الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR

ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMMED SEDDIK BENYAHIA JIJEL

Faculté des Sciences et de la Technologie

Département d'Electronique

N° :……/2022

MEMOIRE DE MASTER

DOMAINE: Sciences et Technologies

FILIERE: Télécommunications

SPECIALITE : Systèmes des Télécommunications

Thème

Utilisation des réseaux de neurones artificiels pour le rehaussement de la parole

Présenté Par : Abdellatif LABRECHE **Encadré Par : Dr. Chabane BOUBAKIR**

Ouanis BOULAROUK

Date de soutenance : 12/07/2022

Jury de Soutenance

Président :	**M. Abdellah KACHA**	**Grade Professeur**	**Univ MSBY jijel**
Encadreur :	**M. Chabane BOUBAKIR**	**Grade MCB**	**Univ MSBY jijel**
Examinateur :	**M. Ammar SOUKKOU**	**Grade MCA**	**Univ MSBY jijel**

Promotion : 2021 /2022

Remerciements

Tout d'abord, nous remercions DIEU le tout puissant qui nous a donné la force, la volonté et la patience pour réaliser ce modeste travail.

A nos parents qui par leurs prières et leur encouragement, on a pu surmonter tous les obstacles.

Nous tenons à exprimer nos sincères et nos vifs remerciements à notre promoteur ***Dr Boubakir Chaâbane*** *pour tous les précieux conseils qu'il nous a donnés, pour la confiance, et son soutien continu.*

Nous souhaitons également remercier tous les membres de jury, pour le grand honneur qu'ils nous font en acceptant de juger ce travail.

Sommaire

Chapitre I : Réseaux de neurones artificiels : principes et fonctionnement

Chapitre II : Soustraction spectrale pour le rehaussement de la parole

Chapitre III : Notions théoriques sur les techniques CNN et NMF

Chapitre IV : Implémentations, tests et résultats

Liste des figures

Liste des tableaux

Liste des abréviations

ANN	Artificial Neural Network
ART	Adaptive Resonance Theory
ART-1	Adaptive Resonance Theory 1
ART-2	Adaptive Resonance Theory 2
ARTMap	Adaptive Resonance Theory map field
BN	Batch Normalization
CNN	Convolutional Neural Network
DFT	Discrete Fourier Transform.
DNN	Deep Neural Network
DSP	Densité Spectrale de Puissance
EUC	**E**uclidienne
FC	Fully Connected
FFT	Fast Fourier Transform.
IFFT	Inverse Fast Fourier Transform.
IS	Itakura-Saito
KL	Kullback-Leibler
LDR	Linear Dimensionality Reduction
LLR	Log Likelihood Ratio
LVQ	Learning Vector Quantization
MSE	Mean Square Error
MU	Multiplicative Updates
NMF	Non Negative Matrix Factorization

PESQ Perceptual Evaluation of Speech Quality

PMC Perceptron Multi-Couche

RBF Radial Basis Function

ReLU Rectifier Linear Unit

RNA Réseau de Neurone Artificiel

RNR Réseau de Neurone Récurrent

RSB Rapport Signal sur Bruit

RSBseg Rapport Signal sur Bruit Segmentale

SGM Split Gradient Methods

SNR Signal to Noise Ratio

SOM Self-Organizing Map

STFT Short Time Fourier Transform

STSA Short Time Spectral Amplitude

TFCT Transformée de Fourier à Court Terme

TFD Transformée de Fourier Discrète

TFDI Transformée de Fourier Discrète Inverse

VQ Vector Quantization

WSS Weighted Spectral Slope

SDR Source to Distorsion Ratio

SIR Source to Interference Ratio

Introduction générale

Introduction générale

Le rehaussement de la parole est un champ de recherche très actif qui s'est largement développé durant ces dernières décennies. Sa nécessité s'est manifestée en particulier dans plusieurs situations, notamment les appareils auditifs, la reconnaissance de la parole/du locuteur et la communication vocale par téléphone et Internet, dans lesquelles le signal vocal communiqué ou enregistré dans un environnement réel est généralement accompagné par le bruit.

Le débruitage de la parole a pour objectif d'améliorer la qualité et l'intelligibilité de la parole et par conséquent améliorer les performances des applications en relation. Les techniques de débruitage sont très nombreuses et plusieurs approches ont été proposées dans la littérature.

En général, les méthodes de rehaussement de la parole peuvent être classées en deux grandes catégories : non supervisées et supervisées. Les méthodes non supervisées comprennent un large éventail d'approches telles que la soustraction spectrale, le filtrage de Wiener et de Kalman, les estimateurs d'amplitude spectrale à court terme (STSA), etc. Dans ces méthodes, un modèle statistique est utilisé pour les signaux de parole et de bruit, et le signal de la parole propre est estimé à partir de l'observation bruitée sans aucune information préalable sur le type de bruit ou l'identité du locuteur. Cependant, la principale difficulté de la plupart de ces méthodes est l'estimation de la densité spectrale de puissance du bruit, ce qui est une tâche difficile si le bruit additif est non stationnaire.

Pour les méthodes supervisées, un modèle est considéré pour les signaux de parole et de bruit et les paramètres du modèle seront estimés en utilisant les échantillons d'apprentissage de ce signal. Ensuite, un modèle d'interaction est défini en combinant les modèles de la parole et du bruit et la tâche de réduction du bruit est exécutée. Parmi les exemples de cette classe d'algorithmes, citons les approches basées sur les réseaux de neurones, les approches basées sur la factorisation de la matrice non négative (NMF), les approches basées sur le deep learning et la machine learning, etc. L'un des avantages de ces méthodes est qu'il n'est pas nécessaire d'estimer la densité spectrale de puissance (DSP) du bruit à l'aide d'un algorithme séparé.

Ce travail présente une vue d'ensemble de certaines méthodes de rehaussement de la parole en situation monovoie, où le bruit est supposé additif et non corrélé au signal de parole propre. Nous étudierons quelques méthodes supervisées et non supervisées de rehaussement de la parole afin de comparer leurs performances en termes de qualité et d'intelligibilité des signaux rehaussés.

Notre projet de fin d'études s'articule autour de l'utilisation des réseaux de neurones pour le rehaussement de la parole et il est réparti en quatre chapitres :

Au cours du premier chapitre, nous allons détailler le domaine des réseaux de neurones, en présentant les différents aspects ainsi que les différentes structures et algorithmes d'apprentissage.

Le deuxième chapitre présentera un ensemble de préliminaires nécessaires pour aborder les problématiques de débruitage, des algorithmes d'estimation du bruit et d'évaluation de la qualité. Ainsi, une méthode de la catégorie des algorithmes non supervisés de rehaussement de la parole sera détaillée et implémentée, il s'agit de la méthode de la soustraction spectrale de puissance.

Dans le troisième chapitre, nous nous intéresserons particulièrement aux méthodes de rehaussement supervisées comme le réseau de neurone covolutionnel (CNN) et la méthode de factorisation matricielle non négative (NMF). Nous y exposerons également leurs principes de fonctionnement, leurs méthodologies d'implémentation, et leurs phases d'apprentissage.

Le quatrième chapitre sera consacré à la présentation des résultats de simulation et de test, les comparaisons et les discussions.

Enfin, nous clôturons ce mémoire par une conclusion générale en faisant ressortir les éventuelles perspectives pour des travaux futurs.

Chapitre I

Réseaux de neurones artificiels : principes et fonctionnement

I.1 Introduction

Le cerveau est un système complexe qui effectue un travail énorme au niveau du corps humain, il est responsable de toutes les communications biologiques grâce à de nombreuse cellules appelées neurones. Leur traitement parallèle de l'information a incité les chercheures scientifiques à modéliser un système mathématique neuronal qui permet l'utilisation des techniques de ces neurones dans divers domaines, ce système est nommé le réseau de neurone.

Les réseaux de neurones sont en passe de devenir un élément incontournable de l'aide à la décision, ils sont arrivés aujourd'hui à un degré de maturité et de flexibilité leur permettant de traiter un bon nombre de problèmes de grande complexité telles que la reconnaissance de formes, le traitement du signal, l'apprentissage, etc.

Au cours de ce chapitre, nous introduirons l'approche neuronal en détails (définition, les différents architectures, domaine d'application, …).

I.2 Historique

Ceci est un bref historique du développement des réseaux de neurones artificiels en mettant en évidence certains moments et événements clés.

En 1943, W. McCulloch et W. Pitts ont présenté une approche mathématique du neurone biologique appelé le neurone formel. Ce neurone de McCulloch et Pitts a des capacités très limitées et n'a aucun mécanisme d'apprentissage. Pourtant, cela jettera les fondations des réseaux de neurones artificiels et d'apprentissage d'aujourd'hui [3]. Dans cette présentation, ils ont voulu prouver que le fonctionnement du cerveau humain durant la prise d'une décision est équivalent à la machine de Turing et que la pensée est une procédure purement logique.

En 1957, F. Rosenblatt dans son article "Le Perceptron : un automate de perception et de reconnaissance", montre le nouveau modèle du neurone de McCulloch-Pitts - "Perceptron" qui avait de véritables capacités d'apprentissage pour effectuer lui-même une classification binaire.

En 1960, La toute première version de la rétropropagation « Back-propagation » a été montrée par H. J. Kelley dans son article "Gradient Theory of Optimal Flight Paths". Ce modèle jette les bases d'un raffinement supplémentaire du modèle et serait utilisé dans les réseaux de neurones artificielles (ANN) dans les années à venir.

En 1965, La naissance du réseau de neurones multicouches par A. G. Ivakhnenko et V. Grigor′evich Lapa, ils ont présenté un réseau neuronal qui utilise la fonction d'activation polynomiale.

En 1969, M. Minsky et S. Papert publient le livre "Perceptrons" dans lequel ils montrent quelques limitations du perceptron de Rosenblatt à résoudre certains problèmes. Ce revers déclenche l'hiver de la recherche sur les réseaux de neurones.

En 1982, J. Hopfield a créé Hopfield Network, qui est le premier réseau neuronal récurrent (RNR) déterminant pour d'autres modèles de l'ère moderne de l'apprentissage en profondeur.

Plus tard, la même année, P. Werbos, basé sur sa thèse de doctorat de 1974, propose publiquement l'utilisation de la rétropropagation « Back-propagation » pour propager les erreurs lors de la formation des réseaux de neurones.

I.3 Principe et fonctionnement des neurones

I.3.1 Neurone biologique

Un neurone est une cellule nerveuse constituant la base du système nerveux humain contient un grand nombre de neurones fortement interconnectés constituant des réseaux de neurones.

Un neurone comprend un corps cellulaire ou cellule somatique, centre de contrôle de celui-ci, qui fait la somme des informations qui lui arrive. Il traite ensuite l'information et renvoie le résultat sous forme de signaux électriques du corps cellulaire vers l'entrée des autres neurones au travers les axones qui relie les neurones entre eux [4].

Le neurone est également constitué de plusieurs branches nommées dendrites, qui sont les récepteurs principaux du neurone, par lesquelles transite l'information venue de l'extérieur vers le corps cellulaire.

Les synapses du neurone quant à eux reçoivent les informations des autres neurones via l'axone et permettent donc aux neurones de communiquer entre eux. La figure suivante illustre le schéma d'un neurone biologique :

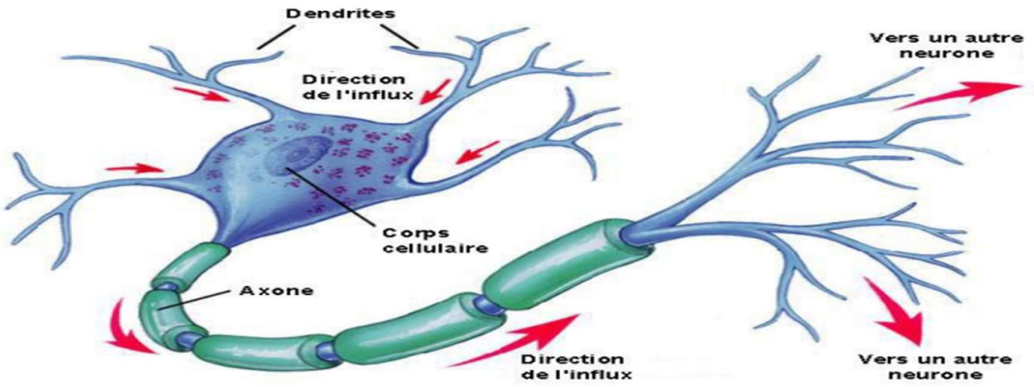

Figure I.1 : Schéma d'un neurone biologique.

I.3.2 Neurone artificiel

Le neurone artificiel est un processeur élémentaire, simulé sur ordinateur ou réalisé sur un circuit intégré. Il reçoit un nombre variable d'entrées en provenance de neurones appartenant à un niveau situé en amont (neurone source). A chacune des entrées est associée un poids « w » (weight en anglais) représentatif de la force de la connexion. Chaque processeur élémentaire est doté d'une sortie unique, qui se ramifie ensuite pour alimenter un nombre variable de neurones appartenant à un niveau situé en aval (neurone destination) [5]. Donc chaque neurone calcule une sortie unique en se basant sur les informations qui lui sont données. La figure (I.2) montre le schéma d'un neurone artificiel :

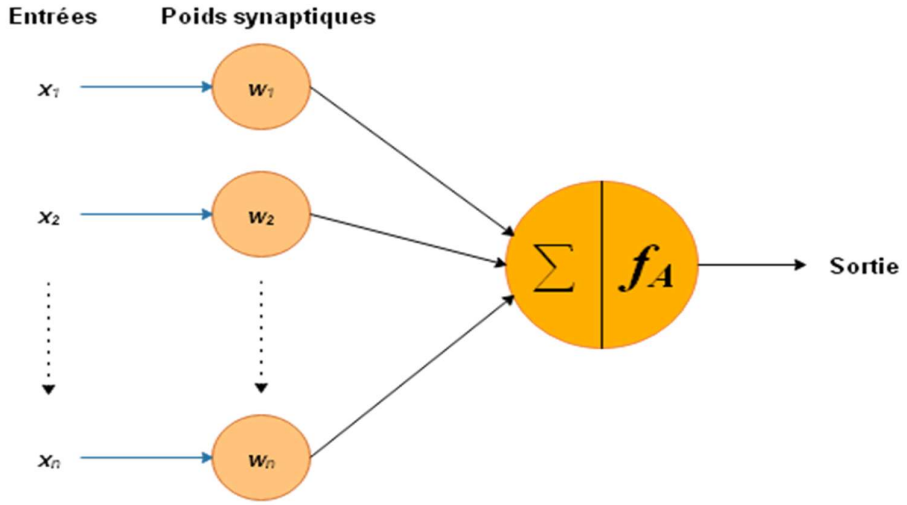

Figure I.2 : Schéma d'un neurone artificiel.

I.3.3 Transition d'un neurone biologique vers neurone artificiel

Les réseaux de neurones biologiques qui constituent le cerveau humain réalisent simplement de nombreuses applications telles que la reconnaissance de formes, le traitement de signal, la mémorisation, la généralisation, l'apprentissage, etc.

Les réseaux de neurones artificiels sont un moyen de modéliser le mécanisme d'apprentissage et de traitement de l'information qui se produit dans le cerveau humain.

Cette inspiration à partir du modèle biologique provient du fait que le cerveau humain est un système apprenant basé sur une structure contenant environ 100 milliards de neurones reliés entre eux par 10000 contacts synaptiques ce qui représente un million de milliards de synapses.

On pourra décrire dans le tableau (I.1) la transition entre le neurone biologique et le neurone artificiel [6] :

Tableau I.1 : Transition du neurone biologique au neurone formel.

Neurone biologique	Neurone artificiel
Synapses	Poids de connexion
Axones	Signal de sortie
Dendrite	Signal d'entrée
Cellule somatique	Fonction d'activation

I.3.3.1 Neurone formel

Créé par W. McCulloch et W. Pitts, C'est l'approximation la plus précise du fonctionnement du neurone biologique, utilisant une fonction binaire, il est caractérisé par [7] :

- Chaque nœud a plusieurs entrées.
- Les entrées proviennent d'autres neurones.
- Les entrées sont pondérées.
- Les poids sont soit positifs soit négatifs.
- Les entrées sont sommées au niveau du nœud pour produire une valeur d'activation.
- Le neurone s'active si l'activation est plus grande qu'un certain seuil.

I.3.3.2 Les propriétés mathématiques d'un neurone formel

Le neurone formel réalise les fonctions mathématiques suivantes :

$$S = f_A(P) \tag{I.1}$$

P = Somme pondérée (somme des produits d'entrées et des poids)

$$P = \sum_{i=1}^{n} w_i x_i \tag{I.2}$$

$x_i = x_1 \dots x_n$ = Entrées ; $w_i = w_1 \dots w_n$ = Poids synaptiques

$f_A =$ Une fonction d'activation ;

On peut la réécrire $S = g(x_i)$ ou $g(x_i) = f_A(\sum_{i=1}^{n} w_i x_i)$. Le processus est paramétré par les w_i.

La figure suivante présente la forme d'un neurone de McCulloch et Pitts :

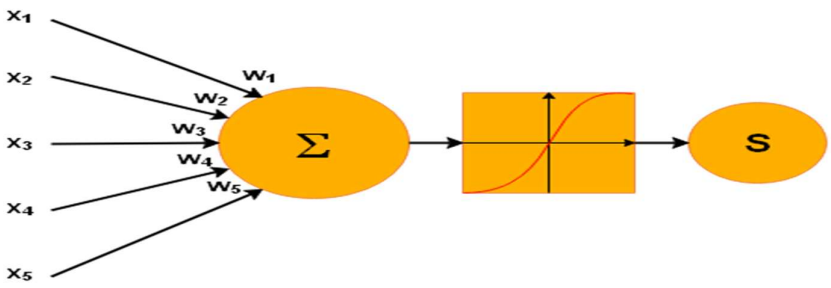

Figure I.3 : Neurone de McCulloch et Pitts.

I.3.4 Fonction d'activation

Une fonction d'activation est une fonction mathématique appliquée au signal de sortie d'un neurone. Le seuil de stimulation, qui, une fois atteint, déclenche une réponse neuronale.

La fonction d'activation est généralement une fonction non linéaire. Un exemple de fonction d'activation est la fonction Heaviside, qui renvoie toujours 1 si le signal d'entrée est positif et 0 si le signal d'entrée est négatif.

Il existe plusieurs types de fonctions d'activation dont les sorties sont soit linéaires soit non linéaires. Les fonctions d'activation usuelles sont :

- **Fonction Heaviside**

$$f(x) = \begin{cases} 1, & si\ x \geq 0 \\ 0, & sinon \end{cases} \quad \text{(I.3)}$$

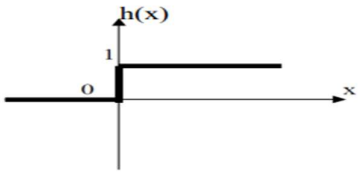

Figure I.4 : Fonction Heaviside.

- **Fonction Signe**

$$Sgn(x) = \begin{cases} +1, & si\ x \geq 0 \\ -1, & sinon \end{cases} \quad \text{(I.4)}$$

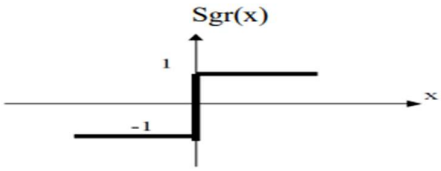

Figure I.5 : Fonction Signe.

- **Fonction linéaire ou identité**

Si la fonction identité a l'avantage d'être simple, il n'a que peu de rapport avec la réalité (le signal de sortie est non borné, linéaire par rapport aux signaux d'entrées, ce qui ne correspond pas du tout au fonctionnement des neurones biologiques). Mais elle est généralement utilisée dans la couche d'entré du réseau [6].

$$f(x) = x \quad \text{(I.5)}$$

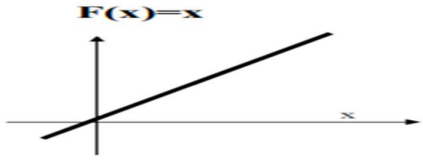

Figure I.6 : Fonction linéaire.

- **Fonction linéaire a seuil ou multi-seuils**

Cette fonction représente un compromis entre la fonction linéaire et la fonction a seuil : entre ses deux barres de saturation, elle confère au neurone une gamme de réponses possibles. En modulant la pente de la linéarité, on affecte la plage de réponse du neurone [6].

$$f(x) = \begin{cases} x, & si\ x \in [u, v] \\ v, & si\ x \geq v \\ u, & si\ x \leq u \end{cases} \quad \text{(I.6)}$$

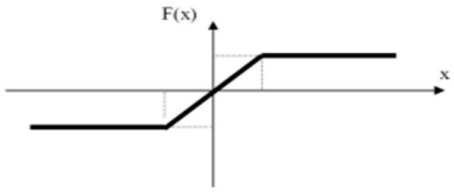

Figure I.7 : Fonction linéaire a seuil.

- **Fonction Sigmoïde**

$$f(x) = \frac{1}{1 + e^{-x}} \quad \text{(I.7)}$$

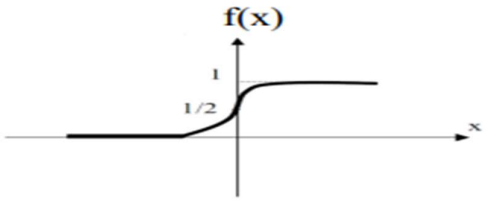

Figure I.8 : Fonction sigmoïde.

I.4 Différentes structures de réseaux de neurones artificiels

L'évolution des poids et biais durant l'apprentissage d'un réseau de neurones est effectué par le format des données entrées. Pour cela on distingue deux types de réseaux :

- Les réseaux de neurones statiques (ou acycliques, ou non bouclés).
- Les réseaux de neurones dynamiques (ou récurrentes, ou bouclés).

Un réseau dit statique est un réseau qui ne contient pas de connexion arrière (feedback or delay). Par conséquent, on peut lui présenter les données en entrée dans n'importe quel ordre, cela n'influencera pas l'évolution de ses poids lors de la phase d'apprentissage. Il est alors préférable de lui donner tout le jeu de donnée en un seul coup lors de la phase d'apprentissage. On parle alors d'apprentissage par paquet (« batch training »). Les réseaux « feedforward » ne peuvent pas simuler des processus dépendant du temps. Par contre, si l'on veut simuler un processus qui dépend du temps, alors on pourra utiliser un réseau de neurones contenant des connections arrières « feed-back ». L'ordre de présentation du jeu de données au réseau de neurone sera alors primordial. On parle alors d'apprentissage séquentiel [5].

1.4.1 Réseaux feed-forward

Appelés aussi « réseau de type Perceptron », ce sont des réseaux dans lequel l'information entrée se propage de couche en couche sans retour en arrière possible.

1.4.1.1 Perceptrons

a. Perceptron monocouche

C'est le neurone artificiel le plus simple (monocouche), qui effectue des calculs afin de détecter des caractéristiques dans les données d'entrées. Il s'agit d'un algorithme pour l'apprentissage supervisé de classificateur binaire, qui permet aux neurones artificiels d'apprendre et de traiter les éléments d'un ensemble de données [8].

Le perceptron monocouche contient une couche d'entrée et une couche de sortie (figure I.9).

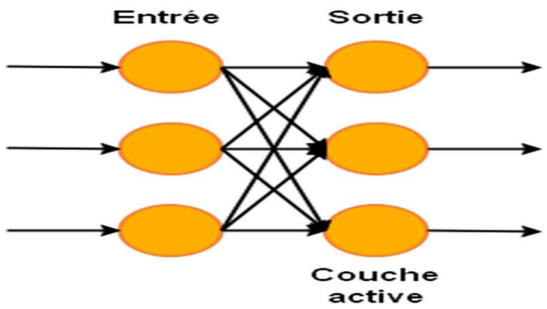

Figure I.9 : Perceptron monocouche.

b. Perceptron multicouche (PMC)

Le perceptron multicouche contient plus de deux couches de base, il comprend une ou plusieurs couches intermédiaires appelés les couches cachées. Ses couches effectuent des calculs mathématiques sur nos entrées. L'un des défis de la création de réseaux de neurones consiste à décider du nombre de couches cachées, ainsi que du nombre de neurones pour chaque couche [8].

Il peut résoudre des problèmes logiques plus compliqués. Avec l'arrivée des algorithmes de rétro propagation, ils deviennent le type de réseaux de neurones le plus utilisé [9]. La figure suivante présente un exemple de perceptron multicouche :

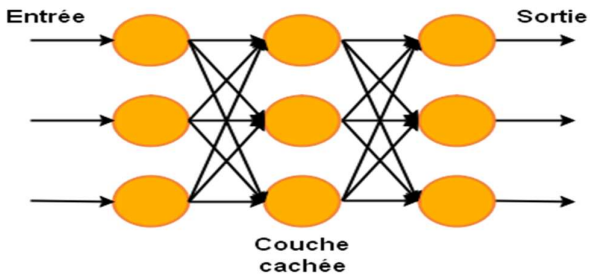

Figure I.10 : Perceptron multicouches.

1.4.1.2 Réseaux à fonction radiale (RBF)

Le réseau RBF est un réseau de neurones supervisé. Il s'agit d'une spécialisation d'un PMC. Un RBF est constitué uniquement de 3 couches [9].

- La couche d'entrée : elle retransmet les entrées sans distorsion.
- La couche RBF : couche cachée qui contient les neurones RBF.
- La couche de sortie : simple couche qui contient une fonction linéaire.

1.4.2 Réseaux feed-back

Ce sont des réseaux qui permettent des connexions arbitraires entre les neurones de toutes les couches ; lorsqu'on se déplace dans le réseau en suivant le sens des connexions, il est possible de trouver au moins un chemin qui revient à son point de départ. Ils constituent la deuxième grande classe de réseaux de neurones artificiels (RNA) [10].

1.4.2.1 Cartes de Kohonen

La carte de Kohonen est une grille de neurones relié entre eux. Chaque neurone du réseau est relié à un neurone de la carte de Kohonen.

Un modèle de neurone réaliste plus proche. Ces réseaux ont été inspirés par des observations biologiques du fonctionnement du système nerveux sensoriel des mammifères. Une loi de Hebb modifiée est utilisée pour l'apprentissage. Là où les neurones connectés sont actifs en même temps, la connexion est renforcée et l'inverse est réduit (alors qu'il ne se passait rien avant).

Utilisation : classification, le traitement d'image, l'aide à la décision et l'optimisation, Data mining. La figure suivante présente un exemple d'une carte de Kohonen.

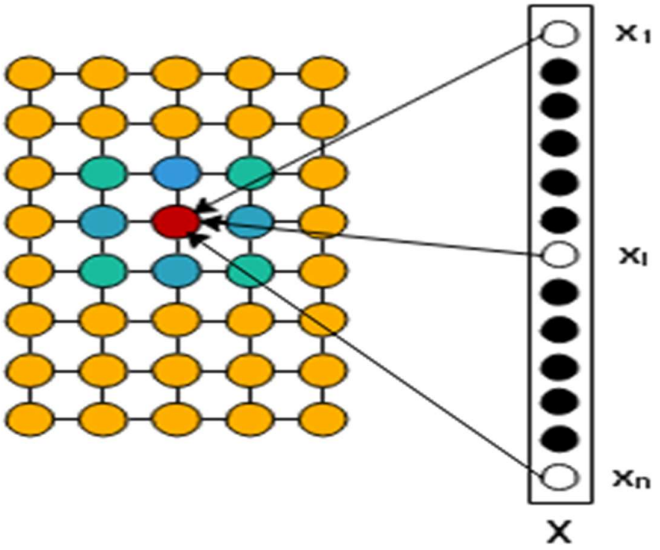

Figure I.11 : Carte de Kohonen.

1.4.2.2 Réseaux de Hopfield

Les réseaux de Hopfield sont des réseaux dont les neurones sont connectés entre eux d'une façon bidirectionnelle, donc il n'y a aucune différence entre les neurones d'entré et les neurones de sortie. Il s'agit d'un système coopératif ou la décision est prise par étapes successives [11]. Ils fonctionnent comme une mémoire associative non-linéaire capable de trouver un objet stocké en fonction de représentations partielles ou bruitées. L'application principale des réseaux de Hopfield est l'entrepôt de connaissances mais aussi la résolution de problèmes d'optimisation.

Utilisation : La reconnaissance de forme, l'entrepôt de connaissances, la résolution de problèmes d'optimisation. La figure (I.12) montre un exemple d'un réseau de Hopfield.

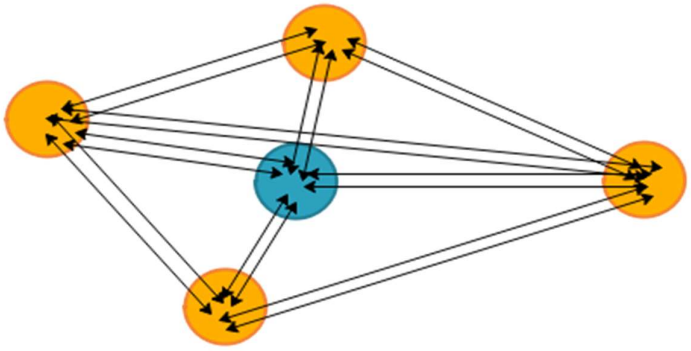

Figure I.12: Réseau de Hopfield.

1.4.2.3 Réseaux de neurones ART

Les réseaux ART (« Adaptive Resonance Theory ») sont des réseaux qui apprennent par la compétition. Il existe deux principaux types de réseaux ART : ART-1 pour les entrées binaires et ART-2 pour les entrées continues.

Utilisation : Catégorisation.

La figure suivante présente un exemple d'un réseau ART-1.

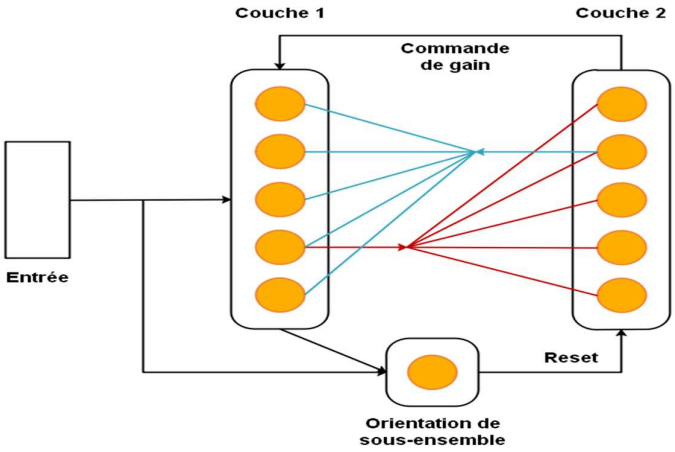

Figure I.13 : Réseau ART-1.

1.4.2.4 Réseaux à compétition

Le réseau à compétition est un réseau monocouche, une couche d'entré et une couche de sortie en compétition. Le fonctionnement de ce type de réseaux et décrit ci-dessous :

« Les données d'entrée sont présentées au réseau, déclenchant différentes réponses dans les neurones de sortie. La concurrence s'installe alors entre ces derniers et prend la forme d'une ruée vers l'influence, qui doit finir par se stabiliser en raison des forces qui inhibent les liens. A l'issue de la compétition, le neurone de sortie le plus actif est déclaré "gagnant" ». La figure (I.14) schématise un réseau à compétition :

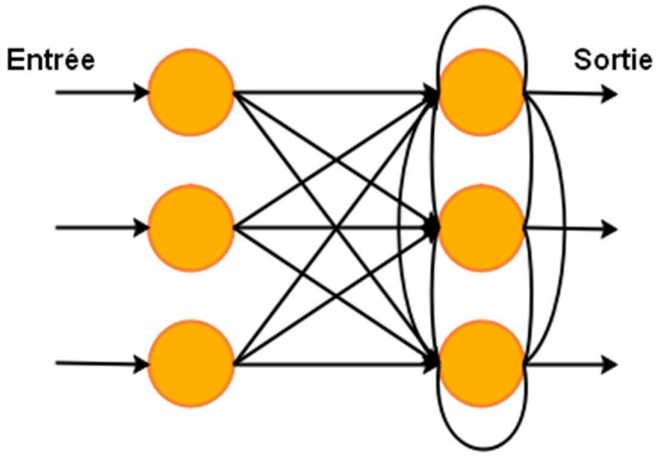

Figure I.14 : Réseau à compétition.

I.4.3 Connexions entre neurones

Les connexions vont relier les neurones entre eux. La structure des connexions peut aller de la connectivité partielle à la connectivité totale. On associe à une connexion entre deux neurones, un poids qui rend compte de l'influence d'un des neurones sur l'autre. La dynamique des états correspond à l'évolution des états des différents neurones d'un réseau. Elle dépend à la fois des fonctions d'activation, de la structure et des poids des connexions. La dynamique des connexions représente le fait que les poids des connexions peuvent être modifiés par une phase appelée l'apprentissage. On peut assimiler cette dynamique à la plasticité synaptique [5].

I.5 Apprentissage

L'apprentissage consiste à adapter la connaissance du RNA au problème posé, La connaissance étant définie par l'ensemble des pondérations sur les liens du réseau ainsi que par sa topologie, l'apprentissage est donc étroitement lié au réglage adéquat des poids d'interconnexion, ainsi qu'à l'adaptation de la topologie [6].

I.5.1 Types d'apprentissage

Il existe trois types d'apprentissage : supervisé, non supervisé et mixte.

I.5.1.1 Supervisé

Dans ce type d'apprentissage, on dispose d'une base de données (appelé ensemble d'apprentissage) qui contient des paires d'entrées et de sorties désirés. A chaque itération, on fournit un exemple d'entrée au réseau, une réponse est attendue en sortie. Le réseau s'adapte par la comparaison entre le résultat obtenu à la sortie et la sortie désiré dans la base de données. Le réseau va corriger les poids des connexions et calculer le nouveau résultat, ainsi de suite jusqu'à ce qu'il trouve la sortie attendue.

I.5.1.2 Non supervisé

Dans le cas où une base de données ou ensemble d'apprentissage n'est pas disponible à cause de manque d'informations sur le domaine ou la taille de données est trop volumineuse pour pouvoir être labelisé a la main, c'est dans ce cas où l'apprentissage non supervisé est utile. Ce type d'apprentissage est le plus proche à l'apprentissage du système biologique.

Le réseau de neurones dans ce cas va s'adapter continuellement en fonction des régularité statistiques en entrée et établira des catégories. Ensuite, l'optimisation d'une valeur de qualité selon les catégories reconnues précédemment.

I.5.1.3 Mixte

Ce type d'apprentissage est une combinaison des autres types cité auparavant. On revient à ce type d'apprentissage lorsque l'ensemble d'apprentissage n'est pas parfait, donc le réseau est besoin d'autre exemples pour bien comprendre le concept. Donc, dans ce cas une partie des poids va être déterminés par apprentissage supervisé et l'autre partie par apprentissage non supervisé.

I.5.2 Algorithme de rétropropagation

L'algorithme de rétropropagation est actuellement l'algorithme le plus utilisé dans l'apprentissage des réseaux de neurones à cause de son efficacité et robustesse en termes de rapidité et allocation de mémoire prouvé dans plusieurs conditions et structures. Son principe est de calculer l'erreur en sortie et la transmise en sens inverse vers l'entrée et exécuter la même procédure jusqu'on obtient l'erreur la plus minimale.

Cet algorithme, nous permet d'obtenir les réponses les plus possiblement correcte. Dans un réseau de neurones, on dispose d'une entrée « p », une sortie désiré « d » et une sortie « a ». On introduit la fonction d'erreur :

$$e = d - a \tag{I.8}$$

Notre objectif est de minimiser la valeur de « e » en modifiant les poids du réseau. Une fois les poids sont établis à partir des ensembles fournis par l'utilisateur, on commence à injecter des entrées inconnues pour tester le rendement de ce réseau.

La performance d'un réseau est caractérisée par l'erreur quadratique moyenne totale E, pour chaque m neurone de sortie, on obtient :

$$E = \frac{1}{N}\sum_{l=1}^{N}\sum_{j=1}^{m} e_{j,l}^2 \tag{I.9}$$

$e_{i,l}$: L'erreur commise sur le j^{eme} neurone de sortie.

I.5.2.1 Algorithme d'apprentissage d'un réseau monocouche

L'expression de l'évolution des poids dans ce cas est très simple et elle est basée sur le calcul du gradient de l'erreur quadratique moyenne totale E par rapport a chaque poid $w_{j,i}$ de la couche de sortie. Ces poids sont modifiés de cette façon :

1. Initialisation des poids a des valeurs aléatoires de faible grandeur.
2. Propagation de l'entrée à travers le réseau et le calcul des sorties y_j.
3. Calcul de l'évolution des poids au cours d'une itération :

$$\Delta w_{j,i} = -2\mu \sum_{i=1}^{N} e_{j,i} f_1'(W^1 \times x_i) x_i \tag{I.10}$$

μ : pas d'apprentissage.

4. Mettre à jour chaque poids synaptique du réseau :

$$w_{j,i}(k+1) = w_{j,i}(k) + \Delta w_{j,i}(k) \tag{I.11}$$

5. Retourner vers l'étape 2 si l'erreur est trop grande.

I.5.2.2 Algorithme d'apprentissage d'un réseau multicouches

Sur un réseau de neurones multicouches, les couches cachées ont une grande influence sur l'évolution des poids à travers le réseau, il diffère de l'algorithme précédent seulement dans la 3eme et la 4eme étape.

3. Pour chaque couche cachée k, on calcule l'évolution du poids au cours d'une itération :

$$\Delta w_{j,i}^k = -2\mu \sum_{i=1}^{N} e_{j,i}^k f_1'(W^k \times y_i^{k-1}) y_i^{k-1} \tag{I.12}$$

4. Mettre à jour chaque poids synaptique du réseau :

$$w_{j,i}(k+1) = w_{j,i}(k) + \Delta w_{j,i}(k) \tag{I.13}$$

I.5.3 Classification de différents apprentissages

Le tableau (I.2) résume les différents algorithmes d'apprentissage avec classification :

Tableau I.2 : Classification de différents apprentissages [6].

Mode	Règle	Architecture	Algorithme	Objectif
Supervisé	Correction d'erreur	Perceptron simple ou multicouche	Rétropropagation, Adaline, Madaline	Classification. Approximation de fonction Prédiction Contrôle
	Bolzmann	Récurrente	Apprentissage de Bolzmann	Classification.
	Hebb	Multicouches no bouclé	Analyse de discriminent linéaire	Analyse de données. Classification.
	Par compétition	A compétition	LVQ^{α}	Catégorisation au sein d'une classe. Compression de données
		ART	ARTMap	Classification. Catégorisation au sein d'une classe.
Non supervisé	Correction d'erreur	Multicouche non bouclés	Projection de Sammon	Analyse de données.
	Hebb	Non bouclé ou à compétition	Analyse en composant principale	Analyse de données. Compression de données.
	Par compétition	A compétition	VQ^{β}	Catégorisation. Compression de données.
		Carte de Kohonnen	SOM^{γ}	Catégorisation. Analyse de données.
		ART	ART-1, ART-2	Catégorisation.
Mixte	Correction d'erreur et par compétition	RBF	RBF	Classification. Approximation de fonction. Prédiction. Contrôle.

I.6 Procédure de développement d'un réseau de neurones

On peut résumer la mise en œuvre d'un réseau de neurones dans les étapes suivantes :

1. **Collection de données (ensemble d'apprentissage) :** cette étape est très importante car son objectif est de rassembler des données suffisante et fiable avec le minimum de bruit pour l'apprentissage et pour le test de notre système.

2. **Analyse de données :** Cette étape permet d'éliminer la redondance de la base de données ce qui nous permettre de minimaliser le temps de simulation et d'apprentissage à travers la réduction de la taille du réseau.
3. **Subdivision de la base de données :** Dans cette étape on va séparer la base de données en deux parties : une pour l'apprentissage et l'autre pour le test. Elle dépend de la quantité de données à notre disposition et le temps disponible pour l'apprentissage.
4. **Choix du type de réseau :** Cette étape dépend de plusieurs critères, comme la nature du problème (Classification, Catégorisation, Compression), la nature de données à analyser, les contraintes temporelles, …
5. **Normalisation de données :** Consiste à faire un prétraitement de données pour déterminer les motifs et les paramètre de notre ensemble ce qui nous permettre de la normaliser.
6. **Choix de l'algorithme d'apprentissage :** Le but derrière ce choix et d'atteindre la performance et la rapidité de convergence optimale. On peut d'abord utiliser un apprentissage structurel ou la comparaison entre différentes architectures testées sur le réseau.
7. **Validation du réseau :** en utilisant la partie de données de test de notre base, on fait l'audit de la performance du système, si on n'est pas satisfait, on va éventuellement soit changer le type de réseau ou la méthode d'apprentissage et de test.

I.7 Domaine d'application

Depuis leur invention, les réseaux de neurones devenue la solution parfaite pour résoudre les problèmes ou un formalisme mathématique exacte n'est pas disponible ou pratiquement impossible de le réaliser en temps réel. Ci-dessus on va vas citer quelques domaines d'application des réseaux de neurones :

- La prévision météologique.
- La reconnaissance de formes ou parole.
- Le pilotage automatique de véhicules
- Le contrôle industriel
- Le traitement du signal
- La robotique

I.8 Avantages et inconvénients

Les réseaux de neurones artificiels ont plusieurs avantages tels que :

- L'accessibilité et la facilité d'implémentation et d'utilisation
- Le parallélisme qui garantit la rapidité d'exécution pour de taux de données importantes.
- L'extensibilité.
- La possibilité de généraliser et distribuer les connaissances acquises.

Malheureusement, les réseaux de neurones présentes des inconvénients que l'on peut citer :

- Le réseau de neurones est une boite noire qui ne justifie pas ses décisions, donc il est impossible d'inspecter le comportement d'un réseau et donc réduire leur interopérabilité.
- La difficulté de définir le type de réseaux et la méthode d'apprentissage.
- Un mauvais choix d'une fonction d'erreur va causer un mauvais apprentissage de réseaux ou conduire à un blocage ou déviation du réseau au loin de la solution désiré.

I.9 Conclusion

Dans ce chapitre, nous avons présenté les réseaux de neurones artificiels basés sur le modèle biologique du cerveau humain. Ainsi que la méthode la plus utilisée pour l'apprentissage des réseaux de neurones statiques, qui est l'algorithme de la rétropropagation. Nous avons cité aussi les architectures de certains types de réseaux de neurones artificiels, ces réseaux sont caractérisés par de fortes capacités d'apprentissage et de généralisation.

Chapitre II

Soustraction spectrale pour le rehaussement de la parole

I.1 Introduction

Le débruitage de la parole est une opération cruciale pour l'amélioration de la qualité et l'intelligibilité de la voix et la réduction de la fatigue de communication dans les systèmes modernes.

Généralement, on distingue deux grandes catégories de méthodes de débruitage de la parole. Une phase dans le domaine temporel et l'autre dans le domaine fréquentiel. Les méthodes fréquentielles les plus utilisées sont la soustraction spectrale et ses variantes et le filtrage de Wiener. Cette étude se base essentiellement sur la méthode de soustraction spectrale.

Celle-ci cherche à estimer le spectre à court-terme du signal propre à partir de l'observation qui est généralement bruitée. Cette technique et ses variantes sont largement connues pour leur simplicité, leur faible complexité et la "bonne" qualité du signal débruité.

Dans ce qui suit, nous allons présenter un état de l'art des méthodes de débruitage de la parole en particulier la méthode de la soustraction spectrale, une version améliorée de cette dernière, ainsi que les méthodes d'estimation du bruit utilisé.

II.2 Divers types de dégradation de la parole

La parole peut être corrompue par un bruit généré par l'environnement à n'importe quelle étape de la chaîne de communication. Les différentes manières dont la parole peut être dégradée sont classées selon les catégories comme suit :

- **A la source de la parole**
  Quand la source elle-même est situé dans un environnement bruité, le bruit s'ajoute au signal de la parole prononcé par un locuteur qui se trouve dans cet environnement.

- **Durant la transmission**
  Le signal de la parole est généralement transmis à un récepteur lointain à travers des canaux de transmission. Durant cette transmission, le bruit additionnel s'ajoute au signal de parole à cause du comportement non idéal du canal. Le bruit peut également être ajouté pendant la conversion de données faite avant la transmission ou durant la reproduction de la parole aux auditeurs.

- **Le bruit à la réception**
  Parfois, bien que la source de la parole puisse être dans un environnement silencieux, le récepteur peut être dans un environnement fortement bruité. Ici également, la fatigue d'écoute surgit pendant que la qualité de la parole est dégradée. Par conséquent, le rehaussement de la parole est nécessaire dans ce cas aussi.

II.3 Dèbruitage de la parole

Les méthodes de débruitage des signaux de parole sont très nombreuses et constituent une grande partie de la littérature lorsqu'il s'agit d'améliorer la qualité ou l'intelligibilité des signaux. Pour ce faire, plusieurs approches de rehaussement de la parole impressionnant ont été développées afin de rendre les techniques de traitement de la parole très actives, fonctionnelles et performantes. Pour que les différentes techniques de traitement de la parole puissent fonctionner correctement, il est important de disposer en entrée d'un signal de parole clair et intelligible.

Dans la pratique, la parole est acquise en présence de bruits, qui viennent dégrader la performance de ces techniques et qui peuvent même les rendre non fonctionnelles. Il est donc nécessaire de rehausser cette parole bruitée.

II.3.1 Intérêt du rehaussement de la parole

Le rehaussement de la parole nous permet de :

- Restituer le signal de parole dans les environnements bruités ;
- Améliorer les performances de la communication vocale dans un milieu bruité ;
- Réduire la fatigue de la parole dans les systèmes modernes ;
- Traiter la parole pour les malentendants ;
- Reconnaitre et authentifier automatiquement la parole.

II.3.2 Classification des techniques de dèbruitage de la parole

Les méthodes de dèbruitage de la parole peuvent être classées de plusieurs façons comme il est montré dans le tableau suivant :

Tableau II.1 : Classification des méthodes de dèbruitage de la parole [12].

Type d'algorithme	Adaptatif / Non adaptatif
Nombres des canaux d'entrée (Nombre de capteurs)	Un / Deux / Multiples
Domaine de traitement	Temporel / Fréquentiel

Dans cette étude, Nous avons choisi de limiter notre étude au cas monovoie car c'est le contexte le plus courant en traitement de la parole, pour le codage ou la reconnaissance de la parole dans un milieu bruité. Dans la catégorie des techniques monovoie, on peut classer les méthodes de rehaussement en trois types : les méthodes basées sur la périodicité de la parole, les méthodes basées sur un modèle de la parole, les méthodes basées sur l'estimation de l'amplitude spectrale à courte terme (STSA). Plusieurs techniques de débruitage emploient cette dernière, parmi lesquelles, on trouve la soustraction spectrale.

II.3.3 Applications :

De façon générale, les techniques de rehaussement de la parole sont exploitées dans les domaines suivants :

- Téléphonie mobile,
- Les systèmes à commande vocale,
- Les terminaux mains-libres,
- La fabrication d'appareils auditifs médicales,
- ...

II.3.4 Modèle d'observation

Le signal de parole obtenu à la sortie d'un microphone est très souvent constitué d'un signal de parole auquel est rajouté un bruit additif :

$$y(n) = s(n) + \mathrm{b(n)} \qquad \text{(II.1)}$$

Où y(n), s(n) et b(n) désignent respectivement le signal observé bruité, le signal propre et le bruit additif.

Le signal de parole peut être considéré comme une réalisation particulière d'un processus aléatoire stationnaire sur des intervalles de temps de courte durée. Ses caractéristiques statistiques peuvent, donc, être estimées sur des intervalles de temps de durée variant de 5 à 30 ms. Ainsi, le traitement d'un tel signal se fait par trames [13].

Dans le but de diviser le signal vocal en trames, on utilise une fenêtre de pondération glissante dans le domaine temporel. La transformée de Fourier à Court Terme (TFCT) de l'observation $Y\ (m,k)$ s'écrit comme suit :

$$Y\ (m,k)\ =\ S(m,k)\ +\ B(m,k) \qquad \text{(II.2)}$$

Où $S(m,k)$ (resp. $B(m,k)$) désigne la TFCT du signal propre (resp. du bruit), m désigne le numéro de la trame et k désigne l'indice fréquentiel.

Très souvent, on suppose que le signal de parole et le bruit sont non corrélés. En effet, ils sont issus de sources différentes. En tenant compte de cette hypothèse, la relation entre les spectres de puissance des signaux s'écrit comme suit :

$$|Y\ (m,k)|^2 = |S(m,k)|^2\ +\ |B(m,k)|^2 \qquad \text{(II.3)}$$

Où $|\mathrm{Y\ (m,k)}|^2$, $|\mathrm{S(m,k)}|^2$ et $|\mathrm{B(m,k)}|^2$ désignent respectivement le spectre de puissance du signal observé, du signal propre et du bruit.

Dans le choix de la méthode à utiliser, plusieurs critères entrent en jeu dont, au premier rang, la complexité et le caractère temps réel. Notre choix se porte sur la technique de soustraction spectrale, en raison de sa simplicité et son efficacité à réduire le bruit de fond.

II.3.5 Soustraction spectrale

C'est la technique la plus ancienne et sans doute la plus facile à réaliser [1]. Elle permet d'atténuer plus ou moins fortement les composantes spectrales du signal dégradé en fonction de l'estimation du niveau du bruit en adoptant les hypothèses sur les signaux de parole et le bruit. En supposant de plus que le signal de bruit est de moyenne nulle, stationnaire et n'est pas corrélé avec le signal propre.

Cette méthode consiste à effectuer une décomposition spectrale uniforme du signal bruité par le biais d'une fenêtre d'analyse suivie d'une transformée de Fourier. La définition de l'approche par soustraction spectrale est donnée par :

$$|\hat{S}(m,k)|^2 = |Y(m,k)|^2 - |\hat{B}(m,k)|^2 \quad \text{(II.4)}$$

où $|\hat{S}(m,k)|^2$ est l'estimé du spectre de puissance du signal propre et $|\hat{B}(m,k)|^2$ est l'estimé du spectre de puissance du bruit.

Le principe de base de cette technique vise à obtenir une estimation spectrale du signal utile dégradé par un bruit additif en soustrayant le module du spectre du bruit à partir de celui de la parole bruitée. L'amplitude moyenne du spectre de bruit pourra, par exemple, être approximé durant les périodes de silence. Durant les intervalles de parole, on suppose que le spectre de puissance du bruit est celui qui a été estimé durant le dernier silence.

Le processus consiste ensuite à soustraire raie par raie l'estimation de bruit du signal bruité. Il est alors essentiel de fonctionner en trames fenêtrées qui seront finalement recombinées par transformée de Fourier inverse pour la synthèse du signal débruité.

Dans le cas où l'équation (II.4) donne un résultat négatif, provoque le problème de l'apparition des valeurs négatives pour l'estimation du spectre de puissance du signal propre, on utilise une rectification demi-onde (mise à zéro des parties négatives) :

$$|\hat{S}(k)|^2 = \begin{cases} |Y(k)|^2 - \left|\hat{B}(k)\right|^2 & si\ |Y(k)|^2 > |\hat{B}(k)|^2 \\ 0 & ailleurs \end{cases} \quad \text{(II.5)}$$

Une fois tout le traitement est effectué, la phase de la parole bruitée sera ajoutée au spectre d'amplitude traité, suivi par l'application d'une TFD inverse (TFDI) afin d'obtenir le signal débruité dans le domaine temporel à court terme.

$$\hat{s}(m,n) = TFDI\ [\ |\hat{S}(m,k)|e^{j\theta_y(m,k)}\ ] \quad \text{(II.6)}$$

Où n désigne l'indice temporel, m désigne le numéro de la trame et $\theta_y(m,k)$ représente la phase du signal observé bruité $Y\ (m,k)$.

La figure (II.1) schématise les différents blocs de la méthode de rehaussement par soustraction spectrale :

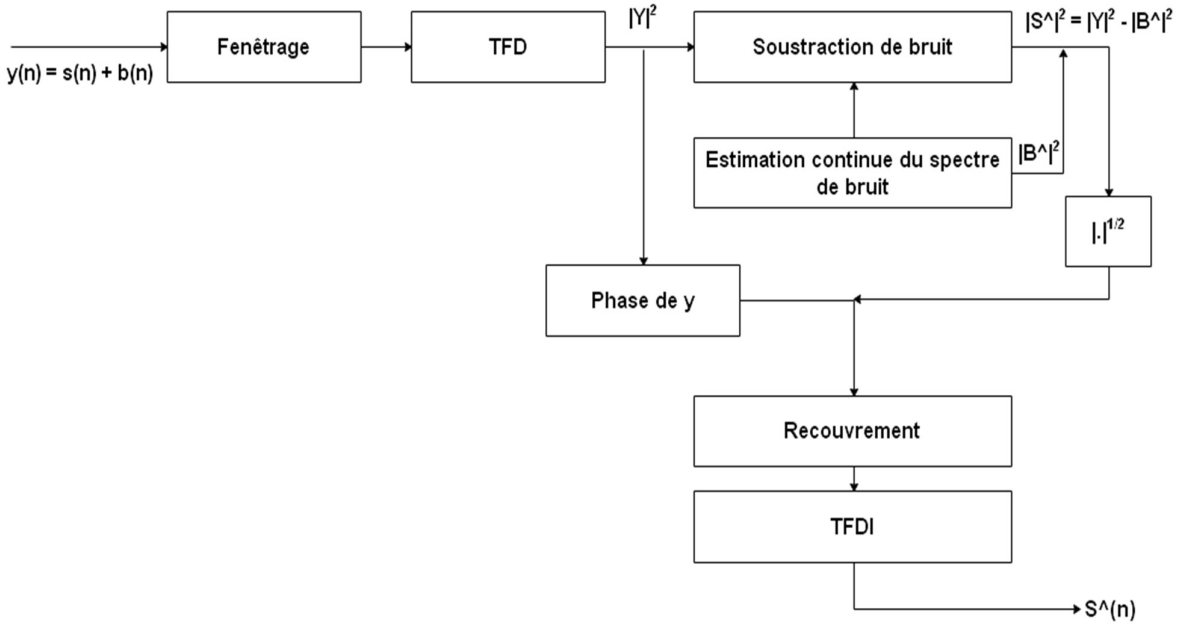

Figure II.1 : Synoptique de dèbruitage par soustraction spectrale.

Cette méthode introduit un bruit musical qui peut être dans certain cas plus gênant que les distorsions causées par le bruit interférant [12].

L'origine du bruit musical est la variance des estimateurs locaux de la densité spectrale des signaux. En effet, comme le spectre à court-terme du bruit fluctue autour des valeurs moyennes, son amplitude atteint à certains instants et pour certains indices fréquentiels des valeurs largement supérieures à la moyenne.

Afin de diminuer l'influence du bruit musical et avoir une bonne qualité du signal à la sortie, Plusieurs variantes de la soustraction spectrale ont été proposées dans la littérature (voir par exemple [14, 15]). Une approche générale permet d'écrire :

$$|\hat{S}(m,k)| = \begin{cases} \left[|Y(m,k)|^{\gamma_1} - \alpha\left|\hat{B}(m,k)\right|^{\gamma_1}\right]^{\gamma_2} & si\ |Y(m,k)|^{\gamma_1} > \alpha\left|\hat{B}(m,k)\right|^{\gamma_1} \\ \beta|\hat{B}(m,k)|^{\gamma_1\gamma_2} & sinon \end{cases} \quad \text{(II.7)}$$

Où α est un paramètre de surestimation du bruit [1], β est un paramètre de contrôle de niveau du bruit résiduel [2] et les exposants γ1 et γ2 sont des termes qui ont un effet sur l'intelligibilité de la parole débruitée [14].

Le cas où $\gamma_1 = 2$ et $\gamma_2 = 0.5$ correspond à la soustraction spectrale de puissance alors que le nom soustraction spectrale d'amplitude est réserve au cas $\gamma_1 = \gamma_2 = 1$.

II.3.5.1 Soustraction spectrale de Berouti

Berouti et al. ont révolutionné la méthode de la soustraction spectrale de puissance en apportant des modifications à l'algorithme de base de cette dernière. Avec ces modifications apportées, la méthode de la soustraction spectrale devient plus efficace et diffère des autres méthodes par deux applications [2] :

- Premièrement, l'estimation du bruit soustrait est amplifiée par un facteur α plus grand que l'unité.
- Deuxièmement, un seuil minimal est fixé pour éviter que la soustraction cause un résultat plus bas qu'un certain niveau.

Ce seuil correspond à une fraction de l'estimé du bruit, représenté par β. Ainsi, l'équation (II.7) devient :

$$|\hat{S}(k)|^2 = \begin{cases} |Y(k)|^2 - \alpha\left|\hat{B}(k)\right|^2 & si\ \left|\hat{S}(k)\right|^2 > \beta|\hat{B}(k)|^2 \\ \beta|\hat{B}(k)|^2 & ailleurs \end{cases} \quad \text{(II.8)}$$

Avec :

- α : Le facteur de soustraction (surestimation), (α >1).
- β : Le paramètre de lissage spectral, (0< β <<1).

La méthode de Berouti est représentée par le diagramme suivant :

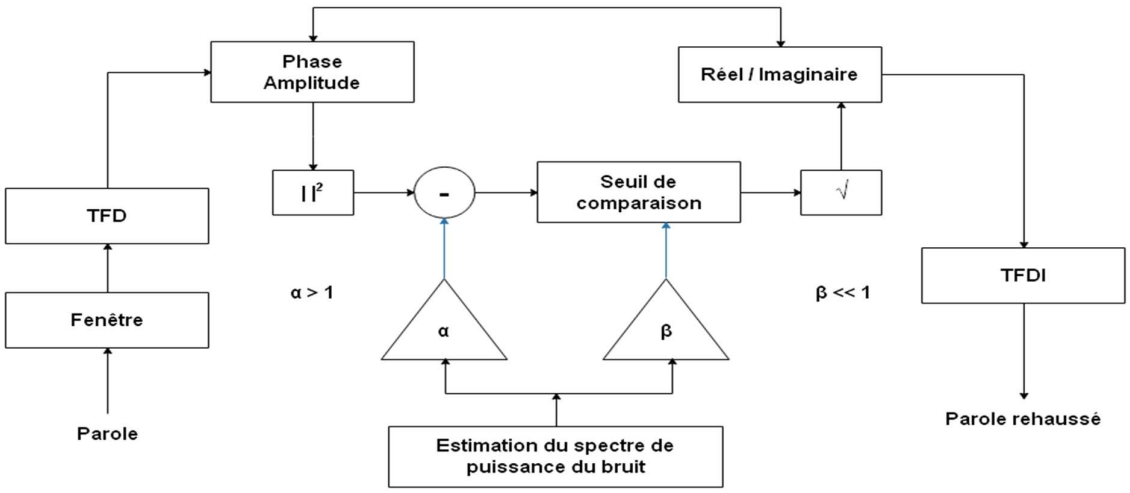

Figure II.2 : Soustraction spectrale proposée par Berouti et al.

II.3.5.2 Influence des paramètres

Le signal de la parole rehaussée, qui résulte de l'application de cette méthode est affecté par deux types de bruits :

- Bruit à large bande, connu sous le nom de bruit résiduel.
- Bruit à bande étroite, connu sous le nom de bruit musical.

Ces deux bruits apparaissent sous forme de pics et de vallées dans le spectre du signal rehaussé. Ils ont une distribution aléatoire et changent aléatoirement en fréquence et en amplitude d'une trame à l'autre.

La réduction des pics spectraux du bruit résiduel est effectuée par le facteur de soustraction qui prend toujours une valeur supérieure à l'unité ($\alpha > 1$). Si une valeur élevée de α est prise, la réduction du bruit à large bande se fait, mais cela provoque une distorsion du signal de la parole.

Afin d'obtenir des valeurs optimales du facteur de soustraction α , il faut prendre en compte que α est une fonction du rapport signal sur bruit segmental, dont sa valeur réelle est donnée par la relation suivante :

$$\alpha = \begin{cases} 5 & SNR < -5dB \\ \alpha_0 - \left(\frac{SNR}{s}\right) & -5dB < SNR < 20dB \\ 1 & SNR > 20dB \end{cases} \quad \text{(II.9)}$$

Avec :

- α_0 : La valeur de α pour un SNR =0. Dans la pratique $3 < \alpha_0 < 6$.
- 1/s : la pente de la droite dans la figure (2.4)
- SNR : le rapport signal sur bruit segmental estimé.

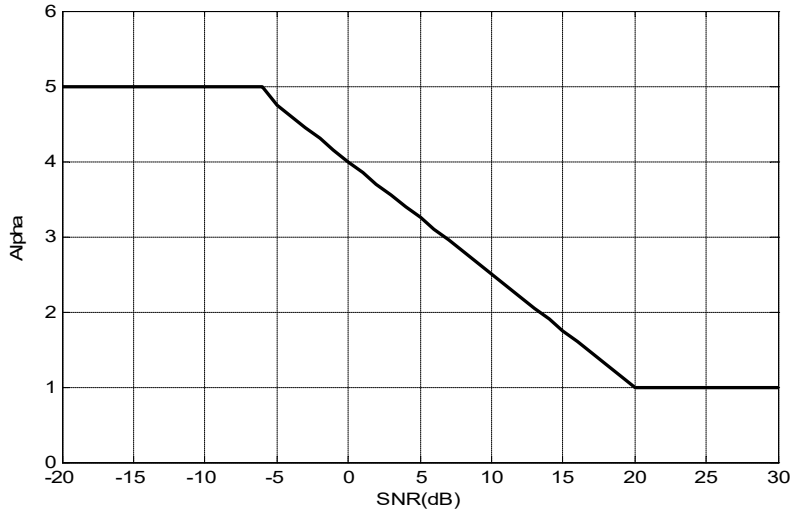

Figure II.3 : Valeurs de α en fonction du SNR.

Outre la réduction des pics, il y a le problème du remplissage des vallées d'où la réduction du bruit musical. Cela est effectué par le facteur de lissage β qui prend des valeurs dans l'intervalle $0 < \beta << 1$.

- $\beta > 0$: les pics du bruit résiduel sont masqués par les composantes spectrales voisines.
- $\beta << 1$: le bruit à large bande est plus bas par rapport à celui obtenu dans le cas où $\beta = 0$.

Donc le choix de β a une importance majeure pour le spectre de puissance du signal propre estimé $\left|\hat{X}(k)\right|^2$, Il est constaté aussi que :

- Si β est faible : le bruit résiduel sera réduit, mais le bruit musical sera audible.
- Si β est grande : le bruit musical n'est pas audible mais le bruit résiduel reste présent.

La soustraction spectrale est efficace du point de vue complexité de calcul, elle est caractérisée par un simple mécanisme de contrôle du compromis entre la distorsion du signal de la parole améliorer et le bruit résiduel dans ce dernier. Un des gros désavantages de cette technique telle que présentée est que l'estimation du bruit induit un nouveau bruit qui contient une certaine musicalité, nommé "bruit musical".

II.3.5.3 Limitations de la technique de soustraction spectrale

Les techniques de soustraction spectrale se basent sur l'unique hypothèse de non corrélation du signal de parole et du bruit [13]. Ces méthodes ne prennent pas en compte les propriétés et les caractéristiques de la parole (forte corrélation de la parole, probabilité de présence de la parole dans le signal bruité, distribution de la parole, …).

Ainsi, les atténuations des composantes fréquentielles de la trame bruitée actuelle de numéro (m) ne dépendent en fait que des mesures réalisées durant la trame elle-même. Ceci est bien entendu contradictoire avec la nature même du signal de parole qui se caractérise par une forte corrélation. Certains travaux ont alors remédié à cette limitation. Par exemple, Boll a proposé de calculer l'atténuation à partir de la moyenne des amplitudes spectrales sur plusieurs trames [1].

II.4 Méthodes d'estimation du bruit

Le problème de la réduction du bruit en parole devient de plus en plus difficile, dès qu'il n'y a aucune référence valable pour estimer le bruit. Dans ce cas, le bruit est estimé en utilisant les propriétés du signal dégradé, comme la stationnarité et le contenu spectral, ou alors durant les moments du silence, où la parole est absente. En l'absence de connaissance a priori sur la

densité spectrale du bruit, l'estimation du bruit utilisée dans ce travail est une estimation continue basée sur la méthode de Hirsch [16] et la méthode du MCRA (Minimum Controled Recursive Averaging) [17].

Cette estimation devra fournir une information précise et fiable sur la densité spectrale de puissance du bruit, cela permet d'avoir une information à la fois sur le niveau du bruit et sur son contenu spectral.

II.4.1 Méthode de Hirsch (weighted averaging technique)

Dans [16], Hirsch a proposé deux méthodes pour estimer les paramètres spectraux du bruit sans détection explicite de pause de la parole. Dans les deux méthodes, l'estimation spectrale du bruit a été mise à jour en comparant le spectre de puissance de la parole bruitée à l'estimation actuelle du bruit.

La première approche consiste en un algorithme qui calcule le niveau du bruit dans chaque composante fréquentielle, comme une moyenne pondérée des valeurs des amplitudes spectrales précédentes, qui sont au-dessous d'un seuil adaptatif. Cette méthode peut être utilisée dans une application en temps réel.

La deuxième approche est basée sur l'histogramme des segments de bruit précédents. Elle est généralement utilisée pour le traitement des enregistrements anciens.

Dans ce qui suit, nous allons uniquement faire l'étude de la méthode de la moyenne pondérée (première approche) en raison de son utilisation en temps réel.

II.4.1.1 Estimation du spectre de bruit

La plupart des techniques monovoie de réduction du bruit ont besoin d'une estimation du spectre du bruit. Ceci est habituellement fait par détection des pauses de la parole pour estimer des segments de bruit pur. Dans les situations pratiques c'est une tâche difficile particulièrement si le bruit de fond n'est pas stationnaire ou le rapport signal sur bruit (SNR) est faible. Cette technique permet d'éviter le problème de la détection de pause de la parole et d'estimer les caractéristiques du bruit uniquement à partir des segments précédents bruités.

II.4.1.2 Présentation de la méthode

La méthode de Hirsch calcule la somme pondérée des amplitudes spectrales précédentes $|Y(k,m)|^2$ pour chaque composante fréquentielle (k). La pondération est faite par un simple système récursif du premier ordre. On aura :

$$B(k,m) = \alpha B(k,m-1) + (1-\alpha)|Y(k,m)|^2 \qquad \text{(II.10)}$$

Où $|Y(k,m)|^2$ est le spectre de la k^{ème} composante fréquentielle de la m^{ème} trame et $D(k,m)$ est une estimation du spectre du bruit. Au lieu d'utiliser une simple moyenne des valeurs de puissance spectrales précédentes comme estimation du spectre du bruit, un seuil adaptatif est introduit dans cet algorithme.

Les amplitudes $|Y(k,m)|^2$ sont modélisées selon une distribution de Rayleigh dans les segments de bruit pur. Des valeurs considérablement plus élevées se produisent au début de la parole. Ainsi un seuil $\beta B(k,m-1)$ est introduit, où β est un facteur de surestimation qui prend une valeur dans l'intervalle [1.5, 2.5]. Ce seuil est adaptatif car il varie selon le niveau de puissance du bruit présent dans la parole bruitée. Ainsi, il peut suivre les changements lents des niveaux de la puissance du bruit, dans le cas des bruits où les statistiques varient lentement.

Quand la composante spectrale actuelle $|Y(k,m)|^2$ excède ce seuil, ceci est considéré comme une détection rugueuse de la parole et l'accumulation récursive est arrêtée. A ce moment-là, les valeurs accumulées sont prises comme une estimation du niveau de bruit. Ainsi l'algorithme de Hirsch peut être récapitulé comme suit :

Si $|Y(k,m)|^2 < \beta\hat{B}(k,m)$ alors

$B(k,m) = \alpha B(k,m-1) + (1-\alpha)|Y(k,m)|^2$

Sinon

$B(k,m) = B(k,m-1)$

Fin

Cette approche peut être combinée avec la technique de la soustraction spectrale. Une bonne suppression est confirmée par des tests d'écoute et les effets négatifs comme les tonalités musicales peuvent être réduits en optimisant des paramètres, par exemple le facteur de surestimation.

Bien que cette approche fonctionne d'une manière satisfaisante dans la plupart des cas, elle échoue dans le cas suivant : Considérons un exemple où on aura une augmentation soudaine du niveau de bruit. Ceci aura comme conséquence une situation où le spectre de la parole bruitée ne sera jamais au-dessous du seuil, puisque le seuil est basé sur l'estimation précédente du bruit déjà très bas.

Ainsi, l'inconvénient majeur de cette méthode est que l'estimation du bruit ne sera pas mise à jour si la puissance du bruit augmente abruptement et demeure à ce niveau élevé. Par contre cet algorithme est simple à implémenter.

II.4.2 La méthode de MCRA (Minima Controlled Recursive Averaging : MCRA)

Dans [17] Cohen et Berdugo ont proposé une approche d'estimation du bruit basée sur le contrôle des minima par une moyenne récursive (Minima Controlled Recursive Averaging :

MCRA) qui consiste à chercher la moyenne des anciennes valeurs du spectre de puissance du signal bruité, à l'aide d'un paramètre de lissage. Ils sont obtenus en se basant sur la probabilité de présence du signal dans chaque composante fréquentielle séparément. Cette probabilité est obtenue à partir du rapport entre le spectre de puissance du signal et son minimum local. Ce rapport est comparé avec un seuil, où un rapport faible indique l'absence de la parole.

II.4.2.1 Estimation du spectre du bruit

Cette méthode est basée sur les deux hypothèses $H_0(k,l)$ et $H_1(k,l)$, qui indiquent respectivement l'absence et la présence de la parole dans la k^{ème} composante fréquentielle de la m^{ème} trame, on a :

$$\begin{aligned} H_0(k,m): Y(k,m) &= B(k,m) \\ H_1(k,m): Y(k,m) &= S(k,m) + B(k,m) \end{aligned} \tag{II.11}$$

La variance du bruit dans la k^{ème} composante fréquentielle est donnée par :

$$\lambda_b(k,m) = E[|B(k,m)|^2] \tag{II.12}$$

Afin d'obtenir une mise à jour de l'estimation du bruit, un lissage récursif dans le domaine temporel est appliqué à l'observation bruitée durant les périodes d'absence de la parole, on aura :

$$\begin{aligned} H_0'(k,m) &: \hat{\lambda}_b(k,m+1) = \alpha_b \hat{\lambda}_b(k,m) + (1-\alpha_b)|Y(k,m)|^2 \\ H_1'(k,m) &: \hat{\lambda}_b(k,m+1) = \hat{\lambda}_b(k,m) \end{aligned} \tag{II.13}$$

Avec $\hat{\lambda}_b(k,m)$: L'estimation de la variance du bruit, $\alpha_b (0 < \alpha_b < 1)$: Le facteur de lissage, H_0', H_1' : Les hypothèses d'absence et de présence de la parole qui contrôlent l'adaptation du spectre du bruit.

Soit $p'(k,l) \cong P(H_1'(k,l)/\mathrm{Y}(k,l))$ la probabilité conditionnelle de la présence de la parole, donc à partir de l'équation (3.27) on a :

$$\begin{aligned} \hat{\lambda}_b(k,m+1) &= \hat{\lambda}_b(k,m)p'(k,m) \\ &\quad + \left[\alpha_b \hat{\lambda}_b(k,m) + (1-\alpha_b)|Y(k,m)|^2\right]\left(1-p'(k,m)\right) \\ &= \tilde{\alpha}_b(k,m)\hat{\lambda}_b(k,m) + [1-\tilde{\alpha}_b(k,m)]|Y(k,m)|^2 \end{aligned} \tag{II.14}$$

Avec :

$$\tilde{\alpha}_b(k,m) = \alpha_b + (1-\alpha_b)\, p'(k,m) \tag{II.15}$$

Où α_b: est un paramètre de lissage variant dans le temps. Par conséquent, le spectre du bruit peut être estimé en faisant la moyenne des valeurs des spectres de puissance précédents, en utilisant un paramètre de lissage qui est ajusté par la probabilité de présence de la parole.

II.4.2.2 Probabilité de présence du signal

L'étape suivante est le calcul de la probabilité de présence de la parole dans une trame donnée, le rapport entre le spectre de puissance du signal bruité et son minimum dans une fenêtre de taille spécifiée, donne cette probabilité pour chaque composante fréquentielle.

L'énergie locale du signal bruité est obtenue en lissant l'amplitude carrée de sa transformée de Fourier à court terme en temps et en fréquence

- En fréquence, on utilise une fonction de fenêtrage 'd' dont la longueur est $2w + 1$:

$$S_f(k,m) = \sum_{i=-w}^{w} d(i)\,|Y(k-i,m)|^2 \qquad \text{(II.16)}$$

- En temps, le lissage est réalisé par une moyenne récursive du 1$^{\text{èr}}$ordre :

$$S(k,m) = \alpha_s S(k,m-1) + (1-\alpha_s) S_f(k,m) \qquad \text{(II.17)}$$

Où α_s est un facteur de lissage.

Afin d'obtenir l'énergie locale minimale $S_{min}(k,m)$et une valeur temporelle $S_{tmp}(k,m)$ les auteurs procèdent comme suit :

- L'initialisation de : $S_{min}(k,0) = S(k,0)$ et $S_{tmp}(k,0) = S(k,0)$
- Calcul de la valeur minimale de la trame actuelle par une simple comparaison entre l'énergie locale et la valeur minimale de la trame précédente, comme suit :

$$\begin{aligned} S_{min}(k,m) &= min\{S_{min}(k,m-1), S(k,m)\} \\ S_{tmp}(k,m) &= min\{S_{tmp}(k,m-1), S(k,m)\} \end{aligned} \qquad \text{(II.18)}$$

Pour chaque ' M ' trames lues, c'est-à-dire ' m' est divisible par M, la valeur temporelle utilisée est initialisée par :

$$\begin{aligned} S_{min}(k,m) &= min\{S_{tmp}(k,m-1), S(k,m)\} \\ S_{tmp}(k,m) &= S(k,m) \end{aligned} \qquad \text{(II.19)}$$

Et la recherche du minimum continue avec l'équation (II.18). Le paramètre 'M' détermine la résolution de la recherche du minimum local. Selon les expériences une fenêtre de

durée allant de 0.5 à 1.5 seconde est souhaitable. Une très petite longueur de fenêtre peut être considérée comme une cause de la surestimation du bruit si la largeur de cette fenêtre est inférieure à la largeur du pic de la parole. En outre, une très grande fenêtre retarde la mise à jour de l'estimation de la variance du bruit, en particulier pour les bruits où les niveaux changent brusquement.

Pour le rapport entre l'énergie locale de la parole bruitée et son minimum local :

$$S_r(k,m) = S(k,m)/S_{min}(k,m) \tag{II.20}$$

La règle de décision basée sur le coût minimum de Bayes est donnée par :

$$\frac{p(S_r/H_1)}{p(S_r/H_0)} \underset{H_0'}{\overset{H_1'}{\gtrless}} \frac{c_{10}P(H_0)}{c_{01}P(H_1)} \tag{II.21}$$

Où : $P(H_0)$ et $P(H_1)$ sont les probabilités a priori pour l'absence et la présence de la parole respectivement et c_{ij} : est le coût de décider H_i' quand H_j' est vraie.

Comme le rapport de vraisemblance $p(S_r/H_1)/p(S_r/H_0)$ est une fonction monotone, la règle de décision de l'équation (II.21) peut être exprimée comme suit :

$$S_r(k,l) \underset{H_0'}{\overset{H_1'}{\gtrless}} \delta \tag{II.22}$$

Donc la probabilité de présence de la parole est donnée par :

$$\hat{p}'(k,m) = \alpha_p \hat{p}'(k,m-1) + (1-\alpha_p)\, I(k,m) \tag{II.23}$$

Avec : α_p est un paramètre de lissage et $I(k,m)$ est une fonction d'indicateur pour le résultat dans (II.22), c'est-à-dire :

$$I(k,m) = \begin{cases} 1 & si\ S_r(k,m) > \delta \\ 0 & ailleurs \end{cases} \tag{II.24}$$

- Le seuil δ n'est pas sensible au type et à l'intensité du bruit de l'environnement.
- La corrélation de la présence de la parole dans les trames consécutives est utilisée via α_p.

L'inconvénient principal de l'algorithme MCRA est la mise à jour du minimum local de la parole bruitée pour les niveaux de bruit croissants. Selon la règle de suivi du minimum, la valeur minimale est choisie comme le minimum entre les minimums locaux estimés précédemment et la puissance de la parole bruitée actuelle, comme définie dans l'équation (II.18). En outre, pour éviter de descendre au-delà d'un minimum global, la variable temporelle est mise à jour à chaque L trames, où sa valeur est prise égale à la puissance de la parole bruitée dans cette M^{ème} trame comme dans (II.19). Le minimum local est mis à jour en utilisant la variable temporelle à chaque 2M trames. Par conséquent, une durée correspondante à 2M trames est nécessaire pour mettre à jour le minimum local pour les niveaux croissants d'un bruit. Donc, pour une fenêtre de longueur de 0.5 à 1.5s, la méthode nécessite de 1s jusqu'à 3s pour s'adapter aux niveaux élevés du bruit.

II.5 Implémentations de la soustraction spectrale et résultats

Le jugement de la qualité du signal de parole après un certain traitement (dèbruitage, codage, transmission, ...) ne peut se faire d'une manière satisfaisante qu'à partir de tests subjectifs. En effet, tout test subjectif doit être réalise avec plusieurs personnes d'âge et de sexe différents et dans des conditions expérimentales bien déterminées. Dans ce cadre, l'Union International des Télécommunications (UIT) a développé des normes spécifiant les procédures expérimentales à suivre pour évaluer la qualité subjective du signal vocal. Citons par exemple, la norme P.800 [12].

Dans la pratique, les tests subjectifs permettent d'évaluer avec précision la qualité de la parole en se basant sur sa perception par l'oreille humaine, mais ils restent coûteux de point de vue temps et équipements expérimentaux. Pour cela, plusieurs critères objectifs ont été établis, trois (3) critères les plus connus et qui sont souvent exploités sont :

- Critères temporels qui visent à comparer les formes temporelles du signal propre et celui traité. Citons par exemple le rapport signal sur bruit (RSB) [46] et le RSB segmental (RSBseg) [19].
- Critères fréquentiels qui permet de comparer les spectres du signal à évaluer avec celui généralement propre. Noter, à titre d'exemple la mesure Log-Likelihood Ratio (LLR) [49], la mesure Weighted Spectral Slope(WSS) [21], etc.
- Critères perceptuels qui utilisent les notions perceptuelles. Par exemple, la mesure PESQ : Perceptual Evaluation of Speech Quality [22].

II.5.1 Critère temporel utilisé

Le Rapport Signal sur Bruit (RSB) est une mesure de la quantité de la dégradation par rapport à celle du signal propre. Elle est formulée par l'expression suivante :

$$RSB = 10log_{10}\left\{\frac{\sum_{n=0}^{n=NT-1} x^2(n)}{\sum_{n=0}^{n=NT-1}[x(n)-y(n)]^2}\right\} \quad \text{(II.25)}$$

où x(n) (resp. y(n)) représente le signal propre (resp. le signal traité). NT désigne le nombre total d'échantillons de la séquence vocale.

Nous avons, également, utilisé le RSB segmental (RSBseg) qui n'est d'autre qu'une moyenne des RSBs calculés chacun sur une trame [19]. Le rapport signal sur bruit segmental présente une forte corrélation avec les tests subjectifs tandis que le rapport signal sur bruit global présente une faible corrélation. Il est obtenu en faisant la moyenne de tous les rapports signal sur bruit de chaque trame d'analyse comme suit :

$$RSB_{SEG} = \frac{1}{M}\sum_{m=0}^{m=M-1} 10log_{10}\left\{\frac{\sum_{n=mN}^{n=mN+N-1} x^2(m,n)}{\sum_{n=mN}^{n=mN+N-1}[x(m,n)-y(m,n)]^2}\right\} \qquad \text{(II.26)}$$

Où M désigne le nombre total des trames, N désigne la taille de la trame et m représente le numéro de la trame.

Durant les périodes de silence, le SNR peut atteindre des valeurs négatives très faibles, d'où la nécessité de considérer une valeur limite inférieure dans la gamme (0dB, -20dB). Généralement, une limite supérieure à 35 dB est utilisée du fait que les trames dont le SNR dépasse 35 dB peuvent affecter la corrélation du SNR segmental avec les tests subjectifs.

II.5.2 Critère fréquentielle utilisé

La mesure LLR se base essentiellement sur la modélisation autorégressive de la parole**.** Cette méthode se base sur la différence entre la forme spectrale des modèles LPC des signaux propres et rehaussé. La mesure LLR peut être vue comme une distance est calculée comme suit :

$$LLR = log\left(\frac{\vec{a}_y \, \boldsymbol{R}_x \, \vec{a}_y}{\vec{a}_x \, \boldsymbol{R}_x \, \vec{a}_x}\right) \qquad \text{(II.27)}$$

Où a_x (resp. a_y) désigne le vecteur de coefficient de prédiction linéaire du signal propre (resp. du signal traité) et R_x représente la matrice d'autocorrélation du signal propre.

II.5.3 Critère perceptuelle utilisé

Le PESQ est un outil permettant d'offrir une méthodologie de tests afin d'évaluer la qualité de la parole dans un contexte de télécommunication [23]. C'est une méthode objective par laquelle la prévision de la qualité subjective des signaux sonores s'effectue. Son principe étant de comparer un signal dégradé avec le signal de référence correspondant (Figure II.4). La mesure PESQ a été adoptée comme une recommandation P.862 de l'UIT.

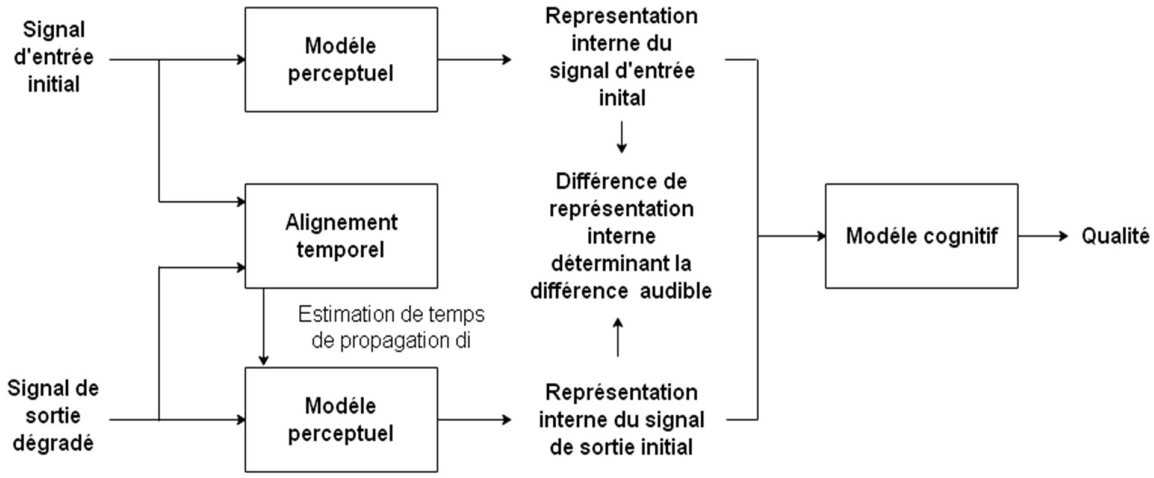

Figure II.4 : Principe de fonctionnement du modèle PESQ.

La mesure PESQ a été créé dans le but de prédire l'évaluation subjective MOS (Mean Opinion Score) [24], La note MOS évalue la qualité de la séquence vocale sur une échelle de 5 valeurs (**Tableau II.2**).

Tableau II.2 : Echelle de la qualité d'écoute pour la méthode ACR.

Note	Catégorie
5	Excellente
4	Bonne
3	Passable
2	Médiocre
1	Mauvaise

La notation se fait généralement avec la méthode d'évaluation par catégories absolues ACR (Absolute Category Rating), définie par l'Union International des Télécommunications (UIT), se forme d'un scalaire compris entre -0.5 et 4.7.

II.5.4 Conditions d'implémentation

Les simulations ainsi que tous les tests effectués sur les différents algorithmes traités ont été effectués sur un PC Intel Core i3, 2 GHz, 8Go de RAM, avec un logiciel MATLAB version 9.7.0.1190202 (R2019b). Ce logiciel permet de simplifier la mise au point des algorithmes de rehaussement sans contraintes de temps et de mémoire, et il est devenu un outil indispensable et simple pour le test des algorithmes de traitement du signal avant l'implémentation en temps réel.

II.5.5 Base de données utilisée

Dans ce chapitre, nous allons utiliser la base de données **« Noizeus »** qui est la base de données standard pour les travaux de rehaussement de la parole bruitée. Cette base de données

est disponible sur le site internet : (https://ecs.utdallas.edu/loizou/speech/noizeus/). L'utilité de cette base de données est l'amélioration et la simplicité de la comparaison entre les performances des algorithmes d'estimation de bruit. Elle comporte trente phrases phonétiquement équilibrées, produites par trois locuteurs et trois locutrices (chaque locuteur produit cinq phrases), l'enregistrement de ces phrases s'est effectué au niveau des salles acoustiquement isolées en utilisant des équipements de type « Tucker Davis Technologies (TDT) avec une fréquence d'échantillonnage égale à 25 kHz (ensuite elles seront rééchantillonnées à 8 kHz).

Au cours de cette étude nous allons bruitées artificiellement les trente phrases par différents bruits pris de la base de données AURORA (blanc, babble, aéroport) avec des rapports signal-sur-bruit de 0 dB et de 5 dB. Ainsi, pour chaque méthode et pour un type de bruit donné et un niveau de signal-sur-bruit bien défini, les trente phrases de parole propre et les phrases de parole bruitée correspondantes seront utilisées pour chaque mesure objective, ensuite une moyenne sera calculée.

II.5.6 Evaluation des performances et résultats

Dans cette partie, nous allons évaluer objectivement les performances de la méthode de la soustraction spectrale implémentée avec les deux algorithmes d'estimation du bruit étudiés précédemment (Hirsch et MCRA).

Ces méthodes d'estimation du bruit sont comparées par des tests objectifs et des tests d'écoute. Les tests ont été effectués en utilisant plusieurs types de bruit, mais on se limitera durant la présentation des résultats a trois types de bruits seulement (bruit blanc, bruit babble et bruit d'aéroport) avec un SNR = 5 dB puis un SNR = 0 dB. Ces algorithmes d'estimation du bruit sont combinés avec un système de réduction de bruit basé sur la soustraction spectrale de puissance développée par Berouti.

Les résultats de test des mesures (LLR, SNRseg, PESQ) sont représentés dans les tableaux présentés ci-dessous.

Les différents paramètres de la soustraction spectrale ainsi que ceux de chaque variante des algorithmes d'estimation ont été sélectionnés et choisis d'une manière à assurer de bons résultats.

- **Paramètres de la méthode de Hirsch**

  $\alpha = 0.85$

  Et

  $\beta = 2.5$

- **Paramètres de la méthode MCRA**

  $\alpha_d = 0.95\,,\ \alpha_{\mathrm{p}} = 0.2\,,\ \alpha_{\mathrm{s}} = 0.8$

  Et

  $w = 1,\ \mathrm{L} = 100,\ \delta = 5$

Les deux tableaux (II.3) et (II.4) nous donnent les résultats de tests des mesures (SNRseg, LLR, PESQ) pour des signaux dégradés par un bruit (Blanc, Babble ou Aéroport) pour les niveaux de bruit 5 dB et 0 dB.

Tableau II.3 : Résultats de test des mesures (SNRseg, LLR, PESQ) pour des signaux dégradés par un bruit seul (Blanc, Babble, Aéroport) avec SNR = 0 dB.

	0dB								
	Bruit blanc			**Bruit babble**			**Bruit d'aéroport**		
	SNRseg	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**
Dégradé	-5.0813	1.8022	1.5393	-4.6320	0.8950	1.7054	-4.5038	1.1789	1.6048
MCRA	**-1.5984**	**1.7377**	**1.8184**	**-2.9547**	**1.1466**	1.7151	-2.5488	**1.0552**	1.7251
Hirsch	-2.1754	1.8081	1.7660	-3.0051	1.1575	**1.7286**	**-2.5282**	1.0963	**1.7555**

Tableau II.4 : Résultats de test des mesures (SNRseg, LLR, PESQ) pour des signaux dégradés par un bruit seul (Blanc, Babble, Aéroport) avec SNR = 5 dB.

	5dB								
	Bruit blanc			**Bruit babble**			**Bruit d'aéroport**		
	SNRseg	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**
Dégradé	-2.3266	1.5450	1.7995	-1.7833	0.7152	2.0061	-1.6719	0.6911	2.0213
MCRA	**1.0245**	**1.4335**	**2.1865**	**-0.2688**	**0.8724**	**2.0713**	**-0.1590**	**0.8403**	2.0987
Hirsch	0.3916	1.5116	2.1329	-0.5314	0.9007	2.0532	-0.4293	0.8542	**2.1345**

II.5.6.2 Interprétations

- L'application de ces algorithmes d'estimation du bruit donne une amélioration en termes de mesures.
- La forme simple du bruit blanc permet d'avoir de bons résultats en général pour les trois mesures de performance, donc l'utilisation de ce dernier facilite la tâche d'estimation de bruit.
- L'algorithme MCRA est plus efficace par rapport à celle de Hirsch pour les trois types de bruit.
- Pour le RSB segmental la technique MCRA ont des résultats supérieurs à celle de la technique Hirsch.
- Les plus petites valeurs de la mesure LLR sont obtenues par l'algorithme MCRA.

- Le score PESQ du bruit babble est plus faible par rapport aux autres bruits dans tous les cas du SNR, ce qui est normal étant donné qu'il est difficile d'estimer le bruit.
- Le bruit babble a la forme d'un signal de parole, ceci explique les résultats que nous avons obtenus pour ce type de bruit par rapport aux autres types.
- Les mesures obtenues sont meilleures dans le cas du rapport signal sur bruit (SNR) égale à 5 dB où le niveau du bruit est faible par rapport au cas du rapport signal sur bruit (SNR) égal à 0 dB.

II.6 Conclusion

Dans ce chapitre, nous avons introduit la technique de débruitage de la parole basée sur l'estimation du spectre du signal propre à partir de celui débruité. Dans ce cadre, Nous nous sommes intéressés, particulièrement, aux techniques de soustraction spectrales et ses dérivées. Un inconvénient majeur de cette technique est l'apparition d'un bruit ayant un caractère musical. Ce caractère est dû à l'apparition des pics dans le spectre du signal débruité. Toutefois, malgré cet inconvénient, les techniques basées sur la soustraction spectrale restent performantes en termes d'atténuation du bruit.

Ainsi, Nous avons également détaillé et expliqué deux méthodes d'estimation du bruit, et qui ont été implémentées avec la méthode de la soustraction spectrale. Ce qui nous a permet de vérifier l'efficacité d'un système complet monovoie de réduction de bruit.

Chapitre III

Notions théoriques sur les techniques CNN et NMF

III.1 Introduction

La restauration du signal vocal fût ainsi un des premiers problèmes à susciter une grande attention. Celle-ci a pour objectif de minimiser les distorsions qui se traduisent lors de l'apparition d'un certain nombre de dégradations, elles se traduisent par une atténuation de l'intelligibilité, et au bruit, intervenant lors de la formation des données.

Après avoir vu l'amélioration de la qualité de la parole dans le chapitre 2 par les techniques classiques de traitement de signal vocale, nous allons essayer deux autres méthodes basées sur les réseaux de neurones covolutionnel et l'autre baser sur la factorisation matricielle non négative pour résoudre le problème.

Le problème étant comment obtenir le signal parole propre ou du moins s'en approcher le plus ?

III.2 Système de rehaussement de la parole basé sur le DNN

Le rehaussement de la parole vise à réduire le bruit et à améliorer la qualité et l'intelligibilité de la parole bruitée, pour palier à ces problèmes les réseaux de neurones profond (DNN) ont connu un grand succès dans de nombreuses applications d'amélioration de la parole. Le rehaussement est effectué par un réseau de neurones profond sur le spectre de puissance ou une version de ce dernier des trames fenêtrées du signal d'entrée.

La structure d'un système de rehaussement de la parole basé sur le DNN est illustrée dans la figure (III.1), on distingue trois parties :

- Extraction des paramètres.
- Rehaussement par DNN.
- Reconstruction de la forme d'onde du signal.

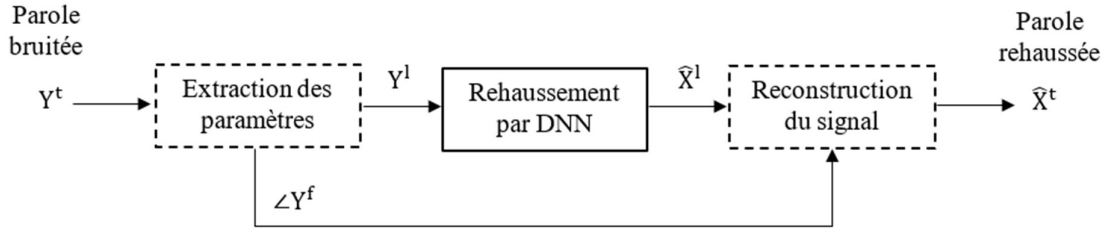

Figure III.1 : Système de rehaussement de la parole.

La première partie de l'extraction des paramètres avant le DNN, représentée à la figure (III.2) montre que le signal d'entrée Y^t dans le domaine temporel discret est divisé en trames à l'aide d'une fenêtre d'une longueur qui assure la stationnarité du signal et un chevauchement adéquat entre trames. La transformée de Fourier discrète (TFD ou DFT) permet d'obtenir les échantillons du spectre de fréquences de trames. Les deux étapes précédentes sont

essentiellement une implémentation de la transformée de Fourier à court terme (TFCT ou STFT).

Le spectre d'une trame est constitué de deux composantes : le spectre d'amplitude et le spectre de phase. Généralement, la phase du signal bruité, ($\angle Y^f$) est utilisée directement dans la reconstruction des trames, alors que les règles de rehaussement de la parole n'ont appliqué que sur le spectre d'amplitude.

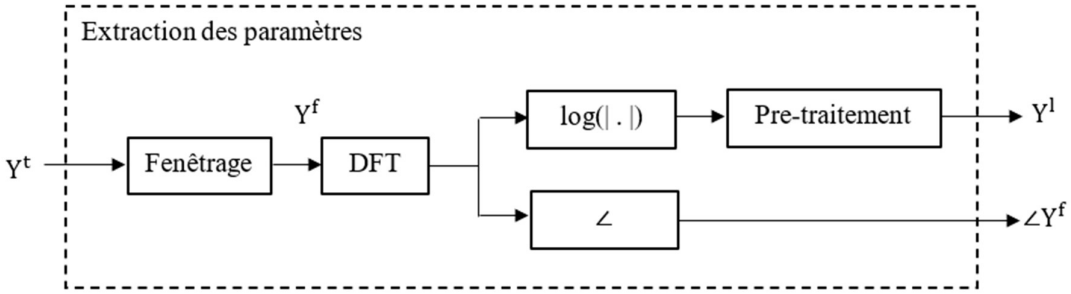

Figure III.2 : Diagramme d'extraction des paramètres.

Par la suite, Le logarithme est appliqué aux composantes du spectre d'amplitude qui sont ensuite mises à l'échelle pour avoir une moyenne nulle et une variance unitaire, en utilisant les estimations de la moyenne et de l'écart-type trouvées pour les données d'apprentissage.

Pour améliorer les performances du système, un contexte acoustique est fourni avec la trame à rehausser. Cela implique de combiner les paramètres spectraux mises à l'échelle de la trame actuelle avec ceux des trames précédentes et suivantes.

Le DNN effectue un mappage (mapping) des caractéristiques d'entrée, Y^l (contenant les composantes spectraux de plusieurs trames de parole bruitée) vers les composantes spectrales d'une seule trame rehaussée $\hat{X}^l$.

Une fois l'opération précédente terminée, la reconstruction de la forme d'onde (figure III.3) est effectuée à partir des vecteurs de sortie du DNN. Les paramètres de sortie du DNN, X^l, sont multipliées par un facteur optionnel d'égalisation de la variance globale et remises à l'échelle en utilisant les mêmes estimations de la moyenne et de l'écart-type que pour l'extraction des paramètres. L'inverse du logarithme est appliqué et l'estimation du spectre d'amplitude unilatéral est mise en miroir et combinée avec la phase de signal bruitée pour construire le spectre de fréquence linéaire complet de la trame rehaussée. La transformée de Fourier discrète inverse (TFDI ou IDFT) est ensuite appliquée et les trames sont combinées avec l'ajout de chevauchement pour former la forme d'onde rehaussée complète.

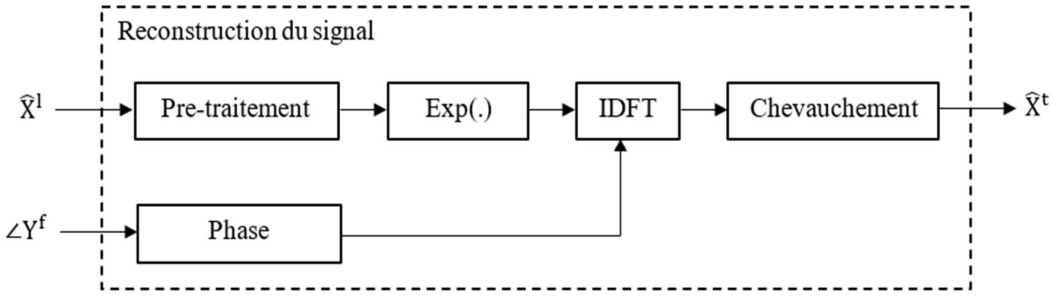

Figure III.3 : Diagramme de reconstruction de signal.

III.3 Réseau de neurones convolutionnel (CNN)

Le réseau de neurones convolutionnel (ou CNN : Convolutional Neural Network) est un algorithme de Deep Learning robuste et très important utilisé dans plusieurs applications. Ils ont été ainsi introduits avec succès en rehaussement de la parole.

En effet, L'élimination de bruit sans créer des distorsions dans la parole humaine est une tâche difficile dans un environnement à faible rapport signal/bruit. Nous avons cherché à résoudre ce problème en trouvant une "correspondance" entre les spectres de la parole bruité et les spectres de la parole propre par apprentissage supervisé.

Par analogie, leur fonctionnement est inspiré par les processus biologiques, il est constitué d'un empilage multicouche de perceptrons, dont le but est de prétraiter de petites quantités d'informations. S'il y a donc aujourd'hui une méthode qui justifie un engouement particulier, il s'agit donc bien des CNN.

III.3.1 Formulation du problème

Pour tous les algorithmes des réseaux de neurones qui seront présentés dans la suite de ce chapitre et comme mentionné auparavant, y(n), s(n) et b(n) désignent respectivement le signal observé bruité, le signal propre et le bruit additif. Par la suite, Étant donné un segment de spectres bruités $\{y_t\}_{t=1}^{T}$et de spectres propres $\{s_t\}_{t=1}^{T}$, notre objectif est d'apprendre une correspondance f qui génère un segment de spectres ' débruiter ' $\{f(y_t)\}_{t=1}^{T}$ qui se rapprochent des spectres propres dans la norme l2.

$$min \sum_{t=1}^{T} ||s_t - f(y_t)||_2^2 \qquad \text{(III.1)}$$

Plus précisément, nous formulons f en utilisant un réseau de neurones convolutionnel (voir Figure III.1) :

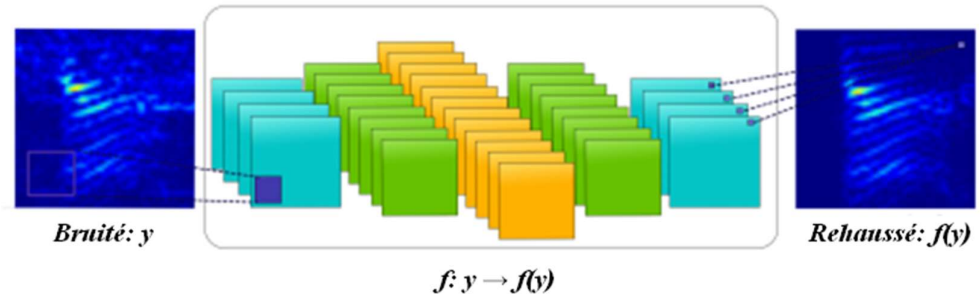

Figure III.4 : Rehaussement de la parole en utilisant le CNN.

Pour ce réseau, les n_T spectres bruités président $\{y_i\}_{i=t-n_T+1}^{t}$sont considérés comme les spectres débruités courants, par ex.

$$\sum_{t=1}^{T} ||S_t - f(y_{t-n_T+1}, \dots, y_t)||_2^2 \qquad \text{(III.2)}$$

III.3.2 Construction d'un réseau CNN

La première partie d'un CNN est la couche convolutive à proprement parler. Elle fonctionne comme un extracteur de paramètres des fichiers de parole. Un fichier de parole est passé à travers une succession de filtres, ou noyaux de convolution, créant de nouveaux fichiers appelés cartes de convolution. Au final, les cartes de convolutions sont mises à plat et concaténées en un vecteur de caractéristiques, appelé code CNN [25].

Généralement, une architecture de réseau de neurones convolutifs est formée par un empilement de couches de traitement : la couche de convolution (CONV), la couche de pooling (POOL), la couche de correction (ReLU) et la couche « entièrement connectée » (FC : Fully Connected), comme présentée sur la figure (III.5) :

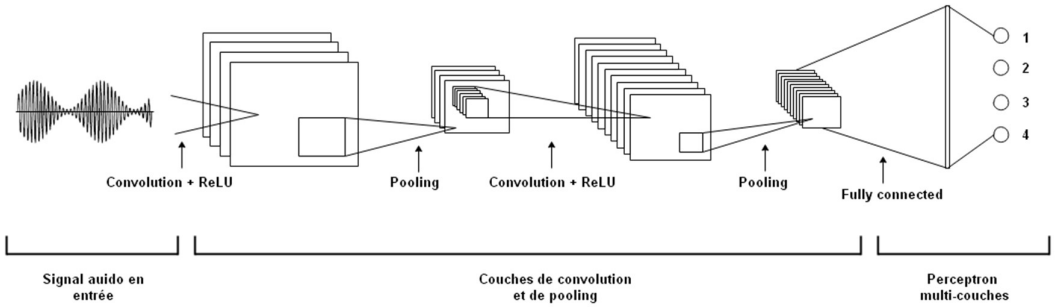

Figure III.5 : Architecture standard d'un réseau de neurone convolutionnel (CNN).

- **Couche de convolution (CONV) :**

La couche de convolution représente la composante clé des réseaux de neurones convolutifs dont le but est de se servir des valeurs présentes dans le filtre à chaque pas.

Au tout début de la convolution, la fenêtre sera positionnée tout en haut à gauche de la matrice puis elle va se décaler d'un certain nombre de cases (c'est ce que l'on appelle le pas)

vers la droite et lorsqu'elle arrivera au bout de la matrice [26], elle se décalera d'un pas vers le bas ainsi de suite jusqu'à ce que le filtre soit parcourue la totalité de la matrice comme représenter sur la figure suivante :

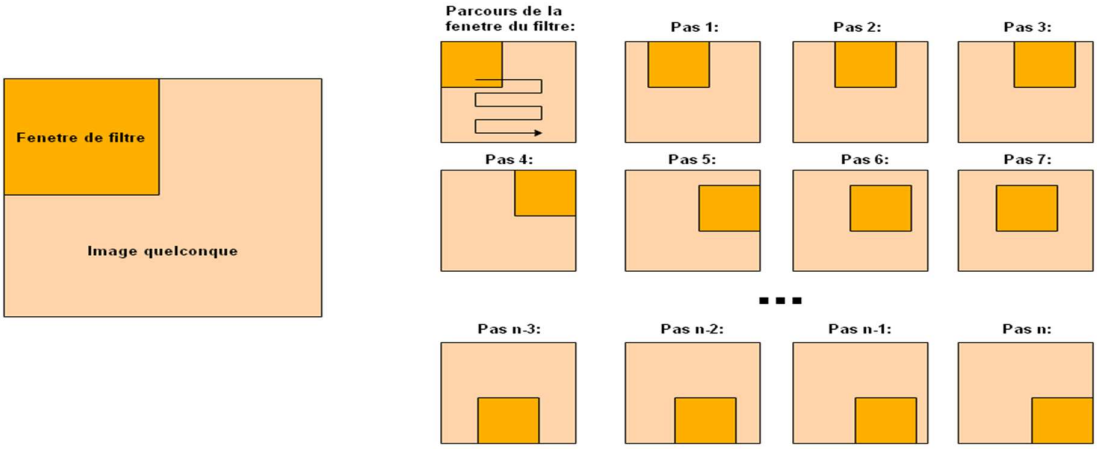

Figure III.6 : Schéma du parcours de la fenêtre de filtre sur la trame.

Ainsi que de calculer le produit de convolution entre ce dernier et chaque portion de la matrice balayée.

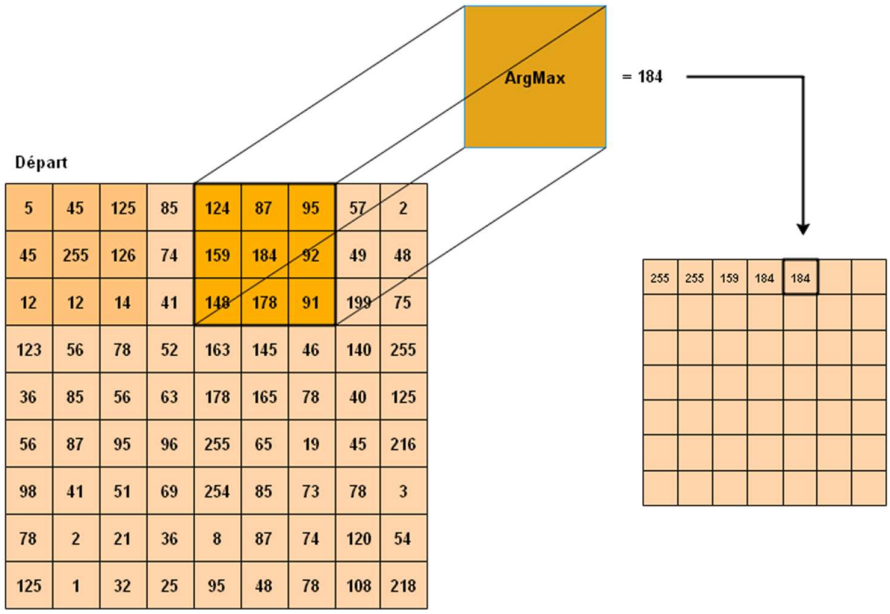

Figure III.7 : Exemple d'une convolution.

- **La couche de pooling (POOL)** :

L'opération de pooling consiste à réduire la taille spatiale des images (réduire le nombre de paramètres et de calculs dans le réseau), on améliore ainsi l'efficacité du réseau et on évite le sur-apprentissage, tout en préservant leurs caractéristiques importantes, elle est souvent placée entre deux couches de convolution [26].

Pour cela, on découpe l'image en cellules régulière, puis on garde au sein de chaque cellule la valeur maximale. En pratique, on utilise souvent des cellules carrées de petite taille pour ne pas perdre trop d'informations. Les choix les plus communs sont des cellules adjacentes de taille (2 × 2) qui ne se chevauchent pas.

La couche de pooling fonctionne indépendamment sur chaque tranche de profondeur de l'entrée et la redimensionne spatialement. Il existe plusieurs types de pooling :

- Le **"Max pooling**", est un processus de convolution où le Kernel (noyau) extrait la valeur maximale de la zone du feature map. C'est le type le plus utilisé car il est rapide à calculer (immédiat), et permet de simplifier efficacement la matrice.

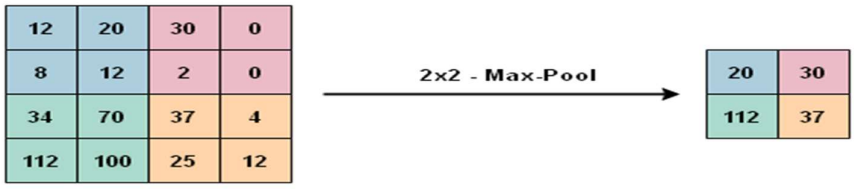

Figure III.8 : Représentation de maxpooling.

- Le **"mean pooling"** Appelé aussi average pooling, consiste à calculer la somme de toutes les valeurs et on divise par le nombre de valeurs. On obtient ainsi une valeur intermédiaire pour représenter ce lot d'éléments.

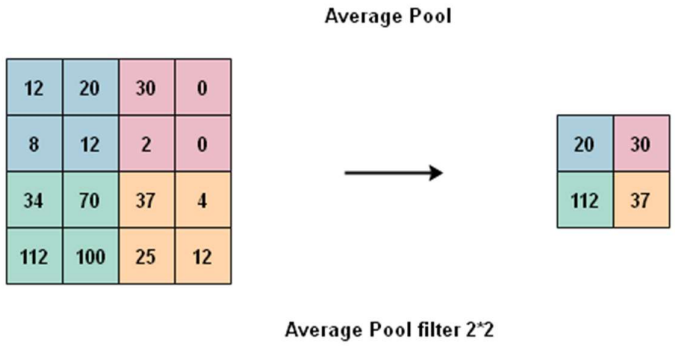

Figure III.9 : Représentation du meanpooling.

- **Le "sum pooling",** c'est la moyenne sans avoir divisé par le nombre de valeurs (on ne calcule que leur somme)

De manière générale, plus souvent on utilise le max-pooling, car il se distingue de mean-pooling sur les cas extrêmes, mais il est quasiment équivalent à mean-pooling dans les autres cas.

- **La couche de correction (ReLU) :**

ReLU (Rectifier Linear Units) est une fonction dite "rectifier" très utilisée en Deep Learning qui joue souvent le rôle de fonction d'activation [27], elle doit être appliquée à chaque élément après convolution, et remplace chaque valeur négative reçue en entrées par des zéros. Si cette fonction n'est

pas appliquée, la fonction créée sera linéaire et le problème XOR persiste puisque dans la couche de convolution, aucune fonction d'activation n'est appliquée.

ReLu est très utilisée dans les réseaux de neurones à convolution car il s'agit d'une fonction rapide à calculer : $f(u) = max(0,u)$, sa forme d'onde est représenté ci-dessous :

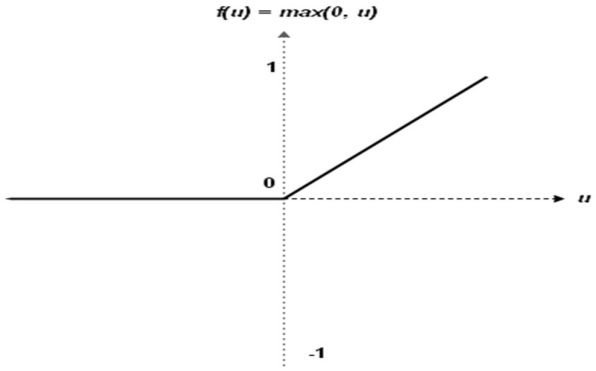

Figure III.10 : Fonction ReLU.

- **La couche "entièrement connectée" (fully-connected)**

Elle comporte toujours la dernière couche d'un réseau de neurones, convolutif ou non, elle n'est donc pas une caractéristique d'un CNN. De plus, Les neurones dans une couche entièrement connectée ont des connexions vers toutes les sorties de la couche précédente. Ce type de couche reçoit un vecteur en entrée et produit un nouveau vecteur en sortie. Pour cela, elle applique une combinaison linéaire puis éventuellement une fonction d'activation aux valeurs reçues en entrée.

La couche fully-connected (Figure III.11), renvoie toujours un vecteur, pour cela elle ne peut pas être placée avant une couche de pooling, puisque cette dernière doit recevoir une matrice 3D.

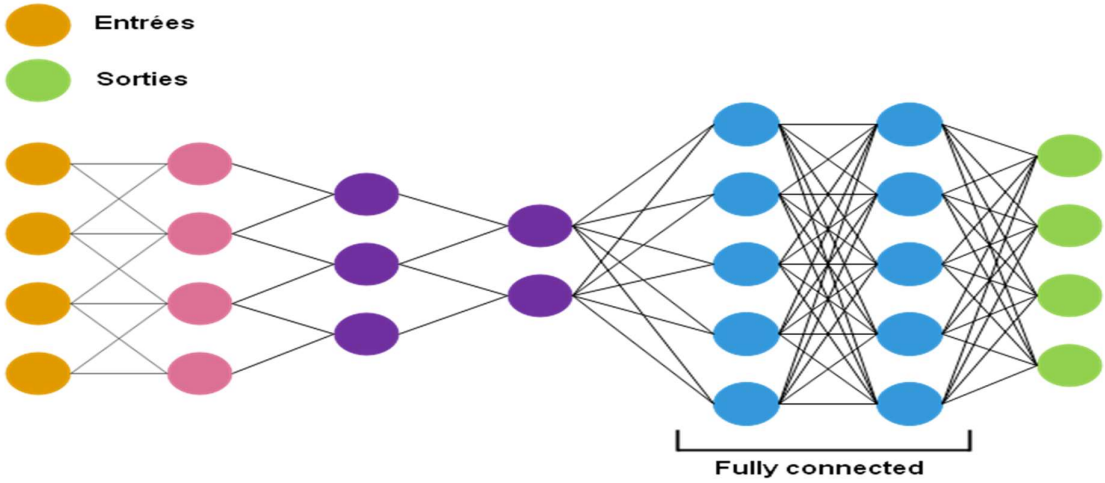

Figure III.11 : Représentation de la couche fully-connected.

En général, un réseau de neurones empile plusieurs couches de convolution et de correction ReLU, ajoute ensuite une couche de pooling (facultative), et répète ce motif

plusieurs fois ; puis, il empile des couches fully-connected. Plus il y a de couches, plus le réseau de neurones est « profond ».

Une règle à respecter est que la fonction de ReLu doit obligatoirement être appliquée après une étape de convolution afin d'avoir une réponse non-linéaire, mais le Pooling n'est pas obligatoire.

III.3.3 Apprentissage et test des réseaux de neurones convolutionnel

Avant son exploitation, le réseau doit pouvoir trouver la valeur des poids de chacune de ses connections. Cette valeur est déterminée durant la phase d'apprentissage.

L'apprentissage des réseaux de neurones est la procédure qui consiste à estimer les paramètres des neurones du réseau, afin que ce dernier remplisse au mieux la tâche qui lui est affectée [28]. Notre filtre neuronal, utilise un apprentissage supervisé qui consiste à présenter un échantillon en entrée du réseau, puis de comparer la sortie obtenue avec la sortie désirée.

Cela peut se faire par l'intermédiaire d'un algorithme d'optimisation basé sur la descente de gradient stochastique par lot (ou mini-batch).

Un élément important dans l'algorithme de descente de gradient est le pas d'apprentissage. Si le pas trop petit, l'algorithme devra effectuer un grand nombre d'itérations pour converger et prendra beaucoup de temps. Contrairement, si le pas est trop élevé, l'algorithme est divergé et s'éloigner ainsi de la bonne solution. Pour pallier ces problèmes, plusieurs variantes de l'algorithme de descente de gradient stochastiques ont été proposées. Parmi elles, on cite la méthode adaptative ADAM [29] que nous avons utilisée dans nos tests. Adam est l'un des algorithmes le plus récent et le plus fiable pour l'optimisation par descente de gradient.

Une fois l'apprentissage effectué, on passe à l'étape du test. La méthode de tests se déroule comme suit : Le réseau devra donc apprendre différents filtres neuronaux. L'avantage principal de ces filtres est qu'il pourra filtrer les signaux vocaux pour lesquelles on ne connaît que la parole bruitée et la parole propre.

III.4 Factorisation des matrices non négatives (NMF)

La factorisation des matrices non négatives (NMF) est l'une des techniques de réduction linéaire de dimensionnalité (LDR) largement utilisé pour l'analyse des données à haute dimension, car elle extrait automatiquement des caractéristiques significatives d'un ensemble de vecteurs de données afin de donner une représentation partielle de cet ensemble, à partir de quelques vecteurs de base seulement contenant des valeurs non négatives. Cette non-négativité rend les matrices résultantes plus faciles à inspecter.

La factorisation non négative de la matrice (NMF) est une technique qui permet la projection d'une matrice non négative donnée par $V = [v_{kl}] \in \mathbb{R}_+^{K \times L}$ sur un espace couvert par une combinaison linéaire d'un ensemble de vecteurs de base, c'est-à-dire $V \approx W H$, où $W = [w_{km}] \in \mathbb{R}_+^{K \times M}$ est appelé la matrice de base ou le dictionnaire, $H = [h_{ml}] \in \mathbb{R}_+^{M \times L}$ est appelée la matrice d'activation ou le coefficient, et M est le nombre de vecteurs de base, typiquement choisi tel que $M < min(K, L)$.

La factorisation est obtenue en minimisant une fonction de coût appropriée $\mathcal{D}(V, W H)$, telle que la distance euclidienne (EUC), la divergence de Kullback-Leibler (KL) ou la divergence d'Itakura-Saito (IS), respectivement [30] :

$$\mathcal{D}_{EUC}(V, WH) = \frac{1}{2}\sum_{k=1}^{K}\sum_{l=1}^{L}(v_{kl} - [WH]_{kl})^2 \quad \text{(III.3)}$$

$$\mathcal{D}_{KL}(V, WH) = \sum_{k=1}^{K}\sum_{l=1}^{L}(v_{kl} ln \frac{v_{kl}}{[WH]_{kl}} - v_{kl} + [WH]_{kl}) \quad \text{(III.4)}$$

$$\mathcal{D}_{IS}(V, WH) = \sum_{k=1}^{K}\sum_{l=1}^{L}(\frac{v_{kl}}{[WH]_{kl}} - ln \frac{v_{kl}}{[WH]_{kl}} - 1) \quad \text{(III.5)}$$

Où $[.]kl$ désigne la $(k, l) - ième$ entrée de son argument matriciel. Les solutions NMF peuvent être trouvées de manière itérative en utilisant les règles de mise à jour multiplicative « Multiplicative updates rule » (MU) correspondantes :

$$EUC: W \leftarrow W \otimes \frac{VH^T}{WHH^T} \qquad H \leftarrow H \otimes \frac{W^T V}{W^T WH} \quad \text{(III.6)}$$

$$KL: W \leftarrow W \otimes \frac{(V/(WH))H^T}{1_{KL}H^T} \qquad H \leftarrow H \otimes \frac{W^T(V/(WH))}{W^T 1_{KL}} \quad \text{(III.7)}$$

$$IS: W \leftarrow W \otimes \left(\frac{(V/(WH)^2)H^T}{(WH)^{-1}H^T}\right)^{1/2} \qquad H \leftarrow H \otimes \left(\frac{W^T(V/(WH)^2)}{W^T(WH)^{-1}}\right)^{1/2} \quad \text{(III.8)}$$

Où l'opération $\otimes$ désigne une multiplication élément par élément, la ligne de quotient et $/$ une division élément par élément, 1_{KL} est une matrice $K \times L$ dont toutes les entrées sont égales à un, $\leftarrow$ désigne un écrasement itératif, et les exposants de (III.6) sont calculés par éléments. Les indéterminations d'échelle dans W et H, qui apparaissent sous la forme d'un produit dans V,

peuvent être évitées en normalisant W à l'aide de la norme L_1 ou L_2 après avoir estimé W, et en calculant H par la suite, pour chaque itération.

La méthode la plus simple qui a été fréquemment utilisé dans le rehaussement de la parole pour obtenir une factorisation matricielle non négative est la distance euclidienne (EUC).

Supposons que Y_{k*t} est une matrice contenant les coefficients complexes de la TFD d'un signal où k et t sont les indices de la case de fréquence et du temps.

L'entrée de la NMF, V, est une transformation non négative de Y. L'un des choix populaires est $v_{kt} = |y_{kt}|$, c'est-à-dire que l'entrée de la NMF est le spectre d'amplitude du signal avec les vecteurs spectraux stockés par colonne.

Pour obtenir une décomposition non négative d'une matrice donnée, une fonction de coût (ou fonction de perte) est généralement définie et minimisée par la règle de mise à jour multiplicative en calculant la distance euclidienne selon l'équation (III.4). Ces règles de mise à jour peuvent être motivées en étudiant les conditions de Karush-Kuhn-Tucker. Une autre dérivation de cet algorithme peut être donnée en utilisant les méthodes du gradient fractionné « Split gradient methods » (SGM).

Les règles de mise à jour multiplicative sont un cas particulier des algorithmes de gradient-descent. Des approches plus efficaces ont également été utilisées pour obtenir des représentations NMF, ce qui peut également améliorer les performances dans une application spécifique.

Pour la plupart des applications, telles que la séparation des sources et le rehaussement de la parole, les performances peuvent être améliorées en imposant des contraintes, telles que la parcimonie (sparsity) et les dépendances temporelles. Dans ces scénarios, une fonction de coût régularisée est minimisée pour obtenir la représentation NMF :

$$(\boldsymbol{W}, \boldsymbol{H}) = argmin_{\boldsymbol{W},\boldsymbol{H}}\, D(\boldsymbol{V}||\boldsymbol{WH}) + \mu g(\boldsymbol{W}, \boldsymbol{H}) \qquad \text{(III.9)}$$

Avec $w_{ki} \geq 0, h_{it} \geq 0, \forall k, i, t$

où $g(.)$ est le terme de régularisation, et μ est le poids de régularisation. Un choix approprié de μ permet d'obtenir un bon compromis entre la fidélité et la satisfaction de la régularisation imposée.

Dans [31], la NMF a été largement utilisée comme technique pour estimer la parole propre à partir d'une observation bruitée. Comme précédemment, nous désignons la matrice des coefficients complexes de la DFT de la parole bruitée, de la parole propre et des signaux de bruit par Y, S et N, respectivement. Pour appliquer la NMF, nous obtenons d'abord une

transformation non négative de ces matrices, qui sont désignées par V, X et U, de telle sorte que $v_{kt} = |y_{kt}|^p$, $x_{kt} = |s_{kt}|^p$ et $u_{kt} = |n_{kt}|^p$ où $p = 1$ pour le spectre d'amplitude et $p = 2$ pour le spectre de puissance.

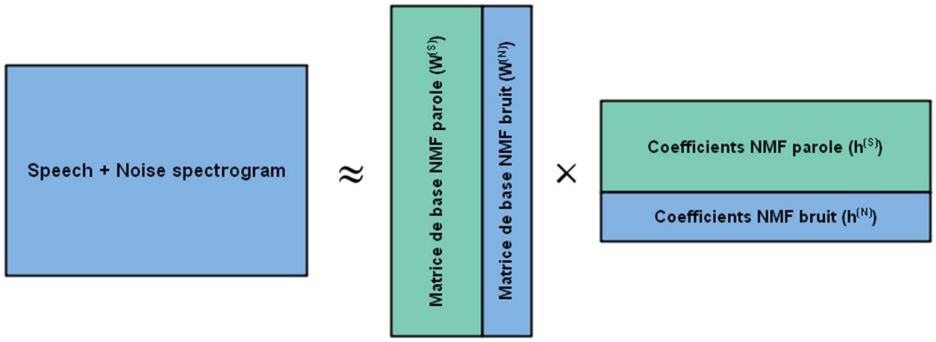

Figure III.12 : Application de la NMF sur la parole bruitée

Considérons une approche de débruitage supervisée des signaux basés sur le modèle NMF, l'un des modèles les plus populaires pour les signaux audios. La procédure générale fonctionne dans le domaine fréquentiel après la transformée de Fourier à court terme (STFT) et consiste en deux phases, comme le montre la figure (III.13) :

1. Apprentissage de modèles spectraux de source NMF à partir de quelques exemples.
2. Débruitage de la parole à l'aide des modèles pré appris.

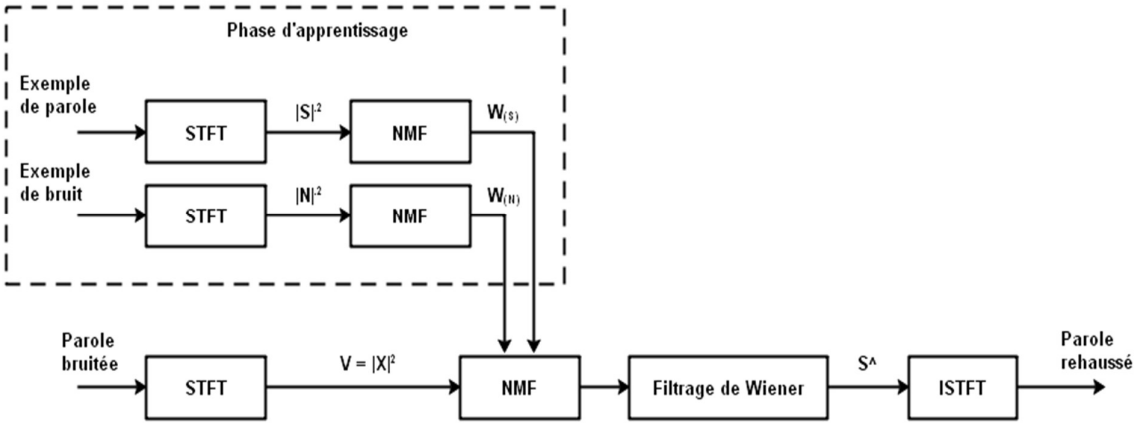

Figure III.13 : Procédure générale de rehaussement de la parole supervisé.

On prendre en considération un problème de séparation de signaux sur un seul canal avec 2 sources (parole et bruit). Soit $X \in \mathbb{C}^{F \times M}$, $S \in \mathbb{C}^{F \times M}$, et $N \in \mathbb{C}^{F \times M}$ les matrices à valeur complexe des coefficients de la transformée de Fourier à court terme (STFT) de la parole bruitée, du signal de parole, et du signal de bruit, respectivement, où F est le nombre de points de fréquence et M le nombre de trames temporelles. Le signal bruité s'écrit comme suit :

$$X = S + N \qquad \text{(III.10)}$$

On note $V = |X|^2$ le spectrogramme de puissance du signal.

La NMF vise à décomposer la matrice non négative V en deux matrices non négatives $W \in \mathbb{R}^{F\times K}$ et $H \in \mathbb{R}^{K\times M}$, respectivement, où $K < \min(F, M)$. Les paramètres W et H sont généralement initialisés avec des valeurs aléatoires non négatives et sont mis à jour de manière itérative par la règle des mises à jour multiplicative (MU).

Dans le cadre supervisé, le modèle spectral de la parole et du bruit, désigné respectivement par $W_{(S)}$ et $W_{(N)}$, est d'abord appris à partir des exemples d'entraînement correspondants. Supposons que l'on dispose de quelques exemples de signaux vocaux et de bruits. Étant donné $W_{(S)}^{p}$ et $W_{(N)}^{q}$ pour tous les exemples de parole $p = 1, \ldots, P$ et de bruit $q = 1, \ldots, Q$, respectivement, les modèles spectraux génériques pour la parole et le bruit sont construits comme suit génériques pour la parole et le bruit sont construits comme suit [32] :

$$\begin{aligned} W_S &= [W_{(S)}^{1}, \ldots, W_{(S)}^{P}] \\ W_N &= [W_{(N)}^{1}, \ldots, W_{(N)}^{Q}] \end{aligned} \tag{III.11}$$

Dans la phase de rehaussement de la parole, ce modèle spectral W est fixé tel que :

$$W = [W_{(S)}, W_{(N)}] \tag{III.12}$$

Et la matrice d'activation temporelle H est estimée via la règle MU. Notez que H est également partitionnée en blocs comme suit :

$$H = [H_{(S)}^{T}, H_{(N)}^{T}]^{T} \tag{III.13}$$

Où $H_{(S)}$ et $H_{(N)}$ désignent un bloc caractérisant les activations temporelles pour la parole et le bruit, respectivement.

Une fois les paramètres W et H sont obtenus, les coefficients STFT de la parole et du bruit sont calculés par filtrage de Wiener comme suit :

$$\hat{S} = \frac{W_{(S)}H_{(S)}}{WH} \odot X \tag{III.14}$$

$$\hat{N} = \frac{W_{(N)}H_{(N)}}{WH} \odot X \tag{III.15}$$

Enfin, les estimations de la source dans le domaine temporel sont obtenues par la STFT inverse comme c'est illustré sur la figure (III.13).

- **Filtrage de Wiener**

Le filtrage de Wiener est l'une des plus anciennes approches utilisées pour la réduction du bruit. Dans ce qui suit, nous examinons le filtre de Wiener dans le domaine de la transformée de Fourier discrète (DFT). Désignons respectivement par y, s et b les signaux de parole bruitée, parole propre et de bruit dans le domaine temporel. De même, l'indice de l'échantillon est désigné par n. Pour un bruit additif, le modèle du signal s'écrit comme suit :

$$y_n = s_n + b_n \tag{III.16}$$

Le filtrage de Wiener est un estimateur linéaire à erreur quadratique moyenne minimale (LMMSE) qui est un cas particulier du théorème bayésien de Gauss-Markov. En utilisant le filtre de Wiener, les coefficients DFT de la parole rehaussée sont estimés par un produit élément par élément des coefficients DFT du signal bruité $\boldsymbol{Y_{k,m}}$ et d'un vecteur DFT des poids $\boldsymbol{H_{k,m}}$:

$$\hat{S}_{k,m} = H_{k,m} \odot Y_{k,m} \tag{III.17}$$

où $\odot$ désigne un produit élément-par-élément. Pour obtenir le vecteur de poids $\boldsymbol{H_{k,m}}$, on minimise l'erreur quadratique moyenne (EQM) entre les signaux vocaux propres et estimés. En supposant que les différentes cases de fréquence sont indépendantes, nous pouvons minimiser l'EQM pour chaque composante fréquentielle k séparément :

$$H_{k,m} = argmin_{H_{k,m}} E(|S_{k,m} - \hat{S}_{k,m}|^2) \tag{III.18}$$

où l'espérance est calculée par rapport à la distribution conjointe f ($S_{k,m}$, $Y_{k,m}$). En fixant à zéro la dérivée partielle par rapport aux parties réelle et imaginaire de $H_{k,m}$, et en supposant que les signaux de parole et de bruit sont de moyenne nulle et non corrélés, les poids optimaux sont obtenus comme suit :

$$H_{k,m} = \frac{E(|S_{k,m}|^2)}{E\left(|S_{k,m}|^2\right) + E(|B_{k,m}|^2)} \tag{III.19}$$

Pour mettre en œuvre l'équation (III.19), les statistiques du bruit et de la parole sont généralement adaptées dans le temps pour obtenir une fonction de gain variable dans le temps. Cela permet de prendre en compte la non stationnarité des signaux. L'équation (III.19) est généralement mise en œuvre en fonction des rapport signal sur bruit a priori et a posteriori. A cette fin, le SNR a priori ($\xi_{k,m}$) et le SNR a posteriori ($\eta_{k,m}$) sont définis comme suit :

$$\xi_{k,m} = \frac{E(|S_{k,m}|^2)}{E(|B_{k,m}|^2)} \tag{III.20}$$

$$\eta_{k,m} = \frac{\left|Y_{k,m}\right|^2}{E(\left|B_{k,m}\right|^2)} \quad \text{(III.21)}$$

Le vecteur de poids optimal peut maintenant être écrit comme suit :

$$h_{k,m} = \frac{\xi_{k,m}}{\xi_{k,m} + 1} \quad \text{(III.22)}$$

Pour mettre en œuvre le filtre de Wiener, nous devons disposer d'une estimation du rapport signal/bruit a priori $\xi_{k,m}$. L'une des approches couramment utilisées pour estimer $\xi_{k,m}$ est connue sous le nom de méthode de décision dirigée, dans laquelle le SNR a priori est estimé comme suit :

$$\xi_{k,m} = max\left\{\xi_{min}, \alpha \frac{|\hat{S}_{k,m-1}|^2}{E(|B_{k,m-1}|^2)} + (1-\alpha) max\{\eta_{k,m} - 1, 0\}\right\} \quad \text{(III.23)}$$

où $\xi_{min} \approx 0{,}003$ est utilisé pour limiter au plus bas l'ampleur de la réduction du bruit.

III.5 Conclusion

Au cours de ce chapitre, nous avons traité les réseaux de neurones convolutionnels et la technique de factorisation matricielle non négative, en détaillant leur conception et leur utilisation dans le traitement de la parole. Leurs avantages étant la capacité d'apprentissage et de généralisation ainsi que leur architecture les rendant très intéressants pour leur implantation dans le traitement de signal vocale en temps réel.

Dans le chapitre qui suit nous allons procéder à l'application de ces deux méthodes (CNN, NMF) dans le rehaussement de la parole.

Chapitre IV

Implémentations, tests et résultats

IV.1 Introduction

L'intérêt principal des algorithmes de rehaussement de la parole est l'amélioration des caractéristiques du signal vocal qui sont la qualité et l'intelligibilité.

Les méthodes à base de réseaux de neurones et de la méthode de factorisation matricielle non négative constituent une alternative intéressante aux méthodes non supervisées comme celle de la soustraction spectrale de Berouti.

Ce chapitre présentera d'une part l'implémentation de l'application des différentes versions des réseaux de neurones ainsi que la technique de factorisation matricielle non négative. Et d'une autre part, les tests effectués afin de pouvoir évaluer l'amélioration de la qualité de signal vocal.

IV.2 Bases de données

Dans ce chapitre, nous allons utiliser deux bases de données, « **Noizeus** » et « **Mozilla Common Voice Dataset** ».

Noizeus est la base de données standard pour les travaux de rehaussement de la parole bruitée. Elle est déjà présentée dans le deuxième chapitre.

Mozilla Common Voice Dataset (disponible sur le site internet : http://ssd.mathworks.com/supportfiles/audio/commonvoice.zip) est une base de données qui appartient au projet « Common Voice » lancé par Mozilla au but de fournir une base de données gratuite pour les logiciels de traitement vocal. Ce projet est soutenu par des volontaires qui enregistrent des phrases types à l'aide d'un microphone et examinent les enregistrements des autres utilisateurs. Elle est disponible dans plus de 54 langues, on utilise la langue anglaise dans notre implémentation.

Notre version de Mozilla Common Voice Dataset contient 2000 fichiers audio avec un échantillonnage de 48kHz (rééchantillonnées à 8 kHz) que l'on utilise pour faire l'apprentissage de notre réseau de neurones, 400 fichiers pour le valider et d'autre 400 fichiers pour faire des tests. Pour réduire le temps d'apprentissage, on choisit un sous-ensemble de 200 fichiers seulement, 1% de ces fichiers sera consacré pour la validation du réseau.

Au cours de cette étude nous allons bruiter artificiellement les phrases obtenues des bases de données par différents bruits pris de la base de données AURORA (blanc, babble, aéroport) avec des rapports signal-sur-bruit de 0 dB et de 5 dB. Ainsi, pour chaque méthode et pour un type de bruit donné et un niveau de signal-sur-bruit bien défini, les phrases de parole propre et les phrases de parole bruitée correspondantes seront utilisées pour chaque mesure objective, ensuite une moyenne sera calculée.

IV.3 Conditions d'implémentation

Les simulations ainsi que tous les tests effectués sur les différents algorithmes traités sont les mêmes que ceux utilisés et déjà présentés dans le deuxième chapitre.

Dans cette partie, nous allons évaluer objectivement les performances des méthodes supervisées de rehaussement de la parole étudiées précédemment (CNN, FNN et NMF) avec une comparaison des résultats obtenus par la méthode non supervisée de la soustraction spectrale de Berouti qui utilise l'algorithme MCRA pour l'estimation du bruit.

Les signaux audios ont été sous-échantillonnés à 8 kHz. Les vecteurs spectraux ont été calculés en utilisant une transformée de Fourier à court terme (STFT) de 256 points (fenêtre de Hamming de 32 ms) avec un décalage de fenêtre de 64 points (8 ms). La résolution en fréquence était de 31,25 Hz (=4kHz/128) pour chaque composante fréquentielle. Les vecteurs d'amplitude STFT de 256 points ont été réduits à 129 points en supprimant la moitié symétrique. Pour les FNN, les paramètres d'entrée consistait en un vecteur d'amplitude STFT bruité (taille : 129×1, durée : 32ms). Pour le CNN, les paramètres d'entrée consistaient en 8 vecteurs d'amplitude STFT bruités consécutifs (taille : 129 × 8, durée : 100 ms). Les deux paramètres d'entrée ont été normalisées pour avoir une moyenne nulle et une variance unitaire.

En outre, la phase spectrale de l'observation bruitée a été utilisée pour effectuer la STFT inverse et récupérer la parole humaine. Pour tous les réseaux de neurones, les paramètres de sortie sont constitués d'un vecteur d'amplitude (taille : 129×1, durée : 32ms), qui a été normalisé pour avoir une moyenne nulle et une variance unitaire.

En ce qui concerne l'optimisation des réseaux de neurones utilisés, les poids des couches entièrement connectées et de convolution ont été initialisés comme dans la référence [33]. Les poids des couches entièrement connectées (FNN) ont été pré-appris à partir des réseaux de plus petite profondeur avec le même nombre de nœuds. Les couches de convolution sont obtenues après apprentissage, avec l'aide d'une couche de normalisation de lot « Batch normalization layer » ajoutée après chaque couche de convolution. Tous les réseaux ont été formés par rétropropagation avec optimisation par descente de gradient en utilisant l'algorithme ADAM [29] avec une taille de mini-batch de 64. Le taux d'apprentissage a commencé à partir de lr = 0,0015 avec $\beta1 = 0,9$, $\beta2 = 0,999$, et $\varepsilon = 1.0e\text{-}8$. Lorsque la perte de validation n'a pas diminué pendant plus de 4 itérations (epochs), le taux d'apprentissage a été réduit à lr/2, lr/3, lr/4, par la suite. L'apprentissage a été répété une fois de plus pour le FNN avec une régularisation l_2 ($\lambda = 10e\text{-}5$) qui a légèrement amélioré les performances.

IV.4 Résultats

Les différents paramètres des réseaux de neurones (CNN et FNN) ainsi que ceux de la méthode de la factorisation matricielle non négative ont été sélectionnés et choisis d'une manière à assurer de bons résultats.

- **Paramètres de la méthode du NMF**

  $K_{\text{parole propre}} = 32$ et $K_{\text{bruit}} = 16$

- **Paramètres du CNN et FNN**

Tableau IV.1 : Configuration des réseaux CNN et FNN.

	Configuration des couches	**Nombre des filtres**	**Taille des filtres**
CNN	(Conv, ReLU, BN) × 15, Conv	(18-30-8) × 5, 1	(9-5-9) × 5, 129
FNN	(FC, ReLU) × 3, FC	1024-1024-1024	

Les résultats de test des mesures (LLR, SNRseg, PESQ) sont représentés dans les tableaux présentés ci-dessous dans le cas d'utilisation de la base de données Noizeus. Les résultats obtenus précédemment dans le cas de la soustraction spectrale de Berouti avec l'algorithme MCRA pour l'estimation du bruit ont été ajoutés au tableau (IV.2) afin de faciliter la comparaison.

- **Résultats avec la base de données Noizeus**

Tableau IV.2 : Résultats de test des mesures (SNRseg, LLR, PESQ) pour des signaux dégradés par un bruit seul (Blanc, Babble, Aéroport) avec SNR = 0 dB (Noizeus).

	0 dB								
	Bruit blanc			**Bruit babble**			**Bruit d'aéroport**		
	SNRseg	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**
Dégradé	-5.0813	1.8022	1.5393	-4.6320	0.8950	1.7054	-4.4140	0.8644	1.7260
CNN	**0.8158**	**0.9616**	**2.2098**	**-2.8022**	**1.0016**	**1.9326**	**-2.1834**	**0.9014**	**1.8686**
FNN	-2.5557	1.4031	1.3618	-3.6456	1.3203	1.2924	-3.9006	1.3575	1.2415
Soustraction spectrale	-1.5984	1.7377	1.8184	-2.9547	1.1466	1.7151	-2.5488	1.0552	1.7251

Tableau IV.3 : Résultats de test des mesures (SNRseg, LLR, PESQ) pour des signaux dégradés par un bruit seul (Blanc, Babble, Aéroport) avec SNR = 5 dB (Noizeus).

	5 dB								
	Bruit blanc			**Bruit babble**			**Bruit d'aéroport**		
	SNRseg	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**
Dégradé	-2.3266	1.5450	1.7995	-1.7833	0.7152	2.0061	-1.6719	0.6911	1.7260
CNN	**3.4202**	**1.0094**	**2.5676**	**0.2424**	**1.005**	**2.2847**	**0.3558**	**0.7127**	**2.2542**
FNN	-1.7418	1.3655	1.504	-2.3123	1.2459	1.4975	-2.6461	1.2537	1.4274
Soustraction spectrale	1.0245	1.4335	2.1865	-0.2688	0.8724	2.0713	-0.1590	0.8403	2.0987

Les deux tableaux précédents présentent les performances de débruitage des réseaux FNN, CNN et la soustraction spectrale. Les résultats obtenus dans les deux cas du SNR global (0 dB et 5 dB), et pour tous les types de bruits montre que le réseau CNN assure des performances meilleures par rapport au réseau FNN en fonction des mesures de qualité objectives (SNRseg, LLR, PESQ) et une nette amélioration des performances par rapport à celles de la soustraction spectrale.

Les tests d'écoutes réalisés sur les signaux audios rehaussés par les méthodes précédentes confirment les mesures objectives présentées dans les tableaux et l'efficacité du réseau CNN.

En revanche, la taille du modèle du CNN était environ 68 fois plus petite que celle du FNN. Nous notons que le FNN a été optimisé pour avoir les plus petites architectures de réseau. Cette expérience valide le fait que le CNN nécessite un nombre bien moindre de paramètres par couche en raison de sa propriété de partage de poids, et peut pourtant atteindre des performances similaires ou supérieures à celles du FNN.

La figure suivante illustre un exemple de spectrogramme d'un fichier de parole propre, un spectrogramme de la parole bruitée par un bruit blanc avec un SNR = 0 dB et le spectrogramme de la parole rehaussée par le réseau CNN.

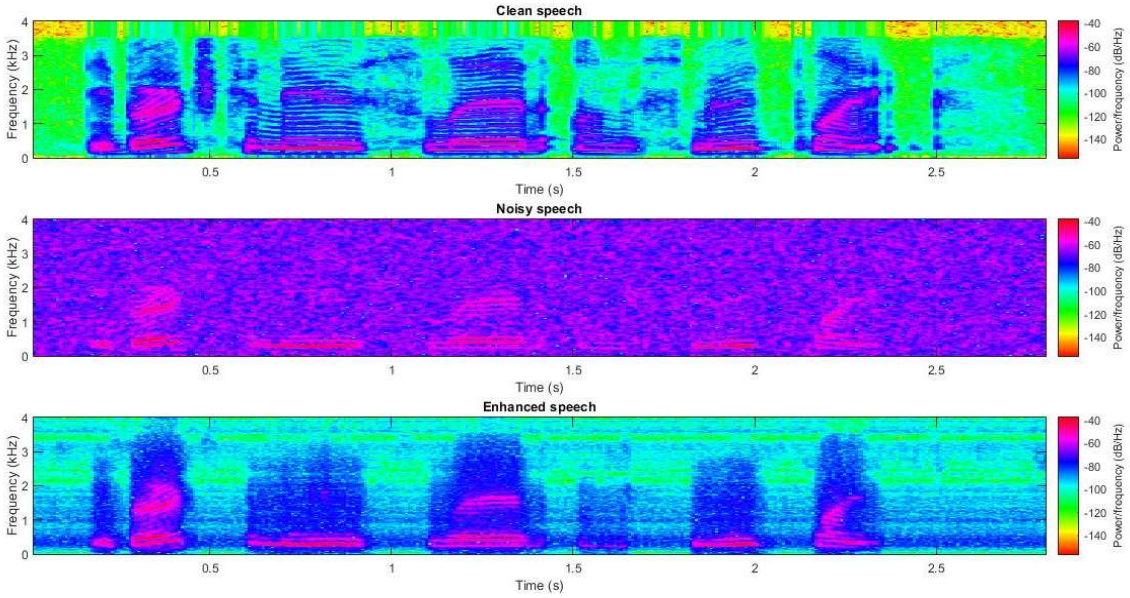

Figure IV.1 : Spectrogrammes de la parole propre, bruitée et rehaussée.

- **Résultats avec la base de données Mozilla common voice**

Dans ce qui suit, nous présenterons les résultats obtenus avec l'application des réseaux de neurones CNN et FNN pour le rehaussement de la parole dans le cas d'une base de données volumineuse. Le tableau (IV.4) dans le cas d'un SNR = 0 dB et le tableau (IV.5) pour le cas d'un SNR = 5 dB.

Tableau IV.4 : Résultats de test des mesures (SNRseg, LLR, PESQ) pour des signaux dégradés par un bruit seul (Blanc, Babble, Aéroport) avec SNR = 0 dB (Base de données Mozilla common voice).

	0 dB								
	Bruit blanc			**Bruit babble**			**Bruit d'aéroport**		
	SNRseg	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**
Dégradé	-5.0813	1.8022	1.5393	-4.6320	0.8950	1.7054	-4.4140	0.8644	1.7260
CNN	**1.1297**	1.5408	**2.1216**	**-2.9096**	**1.1420**	**1.9833**	**-3.2463**	**1.0666**	**1.8566**
FNN	-2.5720	**1.4000**	1.3856	-3.6589	1.3542	1.3146	-3.9008	1.3695	1.2542

Tableau IV.5 : Résultats de test des mesures (SNRseg, LLR, PESQ) pour des signaux dégradés par un bruit seul (Blanc, Babble, Aéroport) avec SNR = 5 dB (Base de données Mozilla common voice).

	5 dB								
	Bruit blanc			**Bruit babble**			**Bruit d'aéroport**		
	SNRseg	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**	**SNRseg**	**LLR**	**PESQ**
Dégradé	-2.3266	1.5450	1.7995	-1.7833	0.7152	2.0061	-1.6719	0.6911	2.0213
CNN	**2.2951**	1.6181	**2.2081**	**-0.2061**	**0.9262**	**2.2572**	**-0.4841**	**0.8992**	**2.1552**
FNN	0.6923	**1.2000**	2.0432	-1.9076	1.0745	1.7452	-1.8914	1.0471	1.7997

Les mêmes remarques observées dans le cas de la base de données Noizeus restent valables dans le cas de cette volumineuse base de données Mozilla common voice qui est largement utilisée dans l'application des réseaux de neurones dans le traitement du signal vocal. Le réseau CNN confirme sa supériorité par rapport aux autres versions des réseaux de neurones.

- **Résultats de l'application du NMF avec la base de données Noizeus**

Nous avons choisi de se limiter à deux mesures objectives (PESQ, LLR) seulement pour représenter les résultats de simulation de la méthode NMF car la mesure SNRseg n'est pas corrélée avec les tests subjectifs et les tests d'écoute dans ce cas, comme nous avons pu remarquer durant les différents tests effectués. De plus, dans la majorité des travaux de recherche qui appliquent la méthode NMF utilisent les mesures SDR (Source to Distorsion Ratio) ou SIR (Source to Interference Ratio) à la place du SNRseg. Le tableau suivant présente les résultats obtenus dans les deux cas (SNR = 0 dB et SNR = 5 dB).

Tableau IV.6 : Résultats de test des mesures (LLR, PESQ) pour des signaux dégradés par un bruit seul (Blanc, Babble, Aéroport) avec SNR = 0 dB (Base de données Noizeus).

	0dB					
	Bruit blanc		**Bruit babble**		**Bruit d'aéroport**	
	LLR	**PESQ**	**LLR**	**PESQ**	**LLR**	**PESQ**
Dégradé	1.8022	1.5393	0.8950	1.7054	0.8644	1.7260
CNN	1.1005	**2.2304**	1.0102	**1.9881**	**0.8518**	**1.9123**
NMF	**0.8327**	2.1227	**0.9035**	1.7119	0.8656	1.7224

Tableau IV.7 : Résultats de test des mesures (LLR, PESQ) pour des signaux dégradés par un bruit seul (Blanc, Babble, Aéroport) avec SNR = 5 dB (Base de données Noizeus).

	0dB					
	Bruit blanc		**Bruit babble**		**Bruit d'aéroport**	
	LLR	**PESQ**	**LLR**	**PESQ**	**LLR**	**PESQ**
Dégradé	1.5450	1.7995	0.7152	2.0061	0.6911	1.7260
CNN	1.0928	2.2878	0.8958	**2.2587**	0.7566	**2.2890**
NMF	**0.6790**	**2.3993**	**0.7140**	2.0091	**0.6879**	2.0092

Les performances des méthodes basées sur le NMF sont comparables aux performances obtenues avec le réseau CNN en termes des deux mesures LLR et PESQ.

IV.5 Conclusion

Nous avons présenté les résultats obtenus lors de l'application des réseaux FNN, CNN et la méthode NMF ainsi qu'une comparaison avec les résultats obtenus avec la méthode de la soustraction spectrale.

Conclusion générale

Conclusion générale

L'objectif du projet qui nous a été proposé est d'étudier les performances des différents types et architectures de réseaux de neurones dans le domaine du rehaussement de la parole, et les comparer avec les performances des méthodes de base comme la soustraction spectrale.

Premièrement, nous avons étudié le principe de fonctionnement des réseaux de neurones en expliquant ces différents types et architectures, sa procédure de développement et ses avantages et inconvénients.

La définition de la soustraction spectrale, ses principes de calculs, ses limitations, ses différents algorithmes y compris l'implémentation de ces algorithmes et les résultats obtenus ont été interprété au cours du deuxième chapitre.

Nous avons étudié au cours de ce travail deux algorithmes d'estimation de bruit : Hirsch et MCRA. Ces deux algorithmes ont été combinés avec un système de réduction de bruit basé sur la soustraction spectrale de puissance développée par Berouti.

Les mesures objectives et les tests d'écoute nous ont montré que l'algorithme MCRA est plus efficace par rapport à celle de Hirsch pour les trois types de bruit.

Par la suite, Nous avons implémenté trois systèmes de rehaussement de la parole : un réseau de neurones convolutionnel (CNN), un réseau de neurone complètement connecté (FNN) et un système basé sur la factorisation matricielle non négative. Ces derniers ont été appris en utilisant deux bases de données consacrées pour le traitement vocal.

Les mesures de performances de ces trois systèmes avec différents niveaux du SNR et différents types de bruit montrent clairement que les réseaux de neurones convolutionnels sont les plus efficaces pour le rehaussement de la parole à condition que l'apprentissage de ces réseaux doit être fait en utilisant des bases de données très riche et variant. Les réseaux FNN présentent un bon choix pour la réduction de bruit comparativement à la méthode du NMF qui a donné des résultats comparables par rapport à ceux obtenus par la méthode de la soustraction spectrale.

Il est très important de noter que malgré leurs efficacités, ces algorithmes et systèmes nécessitent des améliorations notables.

Il a été démontré que les approches supervisées produisent des signaux vocaux améliorés de meilleure qualité par rapport aux méthodes non supervisées, ceci est attendu car plus d'informations préalables sont fournies au système dans ces cas et les modèles considérés sont formés pour chaque type spécifique de signaux. L'information préalable requise sur le type de bruit (et l'identité du locuteur dans certains cas) peut être donnée par l'utilisateur, ou peut être obtenue en utilisant un schéma de classification intégré.

Bibliographie et Webographie

[1] S.F. Boll, *"Suppression of acoustic noise in speech using spectral subtraction,"* IEEE Trans. Acoust., Speech, Signal Processing, vol. 27, pp. 113-120, 1979.

[2] M. Berouti, R. Schwartz and J. Makhoul, *"Enhancement of speech corrupted by acoustic noise,"* in Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 208-211, Washington, DC, USA, 1979.

[3] https://machinelearningknowledge.ai/brief-history-of-deep-learning (consulté le 06/07/2022)

[4] G. Dreyfus et al., *Réseaux de neurones : Méthodologie et applications,* Eyrolles, Paris, France, 417p, 2004.

[5] Khelaf Souaad, *"Analyse du signal ultrasonore par les réseaux de neurones,"* Mémoire de Master, Université Mohammed Seddik Ben Yahia, Jijel, Algérie, 2012.

[6] Fandi Tadj Eddine, *"Simulation d'un classifieur neural sur FPGA,"* Mémoire de Master, Université Abou Bakr Belkaïd, Tlemcen, Algérie, 2013.

[7] Karine Volpi, *"Cours sur les réseaux de neurones, "* https://nanopdf.com/download/karine-volpi-1-reseau-de-neurones_pdf (consulté le 06/07/2022)

[8] Redjradj Amine, Oulebsir Melisa, *"Deep Learning pour La Reconnaissance des caractères manuscrits,"* Mémoire de Master, Université Abderrahmane Mira, Bejaïa, Algérie, 2021.

[9] Mokri Mohammed Zakaria, *"Classification des images avec les réseaux de neurones convolutionnels,"* Mémoire de Master, Université Abou Bakr Belkaid, Tlemcen, Algérie, 2017.

[10] Lyes Abdelli et Rafik Kebiche, *"Reconnaissance automatique de types de modulation à base de réseau de neurones, "* Mémoire de Master, Université Abderrahmane Mira, Bejaïa, Algérie, 2021.

[11] Lotfi Merad, *"Modélisation et optimisation des réseaux d'antennes imprimées par les réseaux de neurones et l'algorithme génétique, "* Mémoire de Master, Université Abou Bakr Belkaid, Tlemcen, Algérie, 2001.

[12] Chergui Laid, *"Débruitage de la parole par de méthodes basées sur les transformées discrètes, "* Thèse de Doctorat, Université Ferhat Abbas - Setif 1 -, Algérie, 2017.

[13] Anis Ben Aicha, *"Réduction du bruit musical et évaluation de la qualité des signaux débruités par approches perceptuelles, "* Thèse de Doctorat en technologie de l'information et de la communication, École supérieure des communications de Tunis, Tunis, 2010.

[14] N. Virag, *"Single channel speech enhancement based on masking properties of the human auditory system,"* IEEE Trans. Speech and Audio Processing, vol. 7, no. 2, pp. 126-137, 1999.

[15] P. Vary *"Noise suppression by spectral magnitude estimation-mechanism and theoretical limits,"* Signal Processing, vol. 8, pp. 387-400, 1985.

[16] H. G. Hirsch and C. Ehrlicher, *"Noise estimation techniques for robust speech recognition,"* Proc. 20th IEEE. ICASSP-95, Detroit, MI, USA, pp. 153-156, May 8-12, 1995.

[17] I. Cohen and B. Berdugo, *"Noise estimation by minima controlled recursive averaging for robust speech enhancement,"* IEEE Signal Process, Lett. vol. 1, pp. 12-15, January 2001.

[18] S. R. Quanckenbush, T. P. Barnwell III and M. A. Clements, Objective measures of speech quality, Prentice Hall, Englewood Cliffs, New Jersey, USA, 1988.

[19] P. Mermelstein, *"Evaluation of segment SNR measure as an indicator of the quality of ADPCM coded speech,"* J. Acoust. Soc. America, pp. 1664-1667, 1979.

[20] J.H.L. Hansen and B.L. Pellom *"An effective quality évaluation protocol for speech enhancement algorithms "* in Int. Conf. on Spoken Language Processing (ICSLP), Sydney, Australia, 1998.

[21] D.H. Klatt, *"Prediction of perceived phonetic distance from critical-band spectra: a first step,"* in Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1278–1281, Paris, France, 1982.

[22] ITU-T P.862, *"Perceptual evaluation of speech quality (PESQ), and objective method for end-to-end speech quality assessment of narrowband téléphone networks and speech codecs"*, 2000.

[23] ITU-T Recommandation P.862. *"Perceptual evaluation of speech quality (PESQ) : An objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs, "* 2001.

[24] Recommandation UIT-T P.800, *"Méthodes d'évaluation subjective de la qualité de transmission, "* 1996.

[25] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, *"Gradient-based learning applied to document recognition,"* Proceedings of the IEEE, vol. 86, no. 11, 1998.

[26] https://datasciencetoday.net/index.php/en-us/deep-learning/173-les-reseaux-de-neurones-convolutifs (consulté le 06/07/2022)

[27] Vinod Nair and Geoffrey E.Hinton, *"Rectified linear units improve restricted Boltzmann machines,"* 27th International Conference on Machine Learning, Toronto, Canada, pp. 807-814, June 2010.

[28] Chikh Mohammed Tahar, *"Amélioration des images par un modèle de réseau de neurones (Comparaison avec les filtres de base),"* Mémoire de Master en Informatique, Université Abou-Bakr Belkaid, Tlemcen, Algérie, 2011.

[29] D.P. Kingma and J.L. Ba., *"ADAM* : *A method for stochastic optimization,"* ICLR, San Diego, California, USA, 2015.

[30] Hanwook Chung, *"Speech enhancement using training-based non-negative matrix factorization techniques,"* Master Thesis, Department of Electrical & Computer Engineering McGill University Montreal, Canada, July 2018.

[31] N. Mohammadiha, *"Speech enhancement using non-negative matrix factorization and hidden Markov models,"* PHD Thesis, Communication Theory Laboratory School of Electrical Engineering KTH Royal Institute of Technology, Stockholm, Sweden, 2013.

[32] Hien-Thanh Duong et al. *"Speech enhancement based on nonnegative matrix factorization with mixed group sparsity constraint,"* 6th ACM International Symposium on Information and Communication Technology, pp. 247-251, Hanoi, Vietnam, Dec 2015.

[33] Xavier Glorot and Yoshua Bengio, *"Understanding the difficulty of training deep feedforward neural networks.,"* in Aistats, 2010, vol. 9, pp. 249–256, 2010.

Abstract

Noise reduction is a very important task in speech signal processing. Spectral speech enhancement approaches are generally successful in reducing background noise. However, their major drawback is the appearance of residual noise with a musical character that can be more annoying in some cases. To overcome this problem, we studied the performance of different types and architectures of neural networks (CNN, FNN), as well as the non-negative matrix factorization (NMF) method and compared them with the performance of basic methods such as spectral subtraction.

Keywords: Speech enhancement, speech signal processing, noise reduction, CNN, non-negative matrix factorization, NMF, FNN, spectral subtraction, neural networks.

ملخص

يعد تقليل الضوضاء خطوة مهمة جدا في معالجة إشارات الكلام. فبالرغم من نجاح الطرق الطيفية في تحسين الكلام و تقليل الضوضاء بشكل مناسب الا ان عيبها الرئيسي هو ظهور ضوضاء متبقية لها طابع موسيقي و التي يمكن ان تكون مزعجة اكثر في بعض الحالات. للتغلب على هذه المشكلة، قمنا بدراسة الأنواع و البنيات المختلفة للشبكات العصبية (CNN, FNN) و كذلك طريقة تحليل المصفوفات غير السالبة (NMF) و مقارنتها بأداء الطرق الأساسية مثل طريقة الطرح الطيفي.

كلمات مفتاحية : تحسين اكلام، معالجة إشارات الكلام، تقليل الضوضاء، CNN، FNN، تحليل المصفوفات غير السالبة، NMF، الطرح الطيفي، الشبكات العصبية.

Résumé

La réduction du bruit est une tâche très importante en traitement des signaux vocaux. Les approches spectrales de rehaussement de la parole réussissent généralement à réduire convenablement le bruit de fond. Toutefois, leur inconvénient majeur est l'apparition d'un bruit résiduel ayant un caractère musical qui peut être dans certain cas plus gênant. Pour pallier ce problème, nous avons étudié les performances des différents types et architectures de réseaux de neurones (CNN, FNN), ainsi que la méthode de la factorisation des matrices non négatives (NMF) et les comparer avec les performances des méthodes de base comme la soustraction spectrale.

Mots-clés : Rehaussement de la parole, traitement des signaux vocaux, réduction de bruit, CNN, factorisation des matrices non négatives, NMF, FNN, soustraction spectrale, réseaux de neurones.