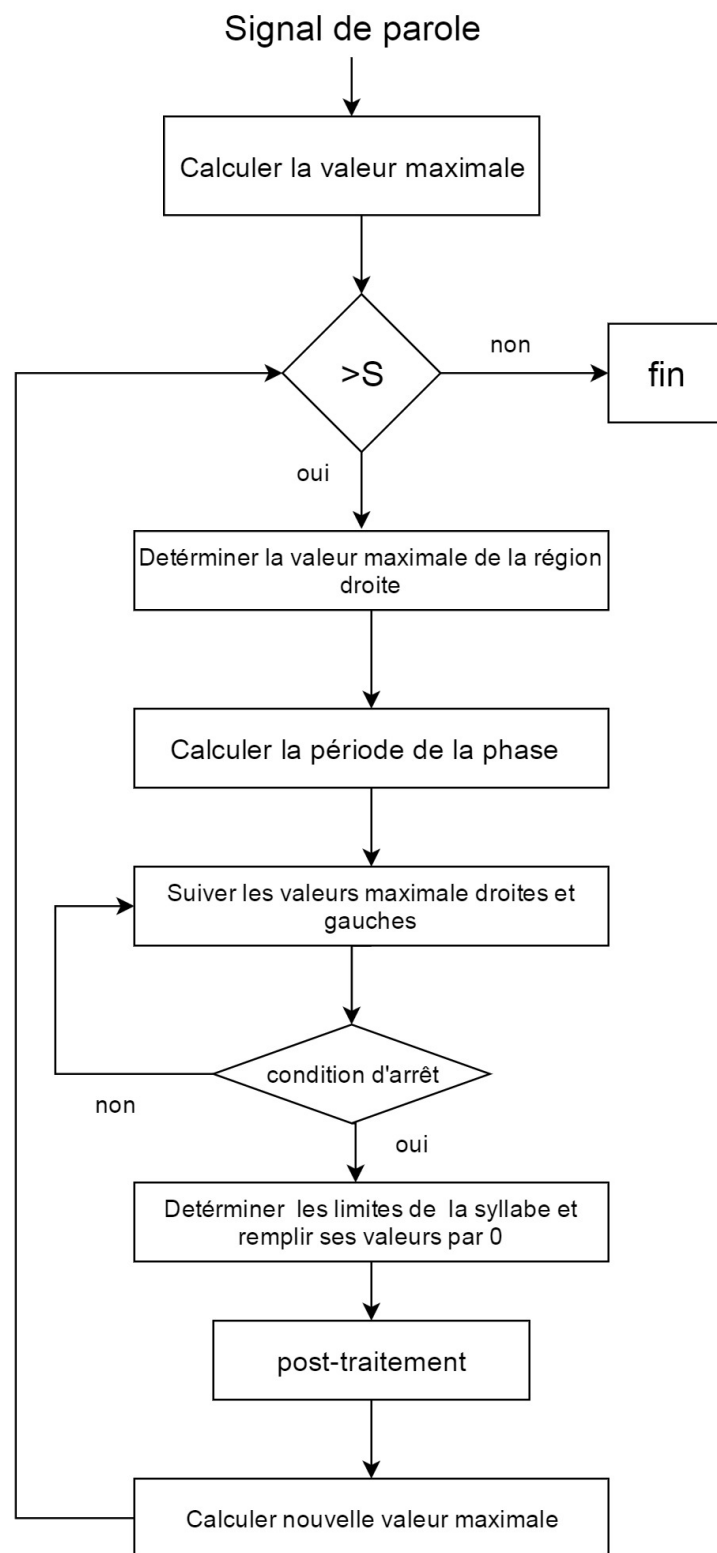


Figure 3.3: Algorithme général de la segmentation .

**Note :** S représente le seuil utilisé.

## 1. Identification de la valeur maximale

La première étape de l'algorithme consiste à trouver la valeur maximale de la forme des ondes pour l'appliquer sur notre algorithme. (la valeur doit être supérieure au, seuil ). Figure :3.4

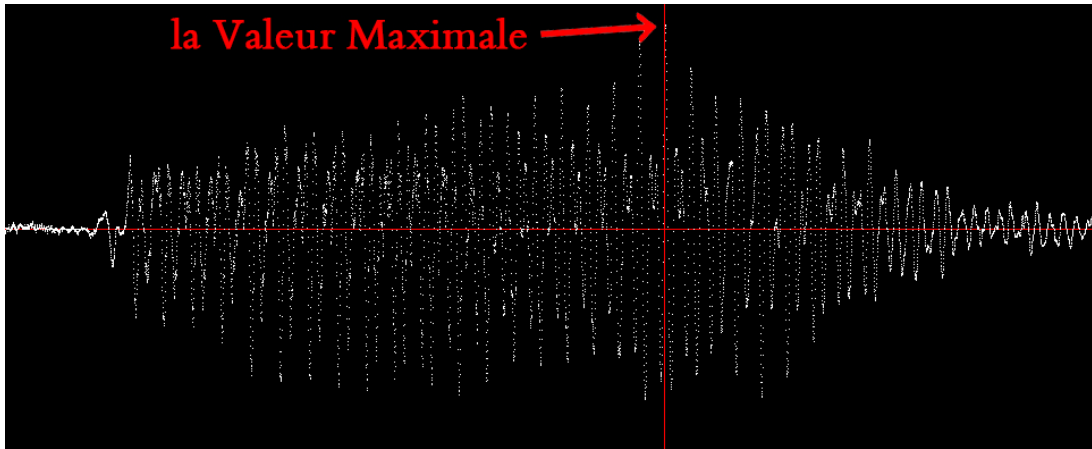


Figure 3.4: Exemple de valeur maximale .

Voici l'algorithme :

---

**Algorithme 3.2:** Algorithme de la valeur maximale.

---

**Fonction**  $max\_ind(A, B : entier; t : Tableau[] \text{ de } entier) : entier$

  |  $i, max, ind\_max : entier$

**Début**

  | **Pour**  $i \leftarrow A$  **Jusqu'à**  $B$  **Faire**

    | **Si**  $((t[i] > max)$  **Alors**

      |  $max \leftarrow t[i];$

      |  $ind\_max \leftarrow i;$

    | **FinSi**

  | **FinPour**

  |  $Retourner \leftarrow ind\_max;$

**Fin**

---

## 2. Calcul de la période de phase de la valeur maximale :

La deuxième étape est la détermination de la frontière gauche et de la frontière droite de la phase de la valeur maximale ,après on calcule la période entre ces frontières facilement. Figure :3.5

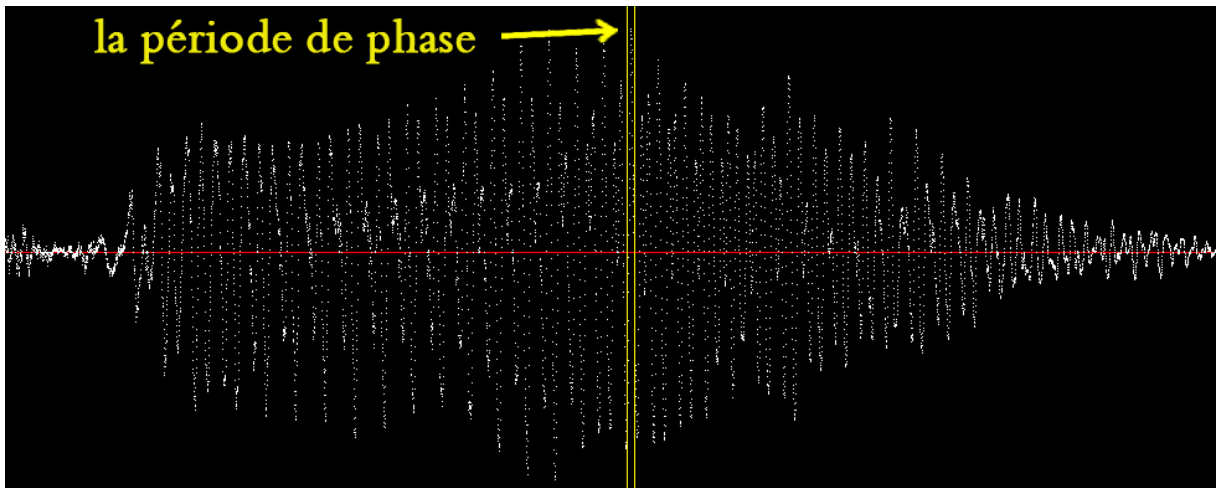


Figure 3.5: Exemple de période de phase de la valeur maximale.

Le tableau T contient les valeurs des variations de puissance acoustique (volume) du signal enregistré.

Voici l'algorithme :

---

**Algorithme 3.3:** Algorithme de frontière droite de la phase.

---

**Fonction** *int\_0\_droit* (*A* : entier; *t* : Tableau[] de entier): entier

**Début**

```
| j : entier  
| j = A  
| while (t[j] > 0) do  
| | j = j + 1;  
| end  
| Retourner j;
```

**Fin**

---

---

**Algorithme 3.4:** Algorithme de frontière gauche de la phase.

---

**Fonction** *int\_0\_gauche* (*A* : entier; *t* : Tableau[] de entier): entier

**Début**

```
| j : entier  
| j = A  
| while (t[j] > 0) do  
| | j = j - 1;  
| end  
| Retourner j;
```

**Fin**

---

---

**Algorithme 3.5:** Algorithme de la période de la phase.

---

**Fonction** *periode* (*i, j* : entier;): entier

**Début**

```
| Retourner abs(i - j)
```

**Fin**

---

### 3. Calcul de la période de phase de la forme répétée :

elle est réalisée à travers trois étapes :

- Chercher les valeurs maximales locales dans des périodes spécifiés par la formule suivante :

$$[tm + f * T0, tm + (2 - f) * T0] \quad (3.1)$$

tels que :

- (1) **tm** est la valeur maximale actuelle (variable dans chaque pas),
  - (2) **T0** représente la période calculée précédemment,
  - (3) **f** est une constante entre 0.5 - 0.9;[11]
- mettre ces valeurs dans un tableau et sélectionner la valeur maximale de ce tableau c'est à dire la région droite .
  - Calculer la période entre la valeur maximale totale et la valeur maximale de la région droite ,c'est la période voulue .

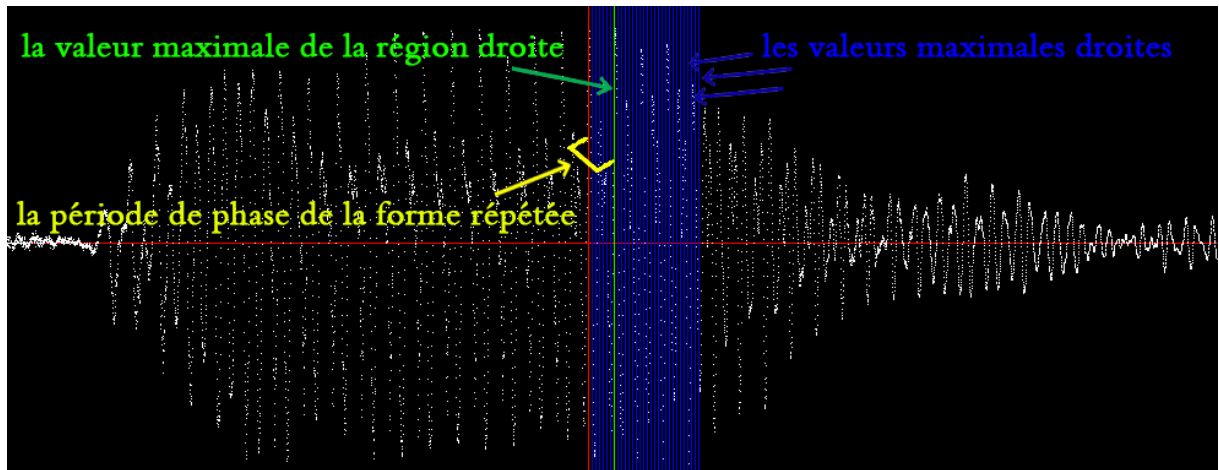


Figure 3.6: Exemple de période de phase de la forme répétée .

l'algorithme utilisé :

---

**Algorithme 3.6:** Algorithme de la période de phase de la forme répétée.

---

```

max_total, droit, gauche, periode, max_total_d : entier
ai, max_total_2_d, max_region_d, periode2 : entier
remplac_skBuffer : tableau[ ] de entier
tab_max_local_d : tableau[30] de entier
Début
  max_total ← max_ind(skBuffer, 0, skBuffer.length - 1);
  Pour i ← 0 Jusqu'à skBuffer.length Faire
    | remplac_skBuffer[i] ← skBuffer[i];
  FinPour
  droit ← ind_0_droit(remplac_skBuffer, max_total);
  gauche ← ind_0_gauche(remplac_skBuffer, max_total);
  periode ← periode(droit, gauche);
  max_total_d ← max_total;
  i ← 0;
  while (i < 30) do
    | ai ← max_abs_ind(remplac_skBuffer, max_local_d + (0.5 * periode), max_local_d +
    |   (1.5 * periode));
    | max_local_d ← ai;
    | tab_max_local_d[i] ← max_local_d;
    | i ++;
  end
  max_total_2_d ← ind_max_special(remplac_skBuffer, tab_max_local_d);
  max_region_d ← max_total_2_d;
  periode2 ← abs(max_total - max_total_2_d);
Fin

```

---

**Remarque :**

Nous n'avons pas besoin de répéter ce processus dans le coté gauche car le signal est périodique, c'est à dire les deux périodes entre les valeurs maximales des deux régions sont les mêmes.

#### 4. Recherche de toutes les valeurs maximales dans les zones gauche et droite :

Avec l'aide de la période que nous avons calculée précédemment et l'équation 3.1 pour La séquence de cette période (chaque fois en mutation), Nous suivons ces valeurs jusqu'à la condition d'arrêt.

voir la figure :3.7

**Condition d'arrêt :**

$$VMA < ((VMT - 50)/6) + 20 \quad (3.2)$$

VMA :la valeur maximale actuelle

VMT :la valeur maximale totale

**Note :** Nous avons défini cette condition à travers l'expérimentation.

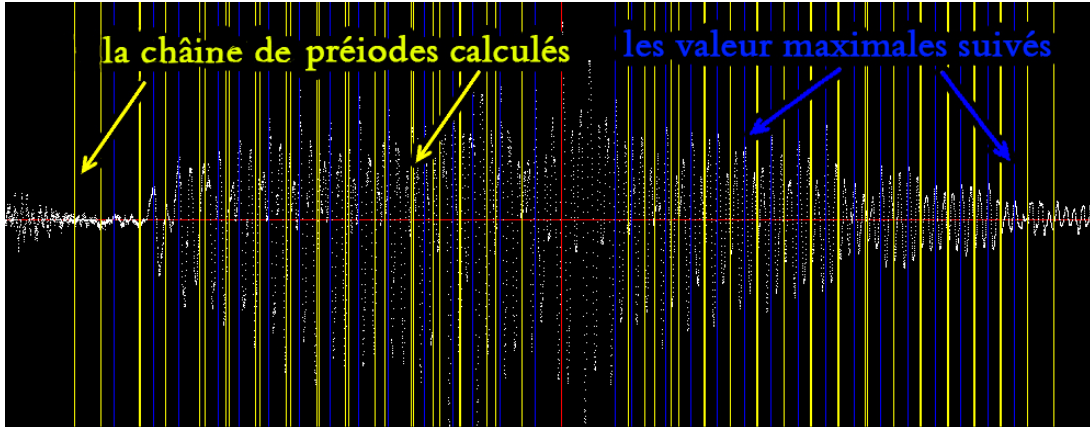


Figure 3.7: Valeurs maximales dans les zones gauche et droite .

Il faut ajouter cet algorithme :

---

**Algorithme 3.7:** Condition d'arret.

---

```

/* Condition d'arret droit */
while (remplac_skBuffer[max_region_d] > ((remplac_skBuffer[max_total] - 50)/6) + 20) do
    ai ← max_ind(remplac_skBuffer, (max_region_d + (0.5 * periode2)), max_region_d + (1.5 *
    periode2));
    max_region_d ← ai;
    painVL(jLabel11.getGraphics(), max_region_d, color.blue);
end
/* Condition d'arret gauche */
max_ragion_g ← max_total;
while (remplac_skBuffer[max_region_g] > ((remplac_skBuffer[max_total] - 50)/6) + 20) do
    ai ← max_ind(remplac_skBuffer, (max_region_g - (1.5 * periode2)), max_region_g - (0.5 *
    periode2));
    max_region_g ← ai;
    painVL(jLabel11.getGraphics(), max_region_g, color.blue);
end

```

---

#### 5. Définition des limites de la syllabe et la mise en oeuvre du post-traitement :

Avec l'enregistrement des caractéristiques de syllabes et le remplissage de leurs valeurs par zéros dans le tableau de signal pour la recherche de nouvelle valeur maximale ,c'est à dire nouvelle syllabe.

**Note :**

Le post-traitement : Un test de syllabe en termes de longueur et d'énergie permet de savoir si les conditions fixiées sont réalisées.

la classe **syllabe** pour enregistrer les syllabes :

---

**Algorithme 3.8:** la classe syllabe.

---

*begin, end, max, max2 : entier;*

**Procédure** *syllabe* (*beg, en, mx, mx2 : entier*)

**Début**

        | *begin*  $\leftarrow$  *beg*;  
        | *end*  $\leftarrow$  *en*;  
        | *max*  $\leftarrow$  *mx*;  
        | *max2*  $\leftarrow$  *mx2*;

**Fin**

**Fonction** *getBegin* ()

**Début**

        | Retourner *begin*;

**Fin**

**Fonction** *getEnd* ()

**Début**

        | Retourner *end*;

**Fin**

**Fonction** *getMax* ()

**Début**

        | Retourner *max*;

**Fin**

**Fonction** *getMax2* ()

**Début**

        | Retourner *max2*;

**Fin**

**Fonction** *compare\_int* (*b : entier*)

**Début**

        | **Si** (*getBegin* = *b*) **Alors** Retourner 0 ;

        | **Sinon Si** (*getBegin*() > *b*) **Alors**

            | Retourner 1

        | **FinSi**

        | **Sinon**

            | Retourner -1

        | **FinSi**

**Fin**

---

l'algorithme utilisé pour le post-traitement et l'enregistrement :

---

**Algorithme 3.9:** Post-traitement.

---

*periode\_car*  $\leftarrow$  *abs(max\_region\_d - max\_region\_g)*;

**Si** (*periode\_car* < 250) **Alors**

*Arrays.fill(ramplac\_skBuffer, max\_region\_g, max\_region\_d, 0)*;

*max\_total*  $\leftarrow$  *max\_ind(remplac\_skBuffer, 0, remplac\_skBuffer.length - 1)*;

**Sinon**

    | *car*  $\leftarrow$  *carractere(max\_region\_g, max\_region\_d, max\_total, max\_total\_2\_d)*;

    | *list\_car.add(car)*;

    | *Arrays.fill(ramplac\_skBuffer, max\_region\_g, max\_region\_d, 0)*;

    | *max\_total*  $\leftarrow$  *max\_ind(remplac\_skBuffer, 0, remplac\_skBuffer.length - 1)*;

**FinSi**

---

6. On répète ce processus jusqu' à ce que la nouvelle valeur maximale soit inferieure au seuil fixe .

### 3.2.3 Extraction des caractéristiques :

Nous avons déjà parlé de façon détaillée sur les méthodes d'extraction des caractéristiques et nous avons aussi dit que l'expérience a montré que la meilleure méthode donnant les meilleurs résultats est la méthode **LPC**. Mais quand nous avons essayé cette méthode nous l'avons trouvée inefficace avec nos segments car les résultats que nous avons obtenus sont très mauvais . Pour cette raison, nous avons besoin de créer une autre méthode.

#### 3.2.3.1 Notre méthode proposée :

Au cours de nos tentatives dans la phase de segmentation ,nous avons remarqué que chaque segment (syllabe) a une forme répétée qui permet de le distinguer. Alors nous avons décidé de faire la distinction entre les syllabes a partir cette forme. voir la figure 3.8

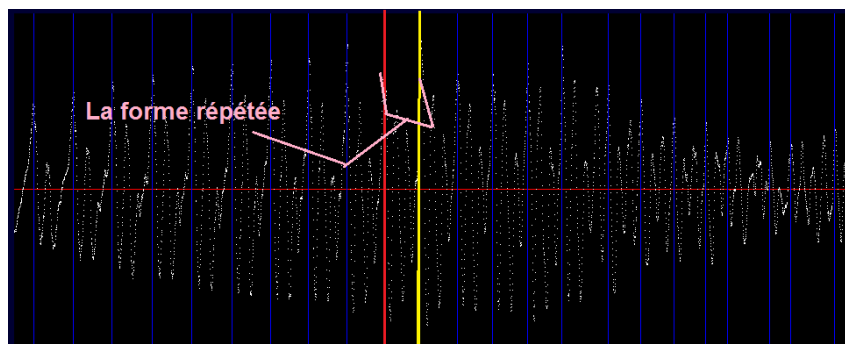


Figure 3.8: Exemple de forme répétée .

Nous avons remarqué que les formes répétées ont des longueurs différentes ,donc nous avons appliqué **la normalisation** pour fixer leurs longueurs. voir la figure suivante.

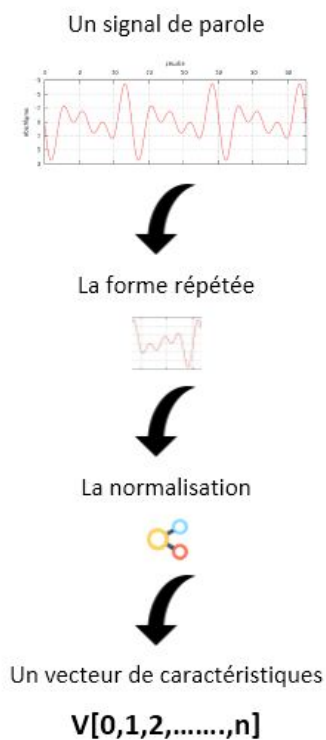


Figure 3.9: la schéma de l'extraction des caractéristiques .

On peut dire que les résultats obtenus par cette méthode ne sont pas ceux escomptés. Mais Au moins, ils sont mieux que les résultats de **LPC**.

### 3.2.3.2 Normalisation :

est une méthode qu'est appliquée sur les segments obtenues par la phase de segmentation afin de corriger les variations systématiques dans la longueur et l'amplitude entre les même échantillons. Elle consiste à fixer les tailles des segments pour réduire ces variations .

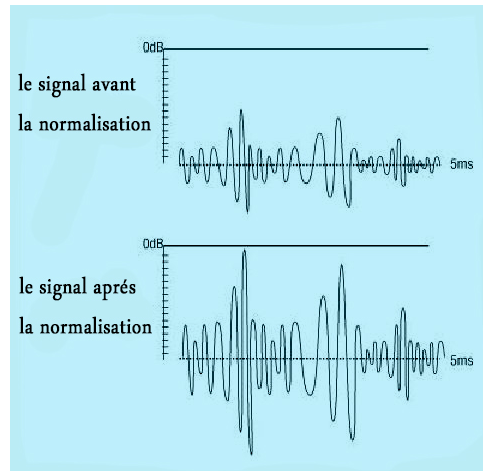


Figure 3.10: La normalisation .

l'algorithme de la normalisation :

---

**Algorithme 3.10:** Algorithme de la normalisation.

---

**Fonction** *normalize* (*s* : Tableau[ ] de entier, *nv*, *nh* : entier)

*nb* : Tableau[*nh*] de entier;  
    *x*, *star*, *end*, *size*, *max\_val*, *max\_value*, *v1*, *v2*, *cc*, *i* : entier;  
    *pos*, *hs* : float;

**Début**

*star*  $\leftarrow$  0; *end*  $\leftarrow$  *s.length*;  
    *x*  $\leftarrow$  0; *v1*  $\leftarrow$  0; *v2*  $\leftarrow$  0; *cc*  $\leftarrow$  0;  
    *pos*  $\leftarrow$  0.0;  
    *size*  $\leftarrow$  *end* - *star*;  
    *max\_val*  $\leftarrow$  *max\_valeur*(*s*, *star*, *end*);  
    *hs*  $\leftarrow$  ((*nh* + 0) / *size*);  
    *v1*  $\leftarrow$  *floor*((*s*[*star*] \* *nv*) / *max\_val*);  
    *nb*[*x*]  $\leftarrow$  *v1*;  
    *max\_value*  $\leftarrow$  *v1*;

**Pour** (*i*  $\leftarrow$  (*star* + 1) **Jusqu'à** *i* < *end*) **Faire**

*v2*  $\leftarrow$  *floor*((*s*[*i*] \* *nv*) / *max\_val*);  
        **Si** (*abs*(*v2*) > *max\_value*) **Alors**  
            *max\_value*  $\leftarrow$  *abs*(*v2*);  
        **FinSi**  
        *nb*[*x*]  $\leftarrow$  *v2*;  
        *pos*  $\leftarrow$  *pos* + *hs*;  
        *x*  $\leftarrow$  *floor*(*pos*);

**FinPour**

    Retourner *nb*;

**Fin**

---

tels que :

*nh* : la longueur fixe (la taille du segment)

*nv* : la hauteur fixe (la valeur maximale dans ce segment)

### 3.2.4 Classification :

Dans cette phase de notre système Nous avons choisi d'utiliser la classification **FLC** (Fast Logic Classifier ).  
Comme nous le savons, il y a deux phases de base : L'apprentissage et la décision.  
La première phase consiste à mettre en place la base d'exemples (Vocabulaire). Et la deuxième phase consiste à attribuer un nouvel élément à la classe la plus proche .

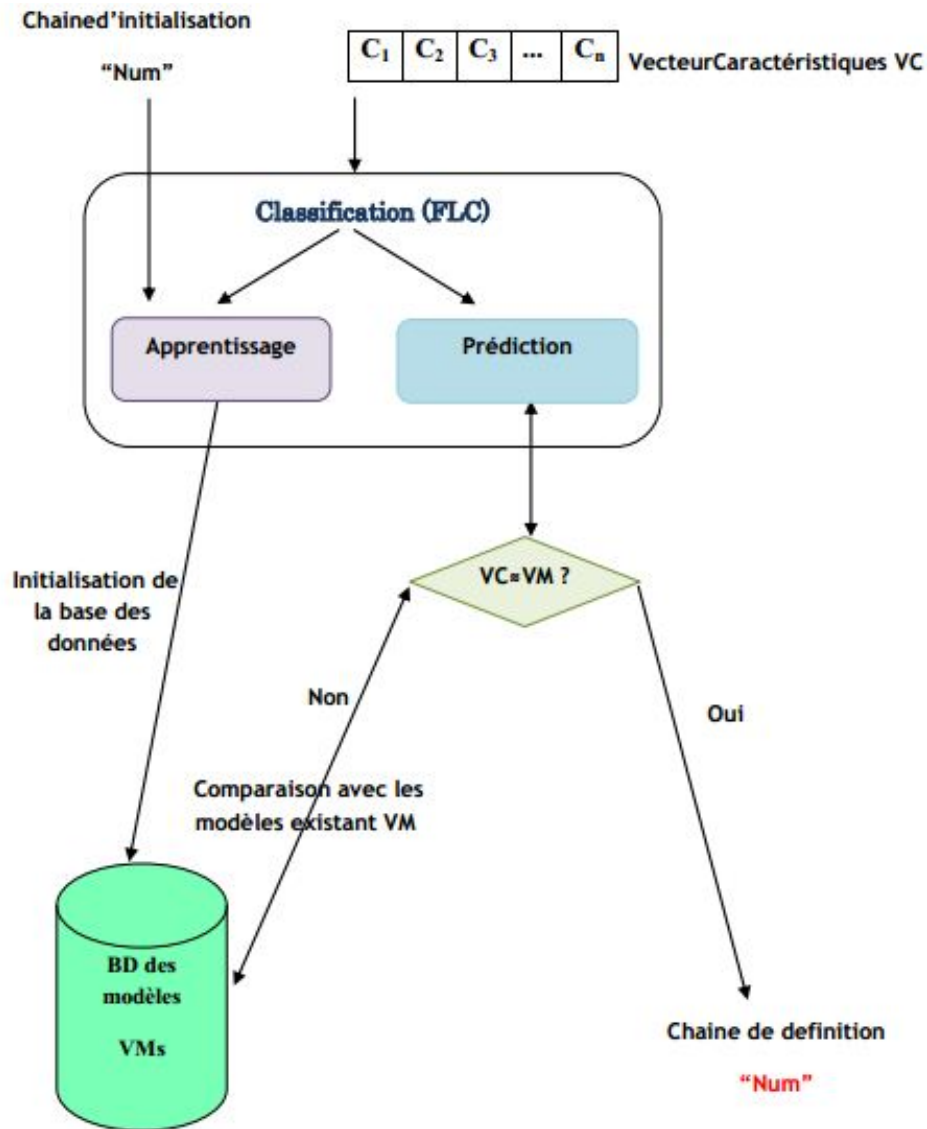


Figure 3.11: Classification par FLC .  
[13]

### 3.3 Conclusion

Dans ce chapitre, nous avons présenté premièrement l'architecture globale du système, ensuite, nous avons présenté en détail les différentes phases. Parmi ces phases nous avons vu les détails de notre contribution qui se présente par un algorithme de segmentation du signal de la parole en syllabes. Dans le prochain chapitre nous allons voir les résultats de segmentation de cet algorithme.

## Résultats et discussion

### 4.1 Introduction

Après avoir expliqué les étapes de notre système, Nous allons maintenant concentrer sur l' évaluation de notre algorithme et le comparer avec les travaux précédents, Sans oublier de donner quelques éclaircissements et informations sur l'environnement de la programmation (IDE) , et sur l'interface global du système avec les fonctionnalités offertes .

### 4.2 Choix du Language de programmation

Nous avons choisi le langage JAVA comme environnement de programmation, car ce langage possède des avantages très intéressants.

#### 4.2.1 Pourquoi JAVA ?

Le langage Java est à la base de la plupart des applications mobiles et imbriquées, des jeux, du contenu Web et des logiciels d'entreprise. Utilisée par plus de 9 millions de développeurs dans le monde, Le langage Java permet de développer, de déployer et d'utiliser efficacement des applications et des services fascinants.

Java est aujourd'hui devenu un outil indispensable En raison des caractéristiques merveilleuses qui distinguent ses bibliothèques, :

- Simple, Robuste, Dynamique et Sécurisé
- Indépendant de la Plateforme (VM)
- Semi Compilé/Semi Interprété
- La portabilité des logiciels

Il existe plusieurs outils de développement visuels (IDE), tels que [4] :

- NetBeans ( SUN)
- Eclipse ( IBM)
- IDEA (IntelliJ)

### 4.3 Interface du système

La figure ci-dessous ,présente l'interface utilisateur globale du système . Cette interface est divisée en trois sections :

- interface d'affichage sous forme d'ondes.
- ensemble des boutons pour l'affichage.
- ensemble des boutons pour le traitement.

Voir la figure : 4.1

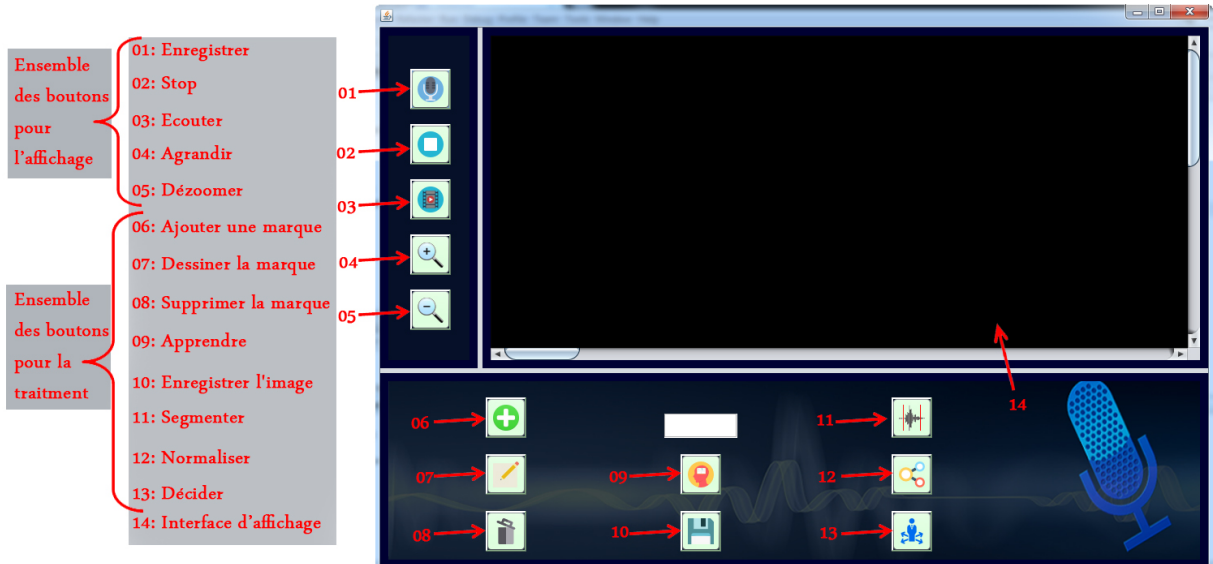


Figure 4.1: L'interface utilisateur globale du système .

### 4.4 Fonctionnalité du système

Notre application permet d'enregistrer la parole et la segmenter, pour faire cela :

1. L'utilisateur clique sur le bouton **numéro 1** et commence l' enregistrement de sa parole ,cette enregistrement dure dix secondes (on peut changer la durée d'enregistrement mais nous avons la fixé pour améliorer le processus de segmentation).

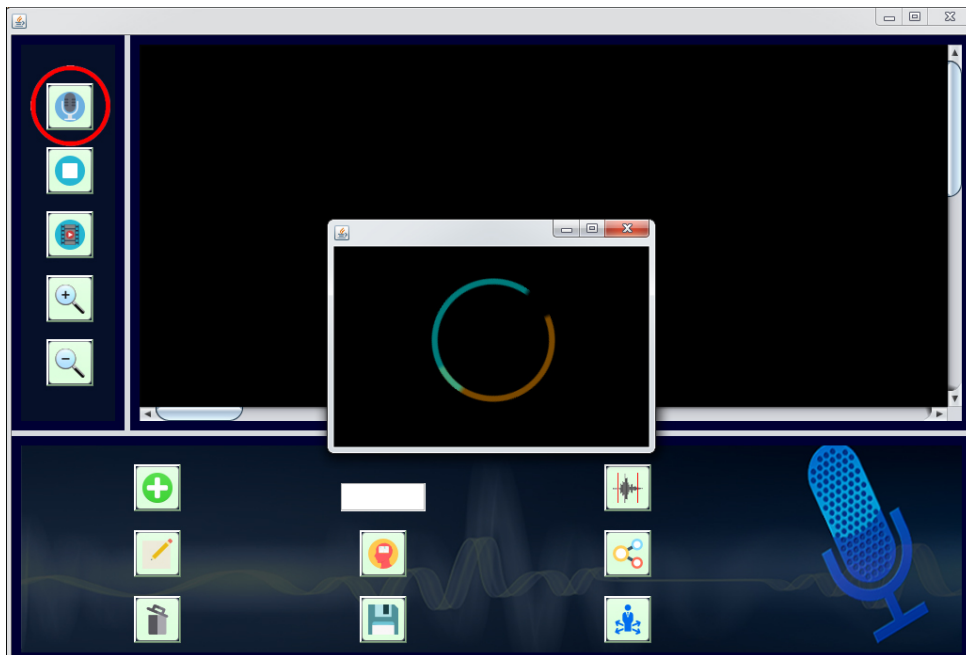


Figure 4.2: L'enregistrement de parole .

2. Nous avons fusionné la segmentation avec l'affichage .  
Pour que l'utilisateur puisse voir plus d'informations sur la segmentation il doit cliquer sur le bouton **numéro 11**.

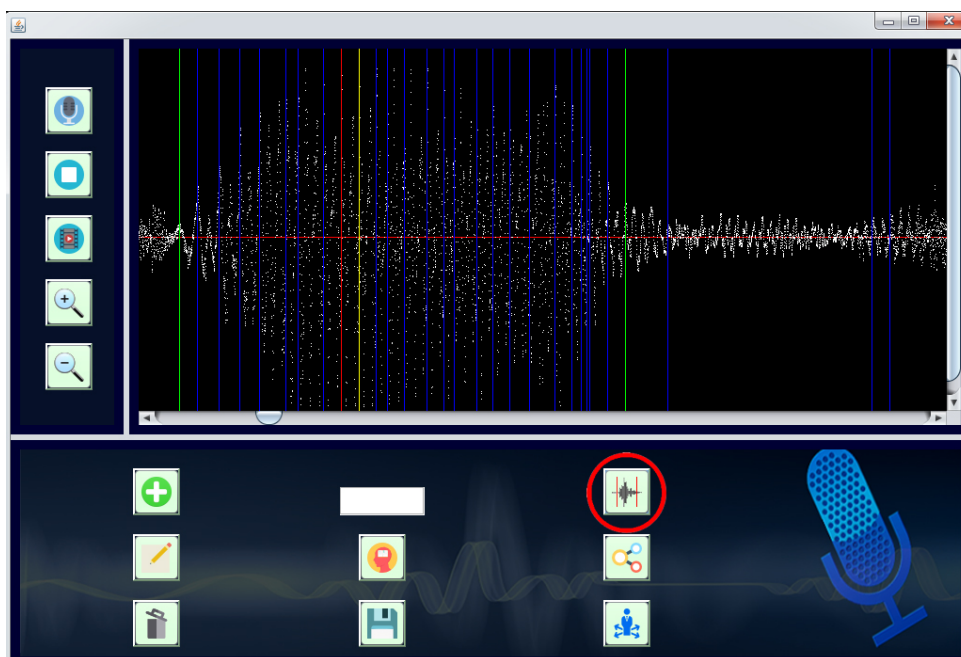


Figure 4.3: Segmentation de parole .

## 4.5 Test et bilan

Afin d'essayer d'estimer le degré de performance de notre programme, Nous l'avons testé sur deux phrases :

- كتب عمر الدرس
- الجهل ظلام

Après la segmentation, nous avons obtenu les résultats suivants :

- 9 syllabes dans la première phrase  
ك ت ب ع م ر ا د ا ر س
- 5 syllabes dans la deuxième phrase  
ا ل ج ه ل ظ ل ا م

Voir les figures 4.4 et 4.5 .

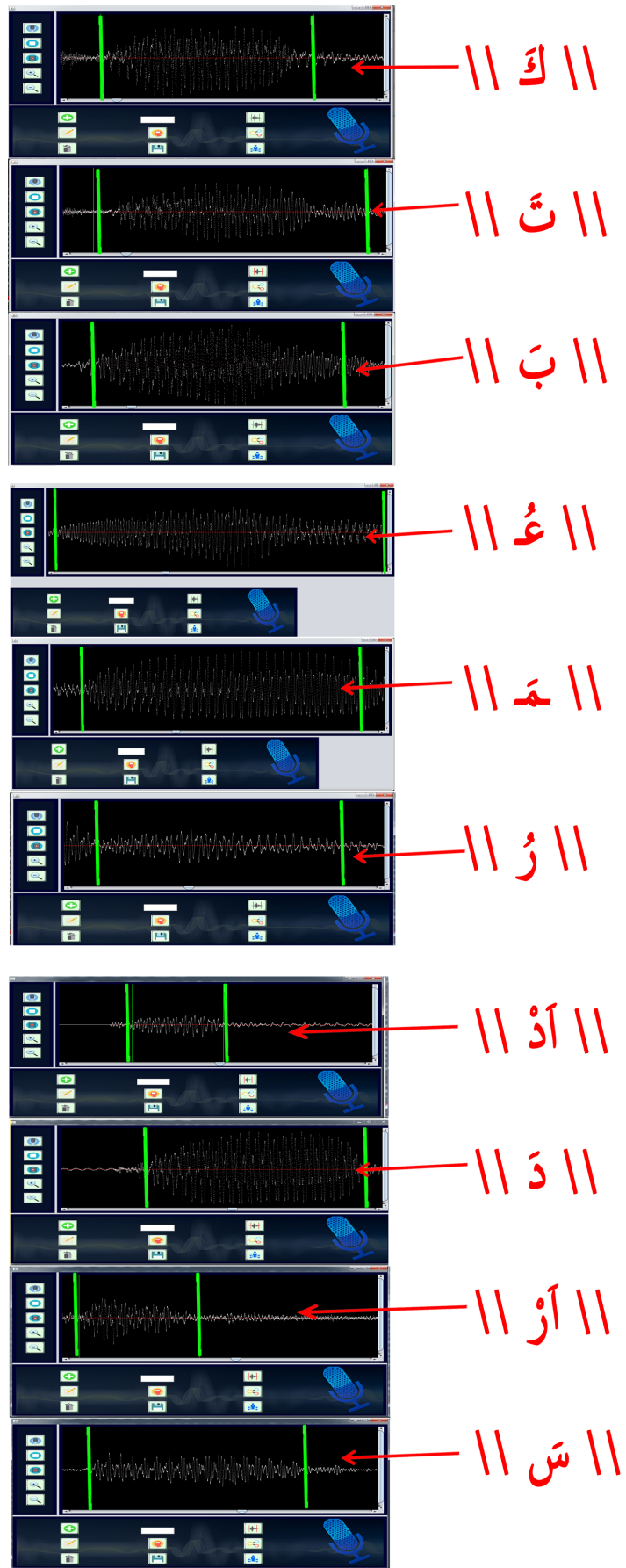


Figure 4.4: Exemple de segmentation (1) .

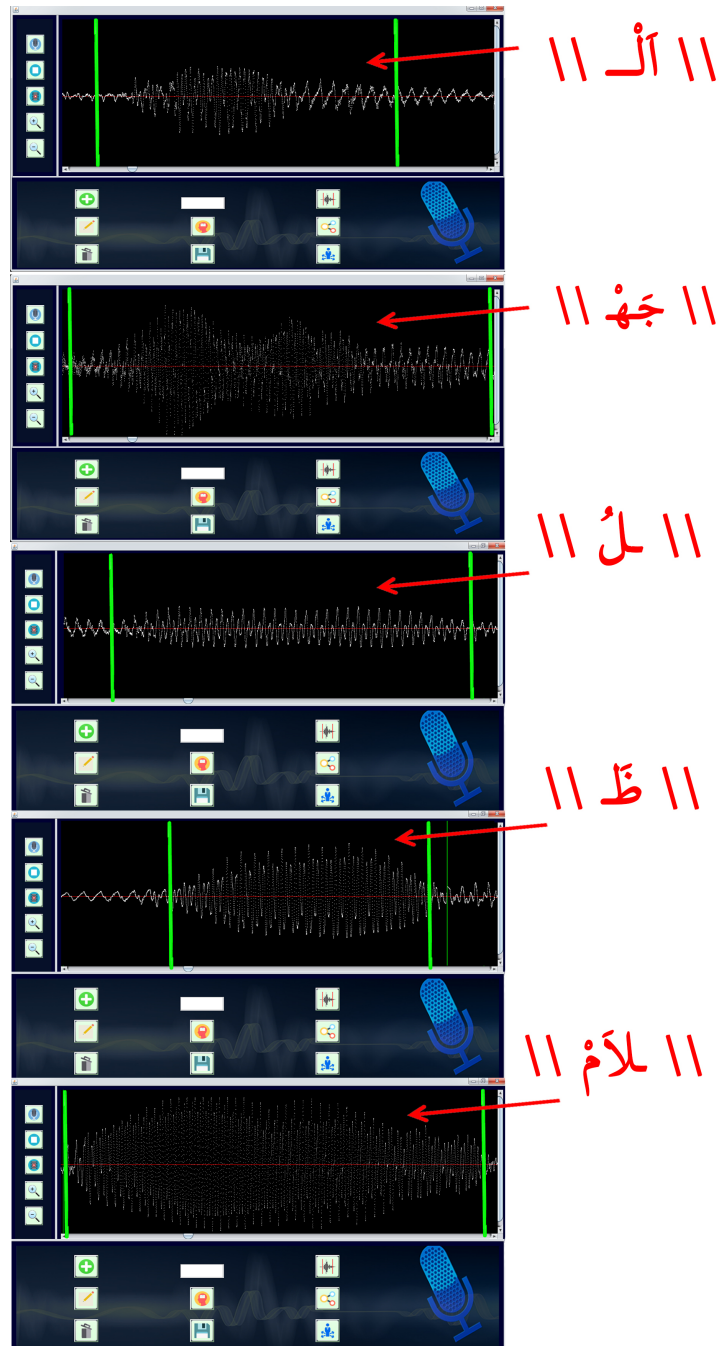


Figure 4.5: Exemple de segmentation (2) .

Maintenant, nous allons essayer d'appliquer cette méthode sur un ensemble de test dont on a pris premièrement les mêmes que les phrases précédentes prononcé plusieurs fois par quatre personnes de différent âge et sexe. On a évalué le degré de segmentation de chaque énoncé à partir de nombre des syllabes correctes que nous avons obtenus. . Enfin, on a calculé le taux de segmentation (TS) correcte de chaque personne. Le tableau suivant resume des résultats :

Tableau 4.1: Résultats de segmentation.

Nbr test personne	كتب عمر الدرس			الجهل ظلام			TS
Homme adulte	7	9	8	5	5	5	0.92%
Femme âgée	8	8	7	5	5	5	0.96%
Petit garçon	8	9	7	5	5	5	0.92%
Petite fille	7	8	9	5	5	5	0.92%

## 4.6 Comparaison des résultats

Nous allons maintenant essayer de comparer les résultats obtenus avec des travaux antérieurs :

- Le travail de **Settou et Maamra** qui se base sur la longueur de segments(parole/non parole) .
- Le travail de **Bekkouche et Sayah** qui se base sur le rapport d'énergie.

### 4.6.1 Première différence :

Elle concerne la méthode de segmentation,de sorte que les autres méthodes segmentent en mots. Voir les deux figures suivantes 4.7 et 4.6 . qui représentent les résultats des méthodes antérieures.

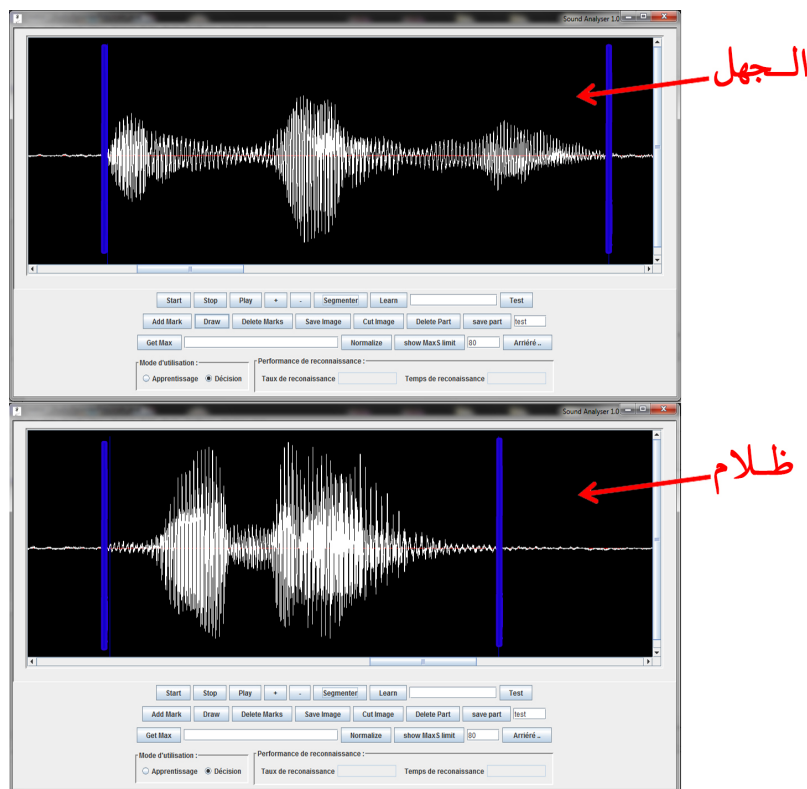


Figure 4.6: la comparaison de segmentation (2) .

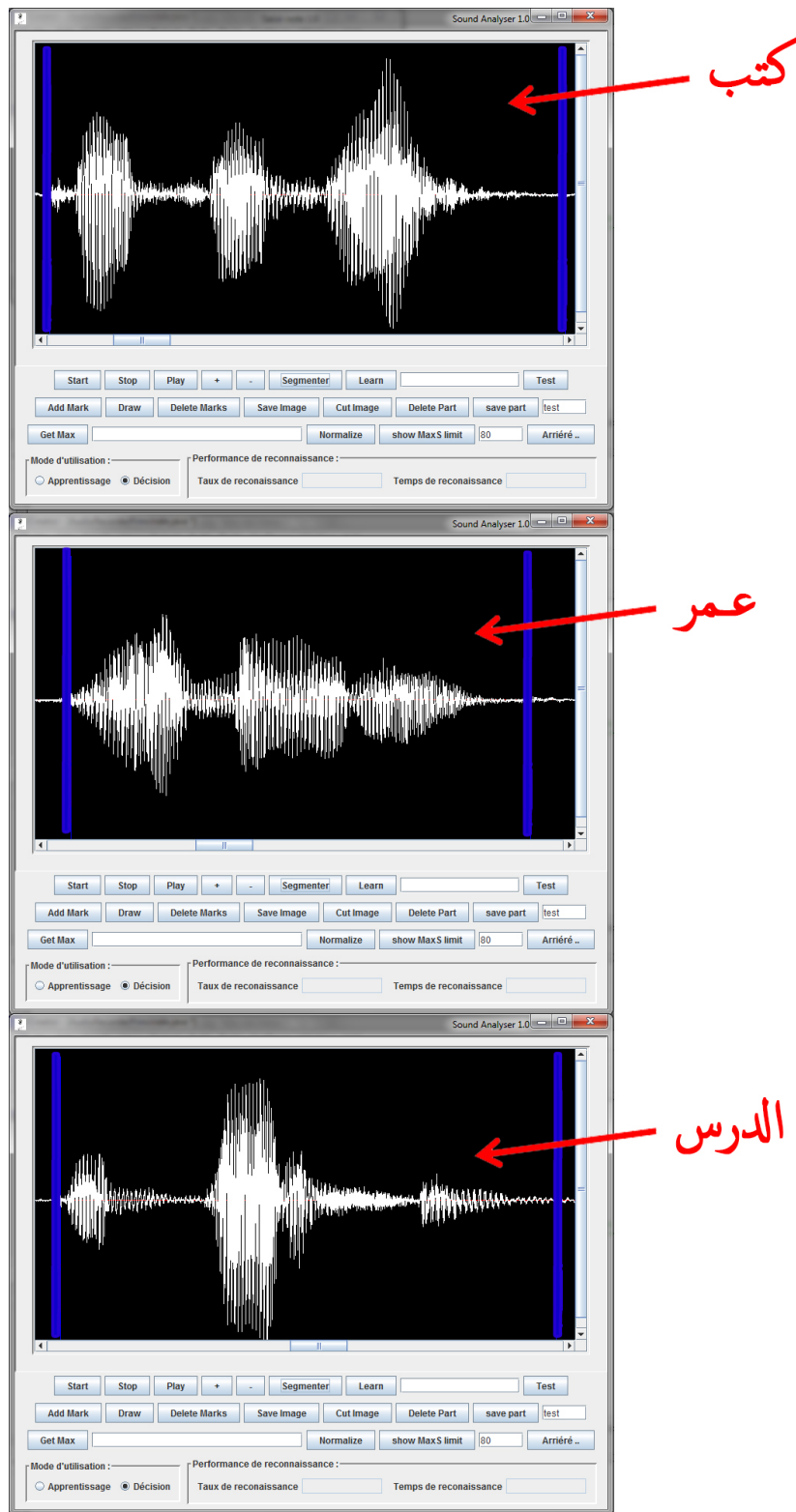


Figure 4.7: Comparaison de segmentation (1) .

#### 4.6.2 Seconde différence :

c'est l'amélioration de temps de calcul, Cela est dû à ce que les méthodes antérieures sont doivent passer sur chaque valeur du tableau pour fair leurs traitements, tandis que ,La méthode que nous avons utilisée se base uniquement sur la projection sur les valeurs sélectionnées du tableau.

**Note :** la méthode (P/NP) : qui se base sur la longueur de segments (parole/non parole).

Tableau 4.2: Comparison du temps de calcul .

le temps de calcul	Nbr test	كتب عمر الدرس			الجهل ظلام		
Méthode (P/NP)		4796.32	5116.83	5340.45	2902.55	3182.18	3112.69
Méthode de rapport d'énergie		5426.22	5226.33	5322.30	2986.34	3120.67	3125.43
Notre méthode		31.80	9.23	8.76	16.66	13.25	10.27

#### 4.6.3 Troisième différence :

L'objectif principal de notre méthode n'est pas de réduire significativement le temps de calcul ,mais, aussi de réduire la taille de la base de données (vocabulaire) dans la phase d'apprentissage lors de la création d'une application **SRAP** complète, et par conséquent, de réduire la taille de mémoire nécessaire.

## 4.7 Conclusion

Nous avons présenté dans ce chapitre l'interface de notre système avec les différentes fonctionnalités, ainsi que les résultats de notre méthode proposée. En plus, nous avons comparé ces résultats avec les résultats de quelques travaux antérieurs.

Les résultats obtenus sont très satisfaisants et ouvrent d'autres perspectives dans le domaine de la reconnaissance de la parole.

# Conclusion générale

Le travail que nous avons fait au long de cette étude, consiste à améliorer la segmentation de signal de la parole arabe en proposant une nouvelle méthode de segmentation à base de zoom-vue.

Après avoir être acquis, le signal de la parole, n'est représenté que par une séquence de valeurs numériques, et par conséquent, ne contient pas d'informations sur l'ordre ni limites des différents mots ou caractères. ces années laisse la segmentation du signal de la parole comme un grand déficit jusque nos jours, malgré les différentes techniques ou méthodes proposées

Nous avons dégagé de cette étude une contribution qui permet de profiter des formes répétées du signal afin de surmonter le problème de segmentation du signal de la parole arabe. Cette méthode permet de résoudre considérablement le problème indiqué.

Le travail réalisé dans ce mémoire, constitue un apport pour la segmentation du signal de la parole arabe, cependant nous voyons qu'il est possible de l'améliorer de l'étendre par les points suivants :

- étendre le principe de l'algorithme à la segmentation en caractères.
- proposer un modèle de segmentation synchrone à la prononciation.



# Bibliographie

- [1] *La carte son* – Ecole Militaire des Nouvelles Technologies.
- [2] J. ALLEGRE – *Approche de la reconnaissance automatique de la parole*, vol. 23, 2003.
- [3] J. AZÉ – « K -plus proches voisins », Thèse, 2007.
- [4] S. Z. BAKKOUCHE DHIYA ELHAK – « Proposition d'une méthode de segmentation adaptative de la parole arabe », **71** (2016).
- [5] F. BENAÏSSA – « Modèles de markov caches appliqués à la reconnaissance automatique de la parole », Thèse, Université Mohamed Boudiaf des sciences et de la technologie d'Oran, 1999.
- [6] R. BENAMMAR – « Traitement automatique de la parole arabe par les hmms : Calculatrice vocale », (2012).
- [7] D. BENOIT – « Les microphones », (1997).
- [8] G. BONNET et G. CAMUS – « Audition humaine ».
- [9] R. G. S. T. DAMIEN CRAM, MADETH MAY – « Réseaux bayésiens », 2005.
- [10] M. B. G. IMANE – « Proposition d'un modèle de classificateur logique, application à la reconnaissance du texte arabe imprimé. », Thèse, 2014.
- [11] INHO KIM – « Automatic pitch detection and shifting of musical tones in real time », Thèse, 2013.
- [12] A. LOTFI – « Un système hybride ag/pmc pour la reconnaissance de la parole arabe. », (2005).
- [13] S. T. MAAMRA OUMÉLHANA – « Proposition d'un modèle de descripteur structurel pour la voix arabe, application saisie des notes », vol. 73, 2015.
- [14] M. DE MARKOV CACHÉS et A. À LA – « Memoire de magister ».
- [15] D. C. MATHILDE GLÉNAT – « Principe du passage de l'analogique au numérique », (2012).
- [16] S. NEFTI – « Segmentation automatique de parole en phones. correction d'étiquetage par l'introduction de mesures de confiance », Thèse, Université Rennes 1, 2004.
- [17] M. D. OUALID – *Reconnaissance automatique de la parole arabe par cmu sphinx 4*, 2013.
- [18] J.-F. PILLOU – « Carte son », (2015).
- [19] T. SAIDANE, M. ZRIGUI et M. BEN AHMED – « La transcription orthographique-phonétique de la langue arabe », *Récital, Fès* (2004).
- [20] H. SATORI, M. HARTI et N. CHENFOUR – « Système de reconnaissance automatique de l'arabe basé sur cmusphinx », *Corpus* **22** (2007), p. 25.
- [21] L. A. SIMARD – « Les microphones », (2014).
- [22] A. ZOUAGHI, M. ZRIGUI et G. ANTONIADIS – « Compréhension automatique de la parole arabe spontanée », *Traitement Automatique des Langues* **49** (2008), no. 1, p. 141–166.