UNIVERSITÉ DE SHERBROOKE
Faculté de génie
Département de génie électrique et de génie informatique

REHAUSSEMENT DE LA PAROLE À L'AIDE D'UN RÉSEAU DE NEURONES À DÉCHARGES

Mémoire de maitrise
Spécialité : génie électrique

Abir RIAHI

Sherbrooke (Québec) Canada

Septembre 2023

MEMBRES DU JURY

Éric PLOURDE

Directeur

Philippe GOURNAY

Rapporteur

Jean ROUAT

Évaluateur

RÉSUMÉ

Le rehaussement de la parole est essentiel pour garantir la fiabilité des outils de communication ou la robustesse des systèmes de reconnaissance vocale. Bien que les réseaux neuronaux conventionnels, connus sous le nom de réseaux neuronaux artificiels (*Artificial Neural Network* - ANN), aient démontré des performances remarquables dans ce domaine, leur utilisation requiert une puissance de calcul considérable et engendre des coûts énergétiques élevés. Ces coûts sont dus à plusieurs facteurs tels que la taille du réseau, le volume de l'ensemble des données utilisé, et le nombre d'itérations nécessaires pour l'entraînement. Ce projet de recherche propose une approche de rehaussement de la parole à l'aide d'un réseau de neurones à décharges (*Spiking Neural Network* - SNN) basé sur une architecture U-Net. Les SNN sont adaptés au traitement de données avec une dimension temporelle, telle que la parole, et sont connus pour leur mise en œuvre économe en énergie sur des processeurs neuromorphiques. Par conséquent, les SNN constituent des candidats intéressants pour des applications en temps réel sur des dispositifs aux ressources limitées. L'objectif principal de ce travail est de développer un modèle basé sur un SNN présentant des performances comparables à celles d'un modèle basé sur un ANN pour le rehaussement de la parole. L'entraînement du SNN proposé s'effectue en utilisant une optimisation basée sur des gradients de substitution. L'évaluation des performances du modèle se fait à l'aide de tests objectifs perceptuels, en prenant en compte différents rapports signal sur bruit et conditions de bruit réelles. Les résultats obtenus démontrent que le modèle proposé surpasse la solution de référence du défi de suppression de bruit profond neuromorphique d'Intel. De plus, il se distingue également par rapport à plusieurs approches non neuromorphiques de l'état de l'art. En outre, il atteint des performances acceptables par rapport à un modèle ANN présentant une architecture similaire. En conclusion, ce travail met en évidence la promesse des SNN en tant qu'alternative performante aux ANN pour le rehaussement de la parole.

Mots-clés : rehaussement de la parole, réseaux de neurones à décharges, gradient substitué

À ma famille, vous êtes ma force, ma source d'inspiration et ma raison de persévérer.

REMERCIEMENTS

Je tiens à exprimer ma profonde gratitude envers toutes les personnes qui ont contribué à la réalisation de ce travail.

En premier lieu, j'aimerais remercier mon directeur de recherche, M. Éric Plourde, professeur au département de génie électrique et de génie informatique de l'Université de Sherbrooke, et directeur du programme de génie informatique, pour son encadrement précieux, ses conseils avisés et son soutien constant tout au long de ce parcours académique.

Je souhaite également exprimer ma reconnaissance envers les membres du laboratoire NECOTIS pour les discussions enrichissantes.

Je tiens enfin à remercier les membres de mon jury, M. Jean Rouat et M. Philippe Gournay, pour leur évaluation bienveillante de mon travail.

Enfin, je suis profondément reconnaissante envers ma famille et mes amis pour leur soutien constant, leur motivation et leur amour inconditionnel. Sans eux, ce travail n'aurait pas été possible.

À toutes ces personnes, je tiens à dire merci du fond du cœur.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

LISTE DES ACRONYMES

Acronyme	Définition
BERT	*Bidirectional Encoder Representations from Transformers*
BSA	*Bens Spiker Algorithm*
CASA	*Computational Auditory Scene Analysis*
CNN	*Convolutional Neural Network*
DEMAND	*Diverse Environments Multi-channel Acoustic Noise Database*
DNS	*Deep Noise Suppression*
DNSMOS	*Deep Noise Suppression Mean Opinion Score*
GAN	*Generative Adversarial Network*
GPT-3	*Generative Pre-trained Transformer 3*
GPU	*Graphics Processing Unit*
GRU	*Gated Recurrent Unit*
HMM	*Hidden Markov Models*
LIF	*Leaky Integrate-and-Fire*
LLM	*Large Language Model*
LPS	*Logarithmic Power Spectrum*
LSD	*Log-Spectral Distance*
LSTM	*Long-Short Term Memory*
MLP	*Multilayer Perceptron*
N-DNS	*Neuromorphic Deep Noise Suppression*
NMF	*Nonnegative Matrix Factorization*
PESQ	*Perceptual Evaluation of Speech Quality*
ReLU	*Rectified Linear Unit*
RNN	*Recurrent Neural Network*
SDNN	*Sigma-Delta Neural Network*
SEGAN	*Speech Enhancement Generative Adversarial Network*
SNN	*Spiking Neural Network*
SNR	*Signal-to-Noise Ratio*
STDP	*Spike Timing-Dependent Plasticity*
STFT	*Short-Time Fourier Transform*
STOI	*Short-Time Objective Intelligibility*
VOCODER	*Voice Coder*
VODER	*Voice Operating Demonstrator*

CHAPITRE 1

Introduction

1.1 Mise en contexte et problématique

La performance des outils de communication vocale est devenue un critère essentiel pour les utilisateurs de divers appareils mobiles tels que les tablettes, les téléphones portables, les montres connectées ou les lunettes intelligentes. Cependant, la qualité de la communication est souvent altérée par la présence de bruit ambiant. Aussi, les applications de traitement de la parole telles que la reconnaissance, ou la compression de la parole, exigent une qualité de signal d'entrée optimale pour fonctionner de manière efficace. Ainsi, pour maintenir l'intelligibilité et la qualité de la parole, il est primordial d'utiliser des algorithmes de rehaussement de la parole. Ces algorithmes sont conçus pour réduire le bruit de fond des signaux vocaux bruités tout en préservant la qualité et l'intelligibilité de la parole elle-même. De nos jours, l'utilisation de tels algorithmes est devenue cruciale pour améliorer les plateformes de communication à distance telles que Zoom, Slack, Microsoft Teams, ou Discord, en raison de la croissance significative des activités en ligne, notamment le travail, l'enseignement et la santé.

De nombreuses techniques de rehaussement de la parole ont été développées afin d'obtenir des systèmes de traitement de la parole ou de communication vocale plus efficaces et résilients aux interférences sonores. Parmi les approches classiques, la soustraction spectrale [7] consiste à supprimer le bruit en soustrayant une estimation du spectre de bruit du spectre de la parole. Une autre méthode est le filtrage de Wiener [48], qui consiste à minimiser l'erreur quadratique moyenne entre le signal propre et celui filtré. Cependant, ces méthodes traditionnelles présentent des limites en raison de l'incertitude associée à l'estimation du spectre de bruit, ce qui peut entraîner une distorsion de la parole. De plus, ces techniques ont du mal à traiter efficacement des situations complexes comprenant différents types de bruits et de variations dynamiques.

Les avancées récentes dans le domaine de l'apprentissage profond ont favorisé une utilisation accrue des réseaux de neurones artificiels dans de nombreux domaines, notamment le diagnostic médical, le traitement d'images, le traitement automatique du langage naturel, la conduite autonome et la prédiction des indices boursiers, [77]. Ces progrès ont conduit à l'intégration de nombreuses applications basées sur des algorithmes d'apprentissage pro-

fond dans notre vie quotidienne. Parmi les exemples marquants figurent les assistants personnels tels que Siri d'Apple, Cortana de Microsoft, Alexa d'Amazon, l'assistant Google et Bixby de Samsung. Parallèlement, plusieurs applications de rehaussement de la parole basées sur des réseaux de neurones artificiels ont également émergé, à l'instar de Krisp, NVIDIA RTX Voice et Mozilla RNNoise.

Toutefois, les architectures profondes de réseaux de neurones artificiels requièrent souvent des capacités de calcul et de stockage importantes, ce qui les rend difficiles à intégrer dans des systèmes embarqués de dispositifs mobiles et portables disposant de ressources matérielles limitées [19]. Aussi, l'entraînement des réseaux de neurones artificiels impliquant des volumes massifs de données nécessite une quantité considérable d'énergie, ce qui peut avoir des répercussions significatives sur l'environnement et l'empreinte carbone. À titre d'exemple, une étude menée à l'Université du Massachusetts à Amherst a révélé que l'entraînement du modèle de langage BERT (*Bidirectional Encoder Representations from Transformers*) [18] sur un processeur graphique émet autant de dioxyde de carbone qu'un vol transaméricain [80].

La taille et la complexité des modèles d'apprentissage profond continuent de croître, augmentant considérablement leur consommation d'énergie, soulevant ainsi des préoccupations quant à leur impact environnemental. En 2020, la société OpenAI a développé un grand modèle de langage (*Large Language Model* - LLM), GPT-3 (*Generative Pre-trained Transformer* 3) [12], qui compte 175 milliards de poids, soit plus de 115 fois le nombre de poids de BERT (1,5 milliard de poids). Gopher [66], un LLM développé par DeepMind, compte 280 milliards de poids et surpasse GPT-3 en termes de performances pour certaines tâches. On peut donc penser que la consommation énergétique de ces LLM dépasse largement celle de BERT qui était déjà problématique. Il est donc crucial de trouver des moyens d'améliorer l'efficacité énergétique des algorithmes d'apprentissage automatique.

Les réseaux de neurones à décharges (*Spiking neural network* - SNN) [36] présentent une architecture extrêmement prometteuse, grâce à leur efficacité énergétique remarquable et leur plausibilité biologique. Contrairement aux réseaux de neurones conventionnels, les calculs dans un SNN ne sont pas effectués simultanément pour tous les neurones. Les neurones communiquent plutôt entre eux par le biais de décharges envoyées à travers des synapses excitatrices ou inhibitrices. Ces décharges, également connues sous le nom de potentiels d'action, sont des formes d'onde particulières, d'une certaine durée temporelle, émises par les neurones à des instants particuliers. L'efficacité énergétique des SNN réside dans le fait que les décharges pour chaque neurone sont parcimonieuses et ne se produisent pas simultanément.

La simulation d'un modèle basé sur les SNN sur une architecture computationnelle standard de von Neumann est réalisable, mais ne permet pas d'atteindre une optimisation maximale des performances. Dans cette optique, plusieurs entreprises et centres de recherche universitaire se concentrent actuellement sur le développement d'architectures neuromorphiques en temps réel. Parmi les exemples notables figurent Loihi d'Intel [16], TrueNorth d'IBM [50], NeuroGrid de l'Université de Stanford [4], SpiNNaker de l'Université de Manchester [27] et BrainScaleS de l'Université d'Heidelberg [49].

L'utilisation de processeurs neuromorphiques suscite un intérêt considérable en raison de leur parallélisme massif et de leur efficacité énergétique. Par exemple, la puce Loihi peut être jusqu'à 1000 fois plus rapides que les processeurs traditionnels, tout en consommant jusqu'à 10 000 fois moins d'énergie [53].

Bien que les réseaux de neurones à décharges offrent des avantages potentiels en termes d'efficacité énergétique, il est essentiel de les situer de manière appropriée dans le contexte de la recherche en traitement de la parole, en particulier dans le domaine du rehaussement de la parole. La littérature montre que de nombreuses approches ont été explorées pour aborder le traitement de la parole corrompue, parfois même plus complexe que le rehaussement. Parmi ces approches, on cite le masquage, qui puise son inspiration dans la perception auditive, ainsi que des méthodes statistiques basées sur les caractéristiques spectrales des signaux propres et bruités, telles que les différences statistiques et les estimations de spectres.

Le masquage a été initialement développé dans le contexte de l'analyse auditive par Bregman [9], conformément à la méthodologie connue sous le nom de CASA (*Computational Auditory Scene Analysis*) [11]. Par la suite, cette méthode a été avec succès intégrée dans des systèmes de reconnaissance de la parole [14], ainsi que dans des approches neuronales, notamment celles de Wang [94] et Pichevar [63], parmi d'autres. Ces contributions ont pavé la voie à diverses solutions basées sur des réseaux de neurones, conventionnels ou à décharges, souvent caractérisées par des équations différentielles non linéaires. Cette approche pourrait sembler en sommeil actuellement, en raison des réussites de l'apprentissage profond, mais une combinaison des approches de masquage et de modification globale du spectre, comme celle que nous explorons dans ce mémoire, pourrait offrir des avantages significatifs, en particulier lorsque les données sont limitées.

Dans ce mémoire, nous avons choisi de nous concentrer sur l'approche de modification globale du spectre, principalement pour des raisons de comparabilité avec les avancées actuelles en apprentissage profond. Cette approche statistique nous permet d'obtenir de

meilleurs rapports signal-à-bruit, même si elle peut entraîner une perte de qualité de compréhension de la parole par rapport à l'approche de masquage. De plus, étant donné notre intérêt spécifique pour le bruit additif, nous estimons qu'il n'est pas nécessaire de développer des systèmes plus complexes inspirés de la perception dans notre contexte.

1.2 Question de recherche

Le présent projet de recherche vise à répondre à la question de recherche suivante :

Est-il possible de développer un algorithme de rehaussement de la parole en utilisant un réseau de neurones à décharges qui serait capable de produire une performance équivalente ou supérieure à celle des algorithmes actuels ?

1.3 Objectifs du projet de recherche

Afin de répondre à la question de recherche énoncée précédemment, l'objectif de ce projet de recherche est donc de développer un modèle de rehaussement de la parole en utilisant des réseaux de neurones à décharges et de le comparer à des algorithmes actuels. L'approche proposée vise à obtenir des performances équivalentes, voire supérieures, à celles obtenues par des architectures qui ne font pas usage de cette technique.

1.4 Contributions originales

Le présent projet de recherche a pour objectif principal de proposer un modèle de rehaussement de la parole en utilisant des réseaux de neurones à décharges qui atteindra une performance comparable ou supérieure à celle des architectures non basées sur les SNN. En comparaison avec les réseaux de neurones conventionnels, les SNN présentent de nombreux avantages, notamment leur capacité à traiter l'information de manière asynchrone et leur faible consommation d'énergie. À notre connaissance, il existe peu de travaux de rehaussement de la parole en utilisant les SNN. Aussi, ce travail trouve son originalité à travers plusieurs aspects :

- Tout d'abord, l'approche proposée consiste à entraîner un modèle capable d'estimer le spectre de puissance logarithmique de la parole propre à partir de celui de la parole bruitée. Cette méthode représente une avancée significative dans le domaine du rehaussement de la parole. En comparaison avec l'approche traditionnelle du masquage, qui implique la multiplication du spectre de la parole bruitée par un masque, notre approche statistique présente plusieurs avantages. Elle offre la possibilité d'améliorer les rapports signal-à-bruit, même si cela peut occasionnellement entraîner une légère dégradation de la compréhensibilité de la parole par rapport

à l'approche du masquage. De plus, notre méthode s'intègre plus aisément dans le contexte de la recherche en apprentissage profond. En effet, elle repose sur des principes statistiques plus intuitifs et transparents que les mécanismes complexes du masquage. Par ailleurs, étant donné que notre étude se focalise exclusivement sur la problématique du bruit additif, elle ne nécessite pas le développement de systèmes plus complexes inspirés de la perception, simplifiant ainsi notre démarche de recherche. À notre connaissance, il s'agit de la première tentative d'exploiter un modèle basé sur des réseaux de neurones à décharges pour le rehaussement de la parole, en adoptant une stratégie d'estimation directe du spectre plutôt que l'estimation d'un masque appliqué ultérieurement au spectre de la parole bruitée.

- Dans un deuxième temps, nous adoptons une architecture profonde, ce qui n'a pas été exploré dans les travaux précédents. Cette décision est étroitement liée à notre approche d'estimation du spectre, car l'utilisation d'une architecture profonde favorise une estimation statistique plus précise et riche en informations. De plus, cette approche nous permet de réaliser une comparaison pertinente avec plusieurs modèles de réseaux de neurones conventionnels de l'état de l'art.
- Dans un troisième temps, ce travail adopte une méthode de codage direct où la première couche du SNN apprend simultanément à convertir l'entrée en décharges et à supprimer le bruit. Ce choix est appuyé par notre désir de maintenir une cohérence avec les avancées actuelles en matière d'apprentissage profond. En effet, cette approche est étroitement alignée sur les méthodologies prédominantes dans le domaine de l'apprentissage profond, ce qui simplifie considérablement la comparaison de notre modèle avec d'autres approches existantes. En outre, cette décision présente l'avantage supplémentaire de ne pas exiger le développement d'un système substantiellement divergent, comme cela aurait été nécessaire en cas d'adoption d'une approche basée sur le masquage.
- Dans un quatrième temps, nous effectuons l'optimisation des paramètres neuronaux, à savoir les constantes de temps et le seuil de décharge, en plus des poids lors de l'entraînement du réseau de neurones.
- Dans un cinquième temps, une étude d'ablation est réalisée pour évaluer l'effet de différents paramètres sur les performances du modèle proposé.
- En dernier lieu, une évaluation comparative entre le modèle de réseau de neurones à décharges proposé et un modèle de réseau de neurones conventionnels d'architecture similaire est réalisée.

Ce travail de recherche a mené à la rédaction d'un article de conférence qui a été accepté [69], soulignant ainsi la contribution de cette étude au domaine du rehaussement de la parole en utilisant les réseaux de neurones à décharges.

1.5 Plan du document

Après ce premier chapitre d'introduction, la suite du document est structurée comme suit :

- Le chapitre 2 présente une revue de la littérature portant sur les différents éléments important pour la compréhension du mémoire, dont : la problématique du rehaussement de la parole, les différents modèles communs de réseaux de neurones conventionnels ou à décharges, ainsi que les approches basées sur les réseaux de neurones pour le rehaussement de la parole.
- Le chapitre 3 présente une description de l'aspect théorique du modèle proposé et l'architecture du réseau de neurones à décharges implémenté.
- Le chapitre 4 présente les conditions expérimentales d'entraînement et d'évaluation de la méthode proposée.
- Le chapitre 5 présente une étude de l'effet des différents paramètres de l'approche proposée et par la suite une analyse des résultats obtenus.
- Finalement, le chapitre 6 présente la conclusion, un retour sur les contributions et les travaux futurs.

CHAPITRE 2

Revue de la littérature

Le présent chapitre propose une revue de littérature approfondie en ce qui concerne le rehaussement de la parole, les réseaux de neurones conventionnels et à décharge, ainsi que leurs utilisations pour le rehaussement de la parole. La première partie formalise la problématique du rehaussement de la parole et présente quelques approches classiques. La deuxième partie est dédiée à une étude des différentes architectures de réseaux de neurones artificiels, notamment les réseaux de neurones à décharges. Enfin, la dernière partie du chapitre propose une synthèse des différentes approches proposées dans la littérature pour le rehaussement de la parole en utilisant des réseaux de neurones conventionnels ou à décharges.

2.1 Rehaussement de la parole

La qualité du signal de la parole peut être considérablement altérée par différents types de bruits, interférences, échos et réverbérations présents dans l'environnement acoustique. Ces dégradations peuvent réduire l'intelligibilité du signal, ce qui peut entraîner une baisse des performances des systèmes de traitement de la parole ou de communication vocale. Ainsi, les algorithmes de rehaussement de la parole sont essentiels pour de nombreuses applications [3], notamment la compression de la parole, la reconnaissance vocale et la synthèse vocale. Ces algorithmes sont soumis à un compromis entre la réduction du bruit et la préservation de l'intelligibilité du signal de parole propre.

2.1.1 Formulation du problème

Le rehaussement de la parole est un domaine de recherche important qui vise à améliorer la qualité des signaux de parole qui ont été altérés par des sources de bruit environnemental. Plusieurs modèles ont été proposés pour modéliser différents types de bruits (p.ex. bruit additif, bruit convolutif). Ce projet de recherche se concentre sur le rehaussement de parole perturbée par un bruit additif. En général, le bruit additif est causé par des perturbations externes présentes dans l'environnement. Dans le modèle du bruit additif, le signal de parole bruité observé est considéré comme la somme du signal de parole propre et d'une composante de bruit additive.

Mathématiquement, un signal de parole bruité x à l'instant n est défini comme suit :

$$x[n] = s[n] + d[n] \tag{2.1}$$

où n est l'indice de l'échantillon dans le temps discret, s correspond au signal de parole propre et d correspond au signal de bruit additif. L'objectif d'un algorithme de rehaussement de la parole est de calculer une estimation du signal de parole propre $\hat{s}[n]$ étant donné le signal de parole bruité observé $x[n]$. Cette estimation doit être aussi proche que possible du signal de parole propre original, tout en éliminant autant que possible l'impact du bruit de fond indésirable.

2.1.2 Approches classiques

Au fil des dernières décennies, divers techniques ont été développées pour le rehaussement de la parole dans un contexte mono-canal ou multicanal. Parmi ces premières approches, on retrouve, entre autres, la soustraction spectrale [7] et l'utilisation d'un filtre de Wiener [48]. La soustraction spectrale consiste à estimer le spectre de bruit et à le soustraire du spectre du signal de parole bruité pour obtenir une estimation du signal de parole propre. La méthode de filtrage de Wiener utilise un estimateur linéaire pour minimiser l'erreur quadratique moyenne entre le signal de parole rehaussé et le signal de parole propre.

Par la suite, la recherche s'est orientée vers des approches statistiques telles que l'algorithme d'erreur quadratique moyenne minimale [23]. Avec l'avènement de l'apprentissage automatique, plusieurs méthodes ont également été proposées en utilisant des approches classiques basées sur l'apprentissage automatique, telles que les modèles de Markov cachés (*Hidden Markov Models* - HMM) [17], la factorisation matricielle non négative (*Non-negative Matrix Factorization* - NMF) [52] et la transformée en ondelettes [62].

Ces techniques ont abouti à des résultats encourageants en matière de réduction du bruit. Cependant, leur performance est limitée par la nature du bruit et des propriétés statistiques du signal de la parole, notamment dans des scénarios comprenant des bruits non stationnaires ou des situations de cocktail party [10].

L'émergence de l'apprentissage profond a profondément influencé l'orientation de la recherche dans plusieurs domaines. Aujourd'hui, plusieurs architectures de réseaux de neurones artificiels font partie des meilleurs modèles de rehaussement de parole [101].

La section suivante se concentrera sur une revue approfondie des différentes architectures de réseaux de neurones. Par la suite, une section subséquente se concentrera sur les approches qui utilisent des réseaux de neurones pour le rehaussement de la parole.

2.2 Réseaux de neurones

2.2.1 Réseaux de neurones biologiques

Les neurones sont des cellules hautement spécialisées du système nerveux, responsables de la réception et de la transmission de l'information. Ils sont composés d'un corps cellulaire, de dendrites et d'un axone. En état de repos, un neurone est caractérisé par une différence de potentiel électrique à travers sa membrane cellulaire appelée potentiel de repos. La communication entre les neurones est effectuée par des jonctions appelées synapses, qui se distinguent en deux types : synapses chimiques (prédominantes) et synapses électriques. Lorsqu'un neurone reçoit un signal, il génère une impulsion électrique appelée potentiel d'action, qui se propage le long de l'axone et déclenche la libération de neurotransmetteurs chimiques (synapse chimique), permettant ainsi la communication entre les neurones.

Les travaux révolutionnaires d'Alan Hodgkin et d'Andrew Huxley sur les potentiels d'action des neurones ont permis de comprendre comment ces impulsions électriques sont générées et propagées le long de l'axone. En 1963, ils ont reçu le prix Nobel de médecine pour leurs travaux [31]. Leurs expériences consistaient à insérer des électrodes minuscules dans les axones des neurones de calmar géant et à mesurer les changements de potentiel électrique lors de la stimulation du neurone. Ils ont découvert que le potentiel d'action est un événement bref, tout ou rien, causé par le mouvement des ions à travers la membrane du neurone.

Un mécanisme fondamental de l'apprentissage du cerveau est la plasticité synaptique, c'est-à-dire la capacité des synapses à renforcer ou affaiblir l'efficacité d'une connexion [37]. Le cerveau humain est un système de traitement de l'information incroyablement complexe et sophistiqué qui a évolué au cours de millions d'années pour permettre un large éventail de capacités cognitives, comme la perception, l'attention, la mémoire et la prise de décision. Cette complexité a inspiré les chercheurs en intelligence artificielle à développer des modèles informatiques plus réalistes biologiquement et capables d'apprendre de leur expérience. Les réseaux de neurones artificiels sont des modèles mathématiques inspirés de la biologie, qui permettent de modéliser le comportement des neurones biologiques et la communication synaptique.

2.2.2 Réseaux de neurones artificiels conventionnels

Depuis plus de 50 ans, les mathématiciens ont cherché à imiter les mécanismes du cerveau humain en intégrant la biologie dans des modèles informatiques [72]. Le but initial de ces travaux était de développer des programmes informatiques capables d'apprendre par eux-

mêmes. Depuis, les réseaux de neurones ont connu un essor remarquable et sont utilisés dans de nombreux domaines en raison de leur grande adaptabilité.

Les réseaux de neurones artificiels apprennent, en général, à l'aide d'un grand nombre d'exemple ce qui nécessite, d'une part, un très grand nombre de données et, d'autre part, une grande capacité de calcul. La croissance exponentielle du volume de données, ainsi que le développement de processeurs de plus en plus puissants, ont été des facteurs clés qui ont permis aux réseaux de neurones conventionnels de produire de meilleurs résultats que les techniques classiques de traitement de données.

Dans les sous-sections suivantes, nous présentons plusieurs architectures de réseaux de neurones conventionnels. Nous examinons les avantages et les limites de chaque architecture. Ces architectures comprennent le perceptron simple, le perceptron multicouche, les réseaux de neurones convolutionnels et les réseaux de neurones récurrents.

Perceptron simple

Le perceptron est l'un des premiers modèles d'apprentissage automatique. Il a été inventé par Frank Rosenblatt, un psychologue américain, en 1957, au sein du laboratoire d'aéronautique de l'Université Cornell [72]. C'est un modèle d'apprentissage simple qui est composé de plusieurs entrées et d'une seule sortie. Il a été principalement conçu pour le problème de classification binaire, c'est-à-dire, séparation de deux classes.

La figure 2.1 présente un exemple de perceptron à une seule couche et deux entrées. Il est formé d'une couche d'entrée pour lire les données et d'une couche de sortie.

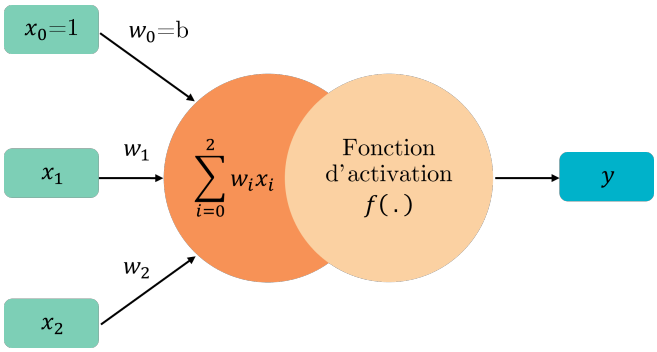

FIGURE 2.1 Architecture d'un perceptron à une seule couche et deux entrées.

L'architecture présentée est composée d'une couche unique de neurones, comprenant des entrées x_1 et x_2 avec leurs poids associés, les coefficients synaptiques w_1, w_2, ainsi qu'une entrée constante x_0 égale à 1 avec un poids b appelé le biais. La sortie y est calculée en appliquant une fonction d'activation f à la somme pondérée des entrées. La fonction d'activation sert à introduire une non-linéarité dans le modèle.

L'équation générale calculée par un perceptron ayant le vecteur d'entrée $\mathbf{x} = [x_1, \ldots, x_n]$, le vecteur de poids $\mathbf{w} = [w_1, \ldots, w_n]$, le biais b et la valeur de sortie y est :

$$y = f(\sum_{i=1}^{n} w_i x_i + b) \tag{2.2}$$

Pour résoudre un problème de classification binaire, la sortie est une valeur binaire qui sépare les données en deux classes. Le perceptron de Frank Rosenblatt n'utilisait que la fonction d'Heaviside. La fonction d'Heaviside est définie comme suit :

$$H(x) = \begin{cases} 0 & \text{si} \quad x < 0 \\ 1 & \text{si} \quad x \geq 0 \end{cases} \tag{2.3}$$

Ultérieurement, plusieurs autres fonctions d'activation ont été proposées, notamment :

- La fonction sigmoïde : cette fonction prend en entrée une valeur réelle et la transforme dans la plage [0, 1]. Elle est définie par :

  $$\sigma(x) = \frac{1}{1 - \exp(-x)} \tag{2.4}$$

  La fonction sigmoïde peut être utilisée dans le cas de la régression logistique pour prédire la probabilité d'appartenance à une classe.
- La fonction tangente hyperbolique : cette fonction prend en entrée une valeur réelle et la transforme dans la plage [-1, 1]. Elle est définie par :

  $$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \tag{2.5}$$

  La fonction tangente hyperbolique peut être utilisée pour la régression logistique, ainsi que pour les problèmes de régression.
- La fonction Unité Linéaire Rectifiée (*Rectified Linear Unit* - ReLU) : cette fonction prend en entrée une valeur réelle et la transforme dans la plage [0, $+\infty$]. Elle est définie par :

  $$f(x) = \max(0, x) \tag{2.6}$$

La fonction ReLU est souvent utilisée dans les réseaux de neurones profonds, non seulement en raison de sa rapidité de calcul, mais surtout en vertu de sa capacité à favoriser une rétropropagation efficace du gradient à travers toutes les couches du réseau. Contrairement à d'autres fonctions d'activation de type compressif, qui ont tendance à rapidement atténuer le gradient, entravant ainsi sa rétropropagation, la ReLU assure une propagation stable et efficiente du gradient, en faisant ainsi un choix essentiel dans la conception des réseaux de neurones profonds.

- La fonction softmax : cette fonction prend en entrée un vecteur de nombres réels et la transforme dans la plage [0, 1[. Elle est définie par :

$$\sigma(\mathbf{z})_i = \frac{\exp(z_i)}{\sum_{j=1}^{K} \exp(z_j)}, \text{ pour } i = 1, \ldots, K \text{ et } \mathbf{z} = [z_1, \ldots, z_K] \tag{2.7}$$

Dans le cas d'un problème de classification multi-classe, l'architecture du perceptron est modifiée. La couche de sortie doit avoir autant de neurones que le nombre de classes. La fonction softmax peut être utilisée pour prédire la probabilité d'appartenance à chaque classe.

Malheureusement, l'application du perceptron à une seule couche était limité. En effet, il ne pouvait séparer que les classes linéairement séparables et ne disposait que de la fonction d'Heaviside [51]. Les limites technologiques, en raison de la faible puissance de calcul, et théoriques ont entraîné la stagnation de la recherche sur cette approche pendant de nombreuses années.

Perceptron multicouche

Tel que mentionné précédemment, le perceptron à une seule couche ne permet pas de résoudre un problème de classification non linéaire. Pour pallier à cette limitation, Marvin Minsky et Seymour Papert ont proposé dans les années quatre-vingt une architecture plus complexe appelée perceptron multicouche (*Multilayer Perceptron* - MLP) [51]. Le perceptron multicouche est mieux adapté que le perceptron simple pour résoudre des problèmes non linéaires grâce à l'utilisation d'une fonction d'activation non-linéaire dont la dérivée est calculable. Cette caractéristique permet l'utilisation du MLP pour des tâches de classification ou de régression selon la fonction d'activation employée.

Un perceptron multicouche est constitué d'une couche d'entrée, une couche de sortie et une ou plusieurs couches intermédiaires dites « cachées ». Chaque neurone d'une couche (à l'exception des neurones de la couche de sortie) est connecté à tous les neurones de la couche suivante. La figure 2.2 illustre un exemple du modèle de perceptron multicouche

possédant une couche d'entrée avec n neurones, deux couches cachées avec respectivement 4 et 3 neurones, et une couche de sortie avec 4 neurones.

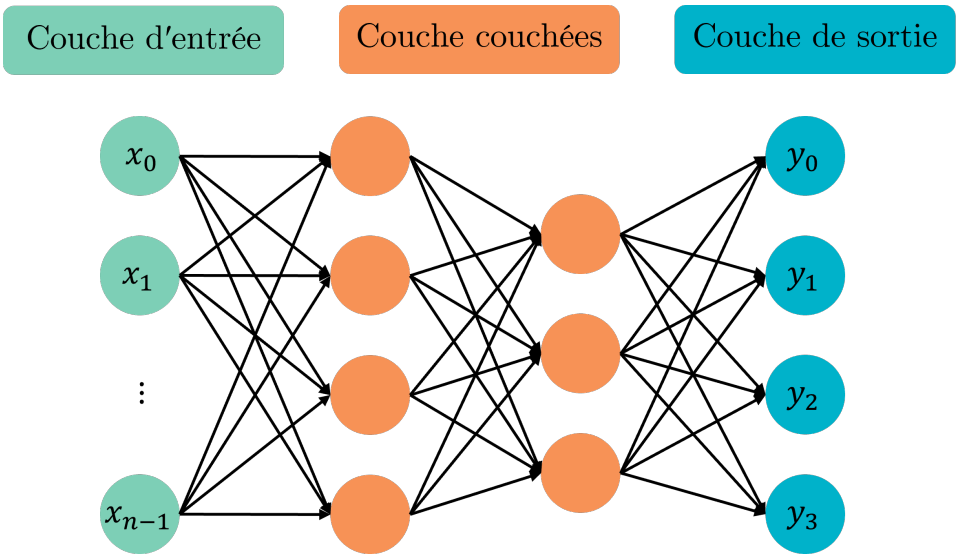

FIGURE 2.2 Architecture d'un perceptron multicouches.

La flexibilité de l'architecture du perceptron multicouche en fait un outil de prédiction puissant pour modéliser des relations complexes entre des données d'entrée et des sorties attendues. Cette architecture a été largement utilisée dans différents domaines tels que la reconnaissance de la parole, la vision par ordinateur, la finance et la biologie [6]. Sa popularité est également attribuée à un algorithme d'apprentissage appelé "algorithme de rétropropagation de l'erreur", qui sera présenté dans la section suivante.

Apprentissage par rétropropagation de l'erreur

Généralement, l'entraînement d'un réseau de neurones peut être réalisé selon différentes approches, dont l'apprentissage supervisé ou non supervisé. Cette section se concentre sur l'apprentissage supervisé. Cette approche nécessite l'utilisation d'un ensemble de données annotées, où chaque donnée est associée à un résultat attendu. L'objectif de l'entraînement d'un réseau de neurones est de trouver la combinaison optimale de poids qui minimise l'erreur entre les prédictions du réseau et les résultats attendus.

L'introduction de l'algorithme de rétropropagation de l'erreur a révolutionné l'entraînement des réseaux de neurones. L'algorithme de rétropropagation de l'erreur a été initialement introduit dans les années soixante, formalisé de manière standard par Paul Werbos en 1974 [97] et popularisé par les travaux de Rumelhart en 1986 *et al.* [75].

Le processus d'entraînement avec rétropropagation de l'erreur comprend deux étapes principales. Tout d'abord, le réseau effectue une propagation avant, où les activations des neu-

rones sont calculées de la première couche jusqu'à la dernière. Ensuite, une propagation arrière permet de calculer le gradient de l'erreur pour chaque poids, de la dernière couche vers la première. Cette étape est appelée "rétropropagation". Par la suite, un algorithme d'optimisation, tel que l'algorithme de descente de gradient, est utilisé pour mettre à jour les poids du réseau en fonction du gradient calculé. Ce gradient permet de déterminer la direction de la modification à apporter à chaque poids pour minimiser la fonction d'erreur. Ce processus itératif d'ajustement des poids par rétropropagation de l'erreur permet au réseau de converger vers une configuration optimale.

Réseaux de neurones convolutifs

L'utilisation d'architectures de réseaux de neurones pleinement connectées entraîne un grand nombre de paramètres à entraîner. Dans le contexte de classification d'images, la figure 2.3 illustre un exemple d'un réseau de neurones artificiel pleinement connecté composé d'une couche à entraîner, avec MNIST [8], une base de données d'images de chiffres écrits à la main.

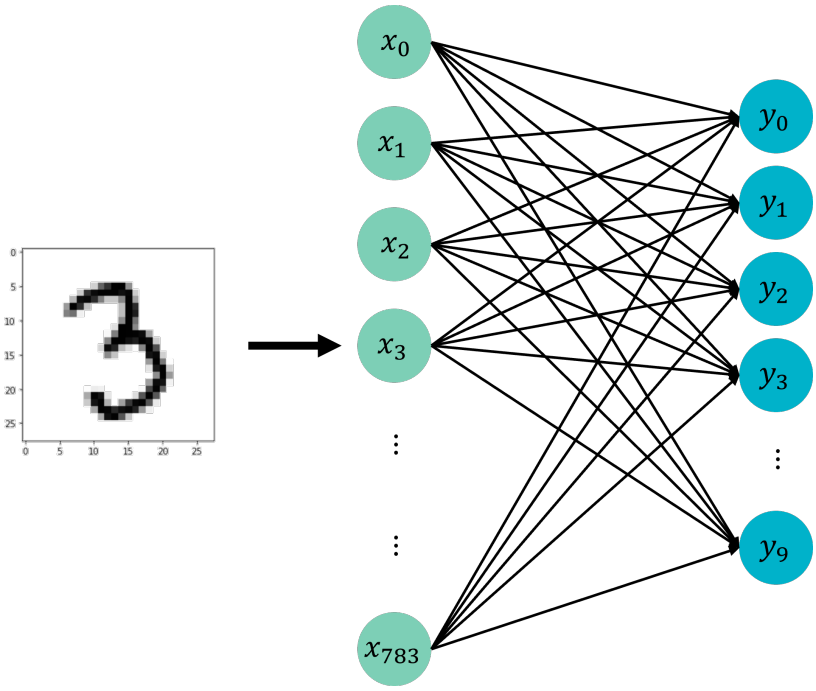

FIGURE 2.3 Réseau de neurones à 1 couche pour MNIST.

En utilisant une image d'entrée de 28×28 pixels (soit 784 neurones d'entrée) et 10 classes de sortie (soit 10 neurones de sortie), ce réseau de neurones nécessite l'entraînement de 7850 paramètres. Ainsi, il est indispensable d'utiliser une architecture qui réduit le nombre de connexions tout en conservant une performance satisfaisante. Une des solutions proposées est le partage des paramètres pour pouvoir implémenter des réseaux de neurones plus

profonds. C'est pourquoi il est courant de recourir aux réseaux de neurones convolutifs (*Convolutional Neural Network* - CNN).

La figure 2.4 présente l'architecture d'un réseau de neurones convolutif et ses principales composantes.

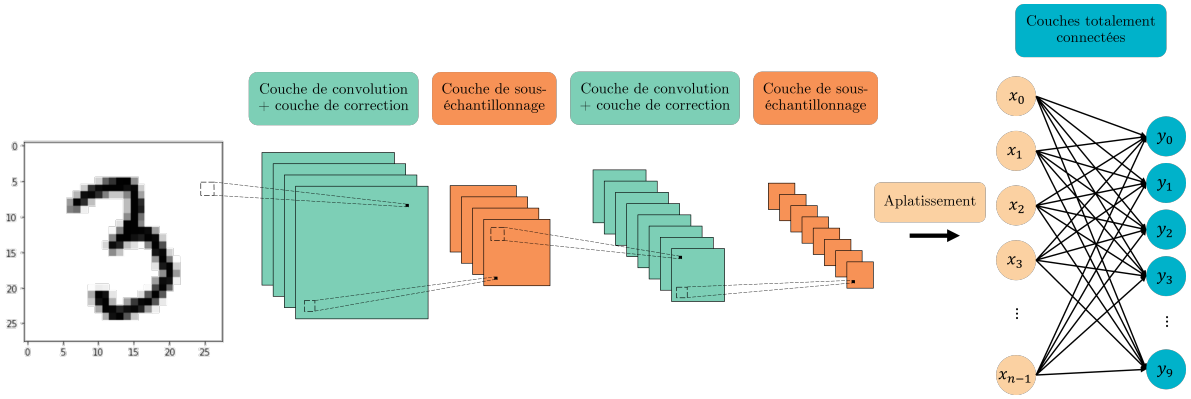

FIGURE 2.4 Architecture d'un réseau de neurones convolutif.

Les CNN utilisent de nombreuses copies identiques du même neurone et sont formés par un empilement de couches consécutives pour extraire les caractéristiques discriminantes de la classe d'appartenance de l'image.

Les couches de traitement d'un CNN sont :

- **Couche de convolution** : La couche de convolution est la composante clé d'un CNN. Elle joue un rôle primordial dans l'analyse des images d'entrée pour identifier la présence d'un ensemble de caractéristiques. Ce processus est accompli par l'opération de filtrage par convolution, qui consiste à calculer la somme pondérée des éléments d'une fenêtre glissante de l'image d'entrée en utilisant un noyau de convolution. La taille du noyau de convolution correspond au nombre de neurones associés à un même champ récepteur (zone analysée dans l'image d'entrée). Lors du processus de convolution, la fenêtre glissante se déplace en fonction d'un paramètre appelé "pas d'avancement" (*stride*) qui détermine le nombre d'éléments parcourus à chaque itération. La sortie de cette couche est appelée carte d'activation (*feature map*).

  La figure 2.5 présente un exemple d'opération de convolution d'une image d'entrée I de taille (6, 6), correspondant à une matrice de dimensions 6×6, avec un noyau de convolution K de taille (3, 3), soit une matrice de dimensions 3×3. Le noyau se déplace avec un pas de déplacement (1, 1), ce qui signifie qu'il se déplace d'un seul élément dans chaque direction. Cette opération génère une carte d'activation de taille (4, 4), soit une matrice de taille 4×4.

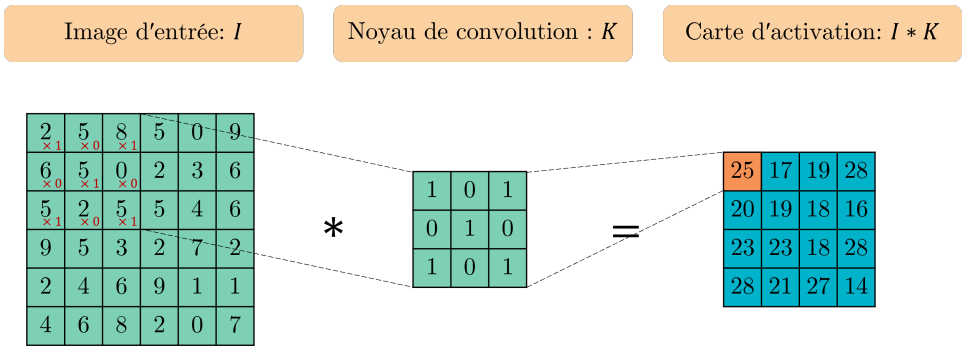

FIGURE 2.5 Exemple illustrant l'opération de convolution.

Il convient de noter que, dans ce contexte spécifique, le terme "convolution" fait référence à une opération de combinaison pondérée de valeurs de fenêtre glissante. L'analogie avec la convolution classique en traitement de signaux est pertinente principalement lorsque les noyaux de convolution présentent une symétrie, ce qui n'est pas toujours le cas.

- **Couche de correction** : La couche de convolution est généralement suivie d'une couche de correction. Cette couche applique une fonction d'activation (p.ex. tanh, ReLU, etc.) à la sortie de la couche de convolution.

- **Couche de sous-échantillonnage (*pooling*)** : La couche de sous-échantillonnage est utilisée pour réduire la taille de l'image d'entrée et par conséquent réduire le nombre de paramètres dans le réseau tout en préservant les caractéristiques les plus importantes, ce qui permet d'accélérer le temps de calcul et de réduire le risque de sur-apprentissage.

  Il existe plusieurs façons d'implémenter l'opération de sous-échantillonnage. Les opérations les plus couramment utilisées sont le sous-échantillonnage par la valeur maximale (*Max pooling*) et le sous-échantillonnage par la valeur moyenne (*Average pooling*). La figure 2.6 illustre un exemple de ces deux types d'opérations de sous-échantillonnage.

  Le sous-échantillonnage par la valeur maximale et le sous-échantillonnage par la valeur moyenne consistent à extraire respectivement la valeur maximale ou moyenne d'une fenêtre glissante sur l'image d'entrée.

- **Couches totalement connectées** : Les couches entièrement connectées correspondent à un perceptron multicouche et sont généralement situées à la fin du CNN. Les cartes d'activations obtenues sont concaténées dans un vecteur par l'opération

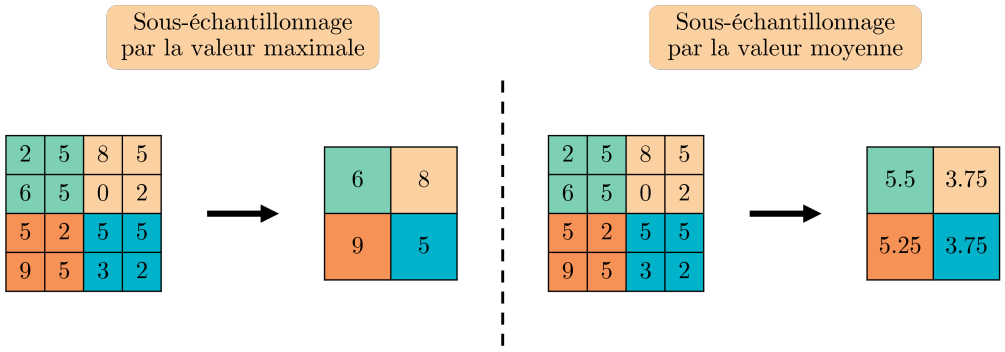

FIGURE 2.6 Exemple d'opérations de sous-échantillonnage.

d'aplatissement (*flattening*) et fournies en entrée aux couches entièrement connectées.

Les CNN sont assez performants pour les problèmes d'analyse d'images, notamment la reconnaissance faciale [38], la reconnaissance de l'écriture manuscrite [47], et le diagnostic médical [60].

Architecture U-Net

L'architecture U-Net, proposée par Ronneberger *et al.* en 2015 [71], est une variante des réseaux de neurones convolutifs adaptée à la segmentation d'images [44, 95]. Elle est particulièrement utilisée pour la segmentation d'images médicales. La segmentation d'image consiste à assigner une étiquette à chaque pixel de l'image en fonction de sa classe (par exemple, les cellules cancéreuses ou les tissus sains dans une image médicale).

Le réseau se compose d'une partie contractante, appelée encodeur, et d'une partie expansive, appelée décodeur, reliées par des connexions de saut (*skip connections*). L'encodeur est responsable de l'extraction des caractéristiques de l'image d'entrée, tandis que le décodeur est chargé de la reconstruction de l'image.

L'encodeur est constitué d'un ensemble de couches convolutionnelles qui permettent de capturer les informations pertinentes de l'image d'entrée. Ce processus d'extraction est réalisé en sous-échantillonnant progressivement les données à l'aide de couches de convolution, en utilisant le paramètre "pas de déplacement", soit des couches de sous-échantillonnage par la valeur maximale ou la valeur moyenne. Ainsi, l'encodeur réduit la résolution spatiale de l'image tout en préservant les caractéristiques essentielles.

Le décodeur, quant à lui, agrandit progressivement l'image, jusqu'à rétablir sa résolution d'origine. Plusieurs méthodes de suréchantillonnage sont utilisées, telles qu'une couche de convolution transposée qui applique une opération de convolution inversée sur l'entrée

en effectuant une multiplication et une sommation élément par élément avec un noyau de convolution. D'autres méthodes, comme le suréchantillonnage basé sur les plus proches voisins ou l'interpolation bilinéaire, peuvent également être utilisées. La figure 2.7 présente ces deux exemples de méthodes de suréchantillonnage.

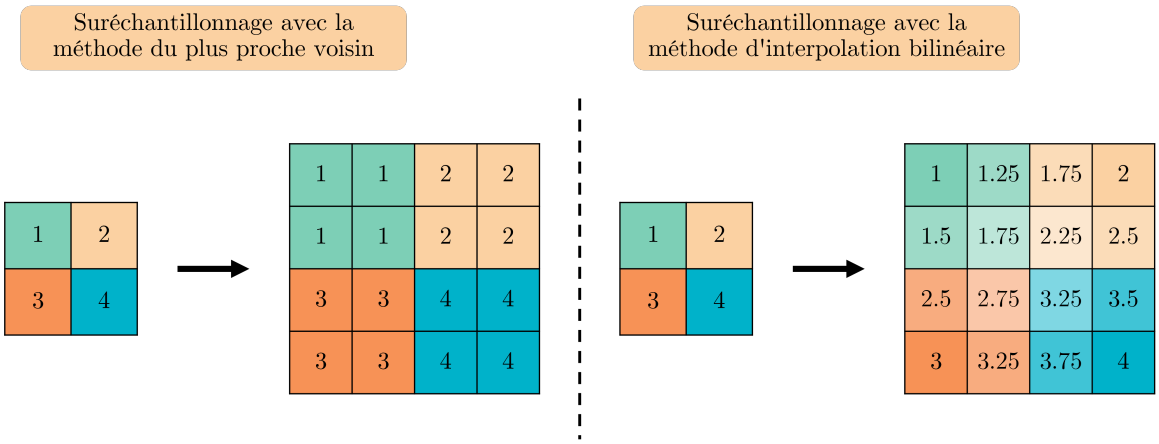

FIGURE 2.7 Exemple d'opérations de suréchantillonnage.

Dans le suréchantillonnage basé sur les plus proches voisins, les valeurs sont simplement dupliquées pour agrandir l'image. En revanche, dans le suréchantillonnage basé sur l'interpolation, une somme pondérée des éléments les proches est calculée pour obtenir les nouvelles valeurs des pixels.

Les deux parties, à savoir l'encodeur et le décodeur, sont connectées par des connexions de saut. Ces connexions permettent la préservation de l'information de haute résolution de l'image d'entrée dans la partie expansive, tout en se focalisant sur les régions d'intérêt de l'image. Elles fournissent une voie directe pour transmettre les informations des couches de l'encodeur aux couches correspondantes du décodeur, améliorant ainsi l'apprentissage, notamment lorsque les architectures sont profondes, et favorisant une convergence plus rapide [21]. Les deux types de connexions de saut les plus couramment utilisés sont les connexions par concaténation ou celles par addition.

En plus de leur utilisation répandue dans le domaine de la segmentation d'images médicales, plusieurs travaux ont également tenté d'appliquer les réseaux U-Net à la tâche de rehaussement de la parole en utilisant des représentations spectro-temporelles en entrée [32, 13]. En effet, en convertissant le signal temporel de la parole en un spectrogramme, le rehaussement de la parole peut alors être considéré comme une tâche de segmentation sémantique d'images. Ces travaux ont montré que les réseaux U-Net peuvent améliorer significativement la qualité de la parole rehaussée.

Réseaux de neurones récurrents

Les réseaux de neurones récurrents (*Recurrent Neural Network* - RNN) sont largement utilisés pour la modélisation des données séquentielles de longueur variable telles que des séries temporelles (p.ex. des signaux audio), des textes [2, 22, 84]. Ces réseaux sont constitués de neurones dans lesquels les informations antérieures peuvent être utilisées comme entrées, via des états cachés, pour traiter des données séquentielles.

La figure 2.8 illustre l'architecture d'un RNN déplié dans le temps, dans lequel l'état caché d'un neurone h_{t-1} au temps $t-1$ est réinjecté en entrée de ce même neurone au temps t.

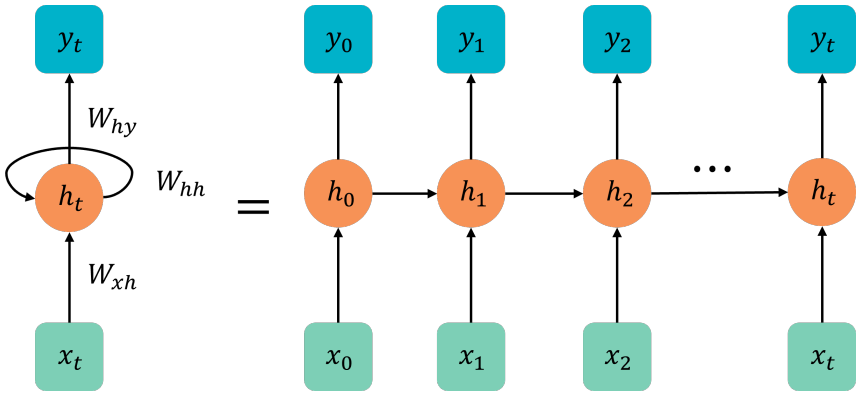

FIGURE 2.8 Architecture d'un réseau de neurones récurrent.

L'état caché h_t et la sortie y_t sont calculés comme suit :

$$h_t = f_1(W_{xh}x_t + W_{hh}h_{t-1}) \tag{2.8}$$

$$y_t = f_2(W_{hy}h_t) \tag{2.9}$$

où W_{xh}, W_{hh} et W_{hy} sont les matrices des poids du réseau et f_1 et f_2 sont des fonctions d'activation.

Les réseaux récurrents sont particulièrement bien adaptés pour traiter des dépendances temporelles (signal audio, texte, vidéo). Cependant, leur architecture de base (souvent appelé *vanilla RNN*) [64] présente une difficulté majeure à apprendre à partir de longues séquences de données en entrée [30]. Pour éviter la disparition ou l'explosion du gradient, il est possible d'utiliser des unités de mémoire à long terme (*Long-Short Term Memory* - LSTM) [30] ou des unités de porte récurrente (*Gated Recurrent Unit* - GRU) [41]. Ces architectures sont très performantes dans divers domaines tels que la reconnaissance automatique de la parole [78], la traduction automatique [81], et la génération des légendes d'images [35].

Les réseaux de neurones convolutifs et récurrents ont été largement utilisés dans différents domaines pour modéliser les dépendances spatiales et temporelles. Toutefois, une autre famille de réseaux neuronaux, les réseaux à décharges, ont récemment suscité un intérêt croissant. Cet intérêt est illustré par un concours récemment lancé par Intel, mettant l'accent sur le rehaussement de la parole [86]. La section suivante se focalise sur une revue approfondie des réseaux à décharges, en soulignant leurs caractéristiques essentielles et leur potentiel pour la tâche de rehaussement de la parole.

2.2.3 Réseaux de neurones à décharges

L'un des principaux défis des réseaux de neurones conventionnels est de réduire la consommation énergétique, les ressources computationnelles et le temps de calcul. C'est pourquoi les chercheurs se sont tournés vers les réseaux de neurones à décharges, également connus sous le nom de troisième génération de réseaux de neurones [65]. Ces réseaux sont plus biologiquement plausibles que les réseaux de neurones conventionnels, car ils reproduisent certaines propriétés des réseaux de neurones biologiques. De plus, les SNN peuvent être utilisés pour les mêmes applications que les réseaux de neurones artificiels [83].

Une des différences majeures entre les réseaux de neurones conventionnels et les réseaux de neurones à décharges est que ces derniers intègrent la notion de temps dans leur traitement de données en utilisant des neurones qui communiquent via des signaux binaires et parcimonieux de manière asynchrone. En outre, les SNN peuvent être mis en œuvre de manière très efficace dans un processeur neuromorphique, qui est lui-même composé de neurones reliés par des synapses. Les processeurs neuromorphiques permettent de déployer des SNN dans des applications du monde réel.

Modèle de neurone à décharges

Les modèles de neurones à décharges sont des modèles mathématiques inspirés des neurones biologiques pour reproduire leur dynamique de traitement de l'information. Ces modèles sont caractérisés par un potentiel membranaire qui intègre les signaux d'entrée et génère une décharge lorsqu'un seuil prédéfini est atteint.

La figure 2.9 montre la modélisation d'un neurone à décharges dans un SNN.

Dans la figure 2.9, l'activité présynaptique est modélisée par des signaux temporels d'entrée $x_1(t)$, $x_2(t)$ et $x_3(t)$ pondérés par des poids synaptiques w_1, w_2 et w_3. Si la somme pondérée des signaux d'entrée dépasse un seuil prédéfini, le neurone de sortie génère une décharge. Le signal de sortie $y(t)$ représente un train de décharges, qui est transmis aux neurones post-synaptiques.

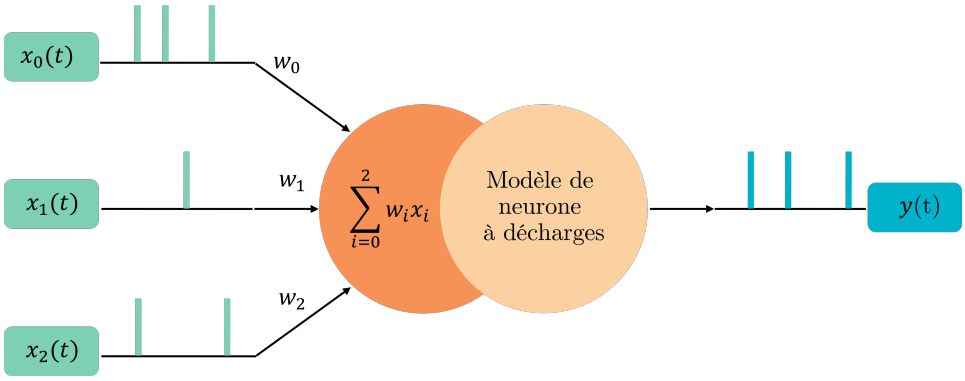

FIGURE 2.9 Illustration du neurone à décharges.

Plusieurs modèles de la dynamique neuronale ont été proposés en neuroscience computationnelle, tels que le modèle de Hodgkin-Huxley [31], le modèle de FitzHugh-Nagumo [26], le modèle de Morris-Lecar [54], le modèle de Hindmarsh–Rose [29], le modèle de Izhikevich [33] et le modèle à intégration et décharge avec fuite [1]. Le choix du modèle à utiliser dépend du domaine d'application, car il y a un compromis entre plausibilité biologique et coût computationnel [34].

Modèle à intégration et décharge avec fuite

Le modèle à intégration et décharge avec fuite (*Leaky Integrate-and-Fire* - LIF) est l'un des modèles les plus simples pour illustrer l'activité d'un neurone biologique, ce qui en fait une option populaire pour la modélisation de réseaux de neurones à décharges profonds. Le modèle a été proposé pour la première fois par Louis Lapicque en 1907 [1].

La dynamique du potentiel membranaire d'un neurone LIF est décrite par l'équation différentielle suivante :

$$\tau_{mem} \frac{dU}{dt} = -(U(t) - U_{repos}) + RI(t) \tag{2.10}$$

où $U(t)$ représente le potentiel membranaire, U_{repos} le potentiel membranaire de repos, $I(t)$ le courant synaptique, $\tau_{mem} = RC$ la constante de temps, R la résistance membranaire, et C la capacité membranaire. Le modèle LIF modélise le neurone comme une combinaison parallèle d'une résistance R et d'un condensateur de capacité C. L'équation (2.10) décrit le potentiel membranaire du neurone LIF tant qu'il est en dessous d'un seuil U_{seuil}. Lorsque ce seuil est atteint ou dépassé, une décharge est générée et le potentiel membranaire $U(t)$ est réinitialisé à U_{repos}. La figure 2.10 illustre un exemple de la dynamique d'un neurone à décharges LIF, où les impulsions vertes représentent des décharges présynap-

tiques, les impulsions bleues représentent des décharges postsynaptiques et la courbe en orange représente le potentiel membranaire postsynaptique.

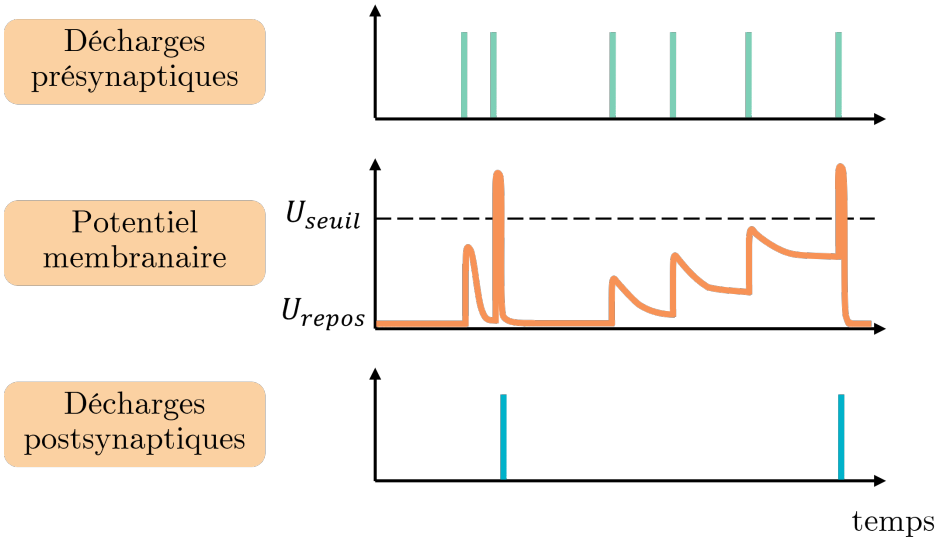

FIGURE 2.10 Illustration de la dynamique d'un neurone à décharges LIF.

Chaque décharge présynaptique entraîne une croissance exponentielle du potentiel membranaire postsynaptique. Si le seuil de décharge est atteint, une décharge post-synaptique est générée et le potentiel membranaire est réinitialisée à U_{repos}. En l'absence de décharges, le potentiel membranaire commence à diminuer jusqu'à atteindre sa valeur de repos.

Bien que le modèle LIF ne modélise pas l'impact complet des canaux ioniques sur l'évolution du potentiel membranaire, il est suffisamment précis pour reproduire des taux de décharges réalistes et décrire la dynamique du potentiel membranaire. Cela en fait une option pratique pour la modélisation de réseaux de neurones à décharges profonds.

Codage neuronal

Les SNN utilisent des décharges binaires et asynchrones pour communiquer entre les neurones. Cependant, les données du monde réel sont souvent continues, nécessitant ainsi une étape de codage permettant de convertir ces signaux continus en décharges pouvant ensuite être traitées par les SNN. Cette sous-section examine brièvement deux des méthodes de codage neuronal les plus couramment utilisées dans les SNN, soient le codage en instants de décharges et le codage en taux de décharges.

Le codage en instants de décharges, également connu sous le nom de codage temporel, consiste à encoder l'information sous forme de décharges à des instants spécifiques dans une fenêtre temporelle donnée. Cette méthode est sensible au bruit neuronal et peut entraîner un grand débit de décharges, car chaque échantillon est encodé en une seule décharge.

Le codage en taux de décharges, également connu sous le nom de codage fréquentiel, est une autre méthode de codage courante. Cette méthode calcule le nombre de décharges dans une fenêtre temporelle donnée et l'encode comme une valeur de taux de décharges. Cette méthode permet de réduire le bruit neuronal en effectuant une moyenne des décharges sur une période de temps assez longue. Cependant, elle peut également entraîner un grand débit de décharges.

Codage d'entrée direct

La méthode de codage d'entrée direct consiste à transmettre les données d'entrée continues directement à la couche d'entrée du SNN [74, 45, 67], sans les encoder en décharges. Contrairement aux méthodes de codage neuronal précédemment présentées, chaque neurone de la couche d'entrée traite la valeur continu reçue en tant que courant synaptique (2.10). Cette approche évite le bruit introduit par le codage neuronal et permet une meilleure précision et une réduction significative du débit de décharges.

L'utilisation du codage d'entrée direct peut offrir de nombreux avantages par rapport aux méthodes traditionnelles de codage neuronal, tels que la réduction du nombre de décharges nécessaires pour transmettre une information spécifique, ce qui réduit les besoins de traitement énergétique [67]. De plus, la méthode de codage d'entrée direct permet de préserver l'information continue d'origine, offrant ainsi une meilleure précision de traitement de l'information.

Les équations caractérisant la dynamique du SNN sont importantes pour comprendre les mécanismes sous-jacents de cette approche. Ces équations décrivent l'évolution de la membrane du neurone, ainsi que la manière dont les impulsions sont générées et transmises entre les neurones. La prochaine section examinera plus en détail la dynamique du SNN.

Dynamique du réseau de neurones à décharges

Cette section aborde l'utilisation de neurones à intégration et décharge avec fuite pour modéliser la dynamique d'un réseau de neurones à décharges. Le potentiel membranaire $U_i^{(l)}(t)$ d'un neurone LIF d'indice i à la couche l est décrit par l'équation différentielle suivante :

$$\tau_{mem}\frac{dU_i^{(l)}}{dt} = -(U_i^{(l)}(t) - U_{repos}) + RI_i^{(l)}(t) \tag{2.11}$$

où, de façon similaire à (2.10), τ_{mem} représente la constante de temps du potentiel membranaire, U_{repos} la valeur de repos de la membrane, R la résistance de membrane et $I_i^{(l)}(t)$ le courant synaptique reçu par le neurone d'indice i à la couche l.

Le train de décharges émises par le neurone est modélisé par une somme d'impulsions de Dirac :

$$S_i^{(l)}(t) = \sum_{k \in C_i^l} \delta(t - t_i^k) \tag{2.12}$$

où C_i^l représente l'ensemble des décharges émises par le neurone d'indice i à la couche l, et t_i^k les instants de décharges.

Les décharges passent par les connexions synaptiques et génèrent un courant synaptique $I_i^{(l)}(t)$, dont l'équation différentielle est la suivante :

$$\frac{dI_i^{(l)}}{dt} = -\frac{I_i^{(l)}(t)}{\tau_{syn}} + \sum_j W_{ij}^{(l)} S_j^{(l-1)}(t) + \sum_j V_{ij}^{(l)} S_j^{(l)}(t) \tag{2.13}$$

où τ_{syn} représente la constante de temps synaptique, $W_{ij}^{(l)}$ la matrice des poids pour les connexions de la propagation avant et $V_{ij}^{(l)}$ la matrice des poids pour les connexions récurrentes. Le premier terme de l'équation représente la fuite du courant synaptique, le deuxième terme représente la propagation avant avec le train de décharges généré par un neurone d'indice j à la couche précédente $l-1$, $S_j^{(l-1)}$, pondérés par la matrice des poids $W_{ij}^{(l)}$, et le troisième terme représente les connexions récurrentes avec le train de décharges généré par un neurone d'indice j à la couche l, $S_j^{(l)}$, pondérés par la matrice des poids $V_{ij}^{(l)}$.

Les équations différentielles du potentiel membranaire $U(t)$ (2.11) et du courant synaptique $I(t)$ (2.13) peuvent être résolues par des méthodes approximatives basées sur la discrétisation de la variable t. Aussi, ces équations sont résolues sous les hypothèses suivantes : $U_{repos} = 0$, $R = 1$ et $U_{seuil} = 1$. Les équations de modélisation d'un neurone LIF sont donc :

$$I_i^{(l)}[n+1] = \alpha I_i^{(l)}[n] + \sum_j W_{ij}^{(l)} S_j^{(l-1)}[n] + \sum_j V_{ij}^{(l)} S_j^{(l)}[n] \tag{2.14}$$

$$U_i^{(l)}[n+1] = \beta U_i^{(l)}[n] + I_i^{(l)}[n] - U_{seuil} S_i^{(l)}[n] \tag{2.15}$$

où $\alpha = exp(-\frac{\Delta t}{\tau_{syn}})$ et $\beta = exp(-\frac{\Delta t}{\tau_{mem}})$ sont des constantes de décroissance pour les courants synaptiques et les potentiels membranaires respectivement, et Δt est le pas de temps.

La condition de génération des décharges est appliquée en utilisant une fonction d'activation non linéaire, la fonction d'Heaviside, selon l'équation suivante :

$$S_i^{(l)}[n] = \Theta(U_i^{(l)}[n] - U_{seuil}) \tag{2.16}$$

avec :

$$\Theta(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{sinon.} \end{cases} \tag{2.17}$$

L'équation (2.16) montre que les décharges sont générées lorsque le potentiel membranaire U dépasse le seuil de décharge U_{seuil}.

Les équations de modélisation de la dynamique d'un neurone LIF (2.14) et (2.15) montrent bien que l'évolution du potentiel membranaire U et du courant synaptique I à un instant $n+1$ dépend des entrées (décharges générées par la couche précédente) et de leurs valeurs précédentes (à l'instant n). Cette propriété de dépendance à court-terme permet d'établir une équivalence entre les réseaux de neurones à décharges et les réseaux de neurones récurrents.

Il est à noter que les équations différentielles et les solutions discrètes présentées dans cette section sont dérivées des travaux de Zenke *et al.* [55].

La présente section a examiné en détail la dynamique de modélisation d'un SNN. La section suivante se concentre sur une présentation des méthodes d'apprentissage les plus utilisées pour les SNN.

Apprentissage

Les paradigmes d'apprentissage des SNN peuvent être regroupés en trois catégories :

- Les méthodes d'apprentissage non supervisé
- Les méthodes de conversion des ANN en SNN
- Les méthodes d'apprentissage supervisé

Une des approches d'apprentissage non supervisé est une loi appelée la plasticité fonction du temps d'occurrence des impulsions (*Spike Timing-Dependent Plasticity* - STDP) [5], qui détermine la force des synapses en fonction de la corrélation temporelle des décharges neuronales. La STDP permet d'apprendre des motifs fréquents dans les données, mais pas forcément ceux qui sont pertinents pour résoudre une tâche donnée.

Une autre approche consiste à entraîner un ANN et à utiliser ensuite les poids appris pour implémenter un SNN [20]. Cette méthode permet de bénéficier de l'efficacité énergétique des SNN et de la performance des ANN. Cependant, elle se base sur le codage en taux de décharges, qui peut générer un grand débit de décharges et entraîner une consommation d'énergie importante.

Enfin, l'apprentissage supervisé des SNN utilise des algorithmes de rétropropagation de l'erreur, similaires à ceux utilisés pour les ANN. Cette méthode permet d'entraîner des architectures profondes de SNN en utilisant des techniques d'optimisation telles que l'apprentissage par le gradient de substitution [55], qui est l'une des méthodes les plus couramment utilisées dans ce contexte. Les récentes avancées dans l'apprentissage profond ont montré que cette méthode peut être utilisée pour entraîner des SNN de manière efficace et précise.

Apprentissage par le gradient de substitution

Les réseaux de neurones conventionnels utilisent généralement un algorithme de rétropropagation de l'erreur pour l'apprentissage. Pendant la phase d'entraînement, une fonction de coût est utilisée pour mesurer l'écart entre la sortie obtenue et le résultat attendu. Les poids du réseau sont ensuite ajustés pour minimiser cette fonction de coût, en utilisant les gradients de la fonction de coût par rapport aux poids.

L'algorithme de rétropropagation utilise des gradients ou dérivées partielles de la fonction de coût par rapport à chaque poids du réseau pour mettre à jour ce poids selon sa contribution à l'erreur. Ainsi, il est nécessaire d'utiliser une fonction d'activation dérivable pour pouvoir ajuster les poids du réseau lors de la rétropropagation. Cependant, les SNN utilisent des fonctions d'activation non-dérivables, telles que la fonction d'Heaviside (sa dérivée est nulle partout sauf en 0 où elle est infinie), pour la génération de décharges. Cela pose un problème lors de la rétropropagation, car les gradients ne peuvent pas être calculés.

La méthode du gradient de substitution [55] est une technique prometteuse pour résoudre ce problème. Elle consiste à utiliser une fonction de substitution du gradient calculable lors de la rétropropagation, tout en conservant la fonction d'Heaviside pour la propagation avant.

Les équations suivantes présentent quelques exemples de fonctions de substitution de gradient $h(x)$:

- Dérivée de la fonction d'arc tangente :

$$h(x) = \frac{1}{\pi} \frac{1}{1 + (\pi |x| \frac{\beta}{2})^2} \tag{2.18}$$

- Dérivée de la fonction sigmoïde :

$$h(x) = \sigma(x)(1 - \sigma(x)) \tag{2.19}$$

- Dérivée de la sigmoïde modifiée (*SuperSpike*) [104] :

$$h(x) = \frac{1}{(1 + \beta |x|)^2} \tag{2.20}$$

- Dérivée d'une fonction linéaire par morceaux [24] (*Piecewise linear*) :

$$h(x) = \max(0, 1 - \beta x) \tag{2.21}$$

où β est un paramètre qui contrôle la pente du gradient de substitution.

La représentation graphique des fonctions communes de substitution du gradient est présentée dans la figure 2.11.

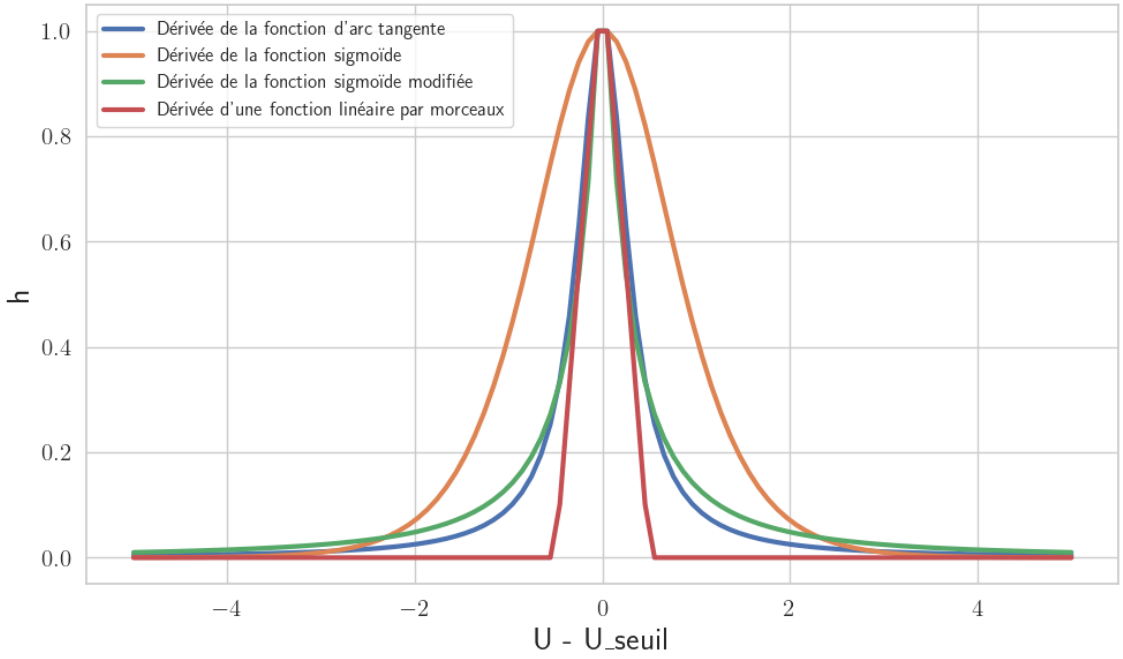

FIGURE 2.11 Représentation graphique des fonctions de substitution du gradient normalisées.

La courbe tracée en bleu illustre la dérivée de la sigmoïde modifiée telle qu'énoncée par Zenke *et al.* [104], tandis que la courbe en orange représente la dérivée de la sigmoïde. La courbe en vert est la représentation graphique de la dérivée de l'arc tangente, et enfin, la courbe en rouge correspond à la dérivée de la fonction linéaire par morceaux, telle qu'introduite par Esser *et al.* [24].

La méthode d'apprentissage par le gradient de substitution a été largement utilisée dans la littérature principalement pour des applications de classification, en raison de sa robustesse et efficacité [105]. Il serait donc intéressant d'évaluer ses performances pour le problème de rehaussement de la parole.

2.3 Rehaussement de la parole à l'aide de réseaux de neurones

2.3.1 Approche générale

Le rehaussement de la parole est une tâche fondamentale dans le domaine du traitement de la parole. Les réseaux de neurones sont devenus une méthode populaire pour le rehaussement de la parole en raison de leur capacité à apprendre des distributions statistiques des caractéristiques à partir des données d'apprentissage. Cela permet de rehausser la parole de manière efficace et précise.

Deux approches principales sont généralement utilisées pour le rehaussement de la parole à l'aide de réseaux de neurones : l'estimation directe du signal de parole propre et l'estimation d'un masque binaire ou continu.

Estimation directe

Dans l'approche de l'estimation directe, le réseau de neurones apprend à estimer directement le signal de parole propre à partir du signal de parole bruité. Dans le domaine fréquentiel, le signal de parole bruité est transformé en représentations fréquentielles telles que le spectre de puissance, qui est ensuite utilisé pour estimer directement le signal de parole propre. Nous présenterons plus bas une des approches permettant d'obtenir une représentation fréquentielle du signal, soit la transformée de Fourier à court terme.

La Figure 2.12 illustre le processus général de rehaussement de la parole en utilisant un réseau de neurones avec la méthode d'estimation directe.

Le rehaussement de la parole par la méthode d'estimation directe peut être divisé en trois phases principales :

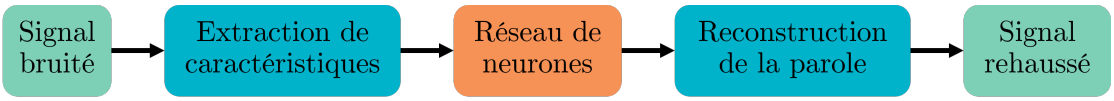

FIGURE 2.12 Processus général de rehaussement de la parole avec un réseau de neurones par la méthode d'estimation directe.

1. Extraction de caractéristiques : Dans cette phase, des caractéristiques pertinentes sont extraites à partir du signal de parole bruité, généralement en utilisant des techniques de transformation fréquentielle. Le spectre de puissance est l'une des représentations fréquentielles couramment utilisées.
2. Rehaussement de la parole : Le réseau de neurones utilise les caractéristiques extraites pour estimer directement le signal de parole propre. L'apprentissage se fait en utilisant des exemples de paires de signaux de parole bruités et propres, afin d'entraîner le réseau à reproduire le signal de parole propre à partir du signal bruité.
3. Reconstruction de la parole rehaussée : Une fois le signal de parole propre estimé, il est souvent nécessaire de le reconstruire dans le domaine temporel pour obtenir le signal de parole rehaussée final. Cela peut être réalisé en inversant les transformations effectuées lors de l'extraction des caractéristiques, ou en utilisant d'autres méthodes de reconstruction.

Estimation d'un masque

Dans le problème du rehaussement de la parole, l'estimation d'un masque est une approche largement adoptée. Elle consiste à entraîner un réseau de neurones pour estimer un masque binaire ou continu, qui est ensuite appliqué à la représentation de la parole bruitée. Le masque permet alors essentiellement de retirer les parties qui sont constituées de bruit et préserver celles qui sont composées de parole, permettant ainsi de rehausser la qualité de la parole.

La Figure 2.13 présente le schéma général du processus de rehaussement de la parole à l'aide d'un réseau de neurones en utilisant la méthode d'estimation d'un masque.

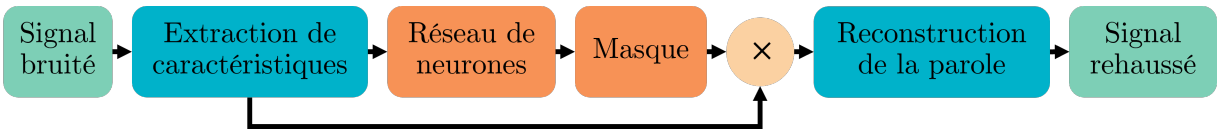

FIGURE 2.13 Processus général de rehaussement de la parole avec un réseau de neurones par la méthode d'estimation d'un masque.

Les étapes de rehaussement de la parole avec un réseau de neurones par la méthode d'estimation d'un masque sont similaires à celles de la méthode d'estimation directe. La

principale distinction entre l'estimation directe et l'estimation d'un masque réside dans la cible d'apprentissage. Dans la méthode d'estimation directe, la cible d'apprentissage est une représentation du signal rehaussée. En revanche, dans la méthode d'estimation d'un masque, le réseau de neurones apprend à estimer un masque (p.ex. le masque binaire idéal [93], le masque de ratio idéal [79], le masque de ratio idéal complexe [99], etc.). Les avantages spécifiques de chaque méthode seront détaillés dans la prochaine section.

Comparaison entre l'estimation directe et l'estimation d'un masque

L'estimation directe et l'estimation d'un masque sont deux approches largement utilisées pour le problème de rehaussement de la parole en utilisant des réseaux de neurones. Plusieurs travaux de recherche ont comparé ces deux approches afin de déterminer leurs avantages respectifs.

Concernant l'estimation d'un masque, des études ont démontré que cette méthode offre une meilleure intelligibilité de la parole rehaussée [96, 58]. En revanche, la méthode d'estimation directe s'est révélée plus robuste aux variations de SNR des signaux bruités comparativement à la méthode d'estimation d'un masque [40, 106]. Donc, l'application ultérieure de l'algorithme de rehaussement de la parole peut être un critère important pour choisir ce qui est plus important, l'intelligibilité de la parole ou l'élimination de bruit.

Il convient également de noter que le choix de l'approche peut être influencé par l'architecture du réseau de neurones utilisé. Par exemple, en termes de performance de rehaussement, la méthode d'estimation directe a été dépassée par la méthode d'estimation d'un masque en utilisant un MLP [40, 106]. Toutefois, l'approche d'estimation directe est plus performante lorsqu'elle est utilisée avec des autoencodeurs convolutionnels tels que le modèle U-Net, par rapport à l'approche d'estimation d'un masque [56].

De plus, ces deux approches, l'estimation directe et l'estimation d'un masque, peuvent être réalisées soit dans le domaine temporel, soit dans le domaine fréquentiel. Une des approches permettant de transformer un signal du domaine temporel vers le domaine fréquentiel est l'utilisation de la transformée de Fourier à court terme.

Transformée de Fourier à court terme

La transformée de Fourier à court terme (*Short-Time Fourier Transform* - STFT) est une méthode de traitement de signal largement utilisée dans le rehaussement de la parole. Cette technique permet d'analyser le signal de parole dans le domaine fréquentiel tout en préservant les informations du domaine temporel. Elle consiste à découper le signal de parole en plusieurs courts segments, appelés fenêtres d'analyse, sur lesquels on applique une transformation de Fourier pour obtenir une représentation fréquentielle locale de ces

signaux. Cette représentation permet de visualiser comment les composantes fréquentielles du signal évoluent dans le temps, ce qui est important, car le bruit affecte souvent certaines régions de fréquence plus que d'autres, et il peut également évoluer dans le temps en raison de divers facteurs. En analysant le signal de parole dans le domaine fréquentiel, il est possible d'identifier et d'isoler ces régions.

La STFT est une méthode rapide et efficace, deux critères importants dans le contexte du rehaussement de la parole en temps réel, qui est souvent requis dans des applications réelles telles que les appareils auditifs ou les appareils mobiles.

L'équation de la STFT d'un signal numérique x est :

$$X[m,k] = \sum_{n=0}^{N-1} x[n+mH]w[n]e^{-j\frac{2\pi nk}{N}} \tag{2.22}$$

où w est une fenêtre d'analyse de taille N, H est le nombre des pas d'avancement et $X[m,k]$ est le coefficient de Fourier d'indice k pour la fenêtre temporelle d'indice m. Le carré du module de la STFT est couramment utilisé pour obtenir une représentation temps-fréquence appelée spectrogramme. Le spectrogramme est défini par l'équation suivante :

$$Y[m,k] = |X[m,k]|^2 \tag{2.23}$$

Le spectrogramme permet de visualiser les variations d'énergie pour la bande de fréquence d'indice k et la fenêtre temporelle d'indice m pour un signal x.

2.3.2 Approches basées sur des réseaux de neurones conventionnels

Grâce aux avancées de l'apprentissage profond, plusieurs architectures basées sur les réseaux de neurones conventionnels ont été développées et largement utilisées pour le rehaussement de la parole. Les réseaux de neurones sont capables de modéliser la relation complexe entre les données d'entrée et les données cibles, même lorsque cette relation est inconnue. Dans le contexte du rehaussement de la parole, l'objectif est de rehausser un signal de parole bruité en utilisant une représentation dans le domaine temporel ou une représentation inversible dans un autre domaine comme entrée d'un réseau de neurones.

L'un des modèles les plus remarquables faisant usage des réseaux de neurones conventionnels est le SEGAN (*Speech Enhancement Generative Adversarial Network*) proposé par Pascual *et al.* en 2017 [61]. Ce modèle repose sur les réseaux antagonistes génératifs

(*Generative Adversarial Network* - GAN) [28] pour améliorer la qualité du signal de parole dans le domaine fréquentiel. Le SEGAN repose sur un générateur qui utilise un réseau de neurones convolutif profond pour reconstruire un signal de parole de haute qualité à partir d'un signal de parole bruité. Simultanément, un discriminateur est entraîné pour distinguer les signaux de parole améliorés par le générateur des signaux de parole originaux et bruités. Ce processus itératif de génération et de discrimination permet au générateur de s'améliorer progressivement.

Le modèle U-Net, initialement conçu pour la segmentation d'images biomédicales [71], a également été adapté avec succès pour résoudre d'autres problèmes, y compris le rehaussement de la parole. Par exemple, le Wave-U-Net, proposé par Macartney *et al.* en 2018 [46], utilise des réseaux de neurones convolutifs pour le rehaussement de la parole dans le domaine temporel. De même, les travaux de Bulut *et al.* en 2020 [13] proposent un modèle U-Net basé sur des réseaux de neurones conventionnels pour le rehaussement de la parole dans le domaine fréquentiel.

Les réseaux de neurones conventionnels se sont révélés très performants pour le rehaussement de la parole, en s'adaptant à différents rapports signal-à-bruit et types de bruit. Cependant, leur entraînement requiert d'importantes ressources computationnelles qui continuent d'augmenter rapidement en raison de l'utilisation de modèles de plus en plus complexes et profonds. Dans ce contexte, les réseaux de neurones à décharges apparaissent comme une alternative intéressante pour la mise en œuvre de modèles performants tout en réduisant la consommation d'énergie.

2.3.3 Approches basées sur des réseaux de neurones à décharges

Les réseaux de neurones à décharges émergent comme une alternative prometteuse aux réseaux de neurones conventionnels pour le rehaussement de la parole. Leur principal avantage réside dans leur efficience énergétique et leur plausibilité biologique, en imitant plus fidèlement le comportement des neurones dans le cerveau humain. Quelques travaux de recherche se sont penchés sur l'utilisation de différents modèles de SNN pour résoudre le problème de rehaussement de la parole.

Par exemple, Wall et Glackin [92] ont proposé un SNN à trois couches pour rehausser un signal de parole bruité en utilisant la méthode basée sur l'estimation d'un masque. Dans cette approche, les caractéristiques sont extraites en calculant la STFT du signal d'entrée, puis en utilisant l'algorithme BSA (*Bens Spiker Algorithm*) [76] pour l'encodage de l'amplitude de la STFT en décharges. L'objectif est de permettre au SNN de capturer des relations de corrélation entre les bandes de fréquences du spectrogramme bruité, afin

d'éliminer les sources de bruit non corrélées, d'où, la qualité de rehaussement obtenue dépend fortement des propriétés statistiques du bruit. Les signaux de parole utilisés sont bruités par différents rapports signal-à-bruit de bruit additif blanc gaussien seulement. Ce travail présente plusieurs limites, notamment l'absence d'algorithme d'apprentissage, une architecture peu profonde et des conditions de test ne permettant pas de démontrer la pertinence du modèle dans des situations réelles.

Xing *et al.* [100] ont également proposé un modèle de SNN à trois couches pour le rehaussement de la parole en utilisant une approche basée sur l'estimation d'un masque fréquentiel. Les caractéristiques sont extraites en calculant la STFT du signal d'entrée, qui est ensuite utilisée comme courant d'entrée pour le SNN. Les signaux de parole utilisés sont bruités par cinq types de bruits additifs du monde réel. Bien que ce SNN présente des performances satisfaisantes, son rehaussement repose principalement sur l'élimination des bruits non corrélés, ce qui limite son efficacité dans des environnements plus complexes. De plus, aucune règle d'apprentissage spécifique n'a été utilisée dans ce modèle.

Plus récemment, Intel a développé une solution de rehaussement de la parole basée sur les SNN dans le cadre de sa compétition de rehaussement de la parole utilisant une solution neuromorphique (*Intel Neuromorphic Deep Noise Suppression Challenge* - Intel N-DNS Challenge) [86]. La méthode proposée repose sur l'utilisation d'un réseau neuronal sigma-delta (*Sigma-Delta Neural Network* - SDNN) [57] à trois couches basée sur l'estimation d'un masque fréquentiel. L'amplitude de la STFT codée en delta est utilisée comme entrée du SNN, qui génère un masque multiplicatif pour calculer la STFT rehaussée. Le SNN est entraîné en utilisant la méthode du gradient de substitution.

En résumé, peu de travaux de recherche se sont intéressées à l'utilisation des SNN pour le rehaussement de la parole et ceux qui ont été réalisés présentent certaines limitations. Les travaux existants se sont principalement concentrés sur des architectures peu profondes. Cependant, le succès des réseaux de neurones conventionnels repose en partie sur la capacité à modéliser des architectures profondes capables d'apprendre à partir de vastes ensembles de données. Il est donc envisageable de s'inspirer des architectures de réseaux de neurones conventionnels pour concevoir des SNN avec des performances équivalentes.

Conclusion

Ce chapitre de la revue de littérature a abordé plusieurs aspects liés à la problématique du rehaussement de la parole en utilisant un réseau de neurones à décharges. Il a débuté par la formulation du problème du rehaussement de la parole, mettant en évidence les défis et les

enjeux associés à cette tâche. Quelques approches classiques ont été brièvement présentées, afin de contextualiser les avancées récentes réalisées grâce aux réseaux de neurones.

Une partie importante de ce chapitre a été consacrée à la présentation des réseaux de neurones. Les bases des réseaux de neurones biologiques ont été évoquées pour mieux comprendre les fondements de ces modèles computationnels. Ensuite, les architectures couramment utilisées dans les réseaux de neurones conventionnels ont été exposées, tout comme les modèles de neurones typiquement employés dans les réseaux de neurones à décharges.

En poursuivant, le processus général de rehaussement de la parole en utilisant un réseau de neurones a été expliqué. Les étapes clés de cette méthodologie ont été présentées de manière à offrir une vue d'ensemble cohérente du sujet. Par la suite, des travaux de recherche pertinents ont été exposés, mettant l'accent sur le rehaussement de la parole à l'aide d'un réseau de neurones conventionnel.

Enfin, les travaux spécifiques effectués dans la littérature pour le rehaussement de la parole en utilisant un réseau de neurones à décharges ont été présentés. Cette approche prometteuse suscite un intérêt croissant dans la communauté scientifique en raison de ses capacités à modéliser les comportements de neurones biologiques et à améliorer la qualité de la parole rehaussée.

CHAPITRE 3

Méthodologie

Le chapitre précédent a abordé le problème du rehaussement de la parole ainsi que plusieurs travaux existants basés sur les réseaux de neurones conventionnels et les réseaux de neurones à décharges. Les ANN ont été largement utilisés pour résoudre le problème du rehaussement de la parole. Cependant, malgré le potentiel prometteur des SNN, les études adoptant cette approche sont limitées et présentent certaines contraintes. Dans ce contexte, l'objectif de ce projet est de mettre en œuvre et évaluer un modèle de rehaussement de la parole basé sur un SNN, en visant des performances supérieures ou comparables à celles des méthodes qui n'utilisent pas de SNN.

Le présent chapitre expose la méthodologie adoptée pour mener à bien ce projet de recherche, en s'appuyant sur les travaux existants dans le domaine étudié présentés dans le chapitre précédent. La première partie de ce chapitre se concentrera sur une présentation approfondie du système de rehaussement de la parole proposé. Nous décrirons en détail les aspects fondamentaux de ce système, notamment les étapes d'entraînement et de test qui ont été utilisées. La deuxième partie de ce chapitre portera sur l'architecture détaillée du réseau de neurones à décharges mis en œuvre. Nous expliquerons en détail les différentes couches du réseau et les connexions entre elles. De plus, nous fournirons des informations sur la configuration des couches utilisées. Enfin, la troisième partie de ce chapitre se concentrera sur la fonction de coût utilisée.

3.1 Système de rehaussement de la parole proposé

Le système de rehaussement de la parole proposé repose sur deux étapes distinctes : une première étape d'entraînement d'un réseau neuronal à décharges, suivie d'une deuxième étape de test du réseau neuronal à décharges entraîné afin d'évaluer ses performances.

La Figure 3.1 illustre le flux de travail de l'étape d'entraînement.

L'étape d'entraînement consiste à utiliser un réseau de neurones à décharges dont les poids sont initialisés de manière aléatoire avec une distribution prédéfinie. Ce réseau est entraîné sur un ensemble de données préalablement collectées, comprenant notamment des paires de signaux bruités et propres. L'entraînement vise à mettre à jour itérativement les poids du réseau en utilisant les gradients calculés lors de la rétropropagation. L'objectif de

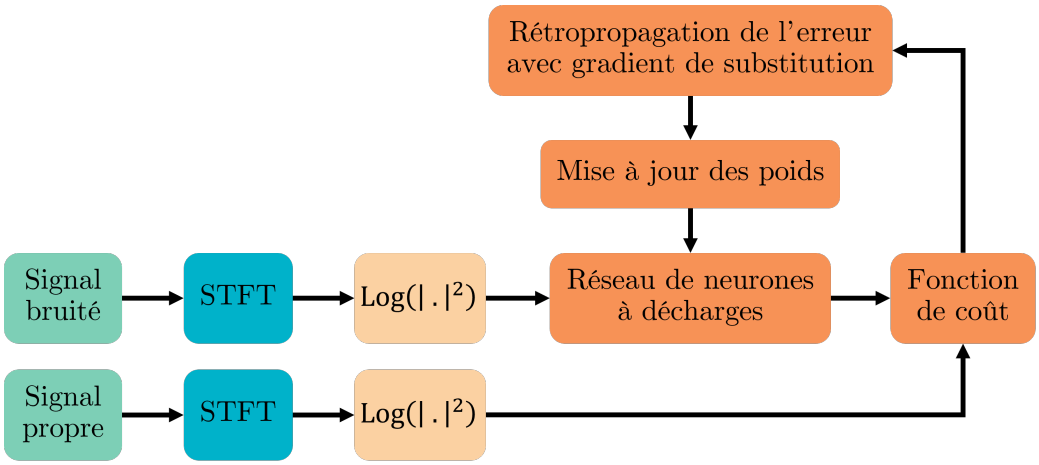

FIGURE 3.1 Étape d'entraînement d'un réseau de neurones à décharges pour le rehaussement de la parole.

l'entraînement est d'apprendre au réseau à estimer la représentation du signal de parole propre à partir de celle du signal bruité.

Dans ce projet, le modèle à intégration et décharge avec fuite [1] est utilisé en raison de sa capacité à produire une dynamique de décharge satisfaisante pour le contexte de rehaussement de la parole. De plus, sa simplicité permet une exécution rapide du SNN. Pour convertir les données d'entrée en décharges, nous avons adopté la méthode de codage d'entrée directe.

L'étape de test vise à évaluer les performances du réseau de neurones à décharges préalablement entraîné. Le flux de travail de cette étape est présenté dans la Figure 3.2.

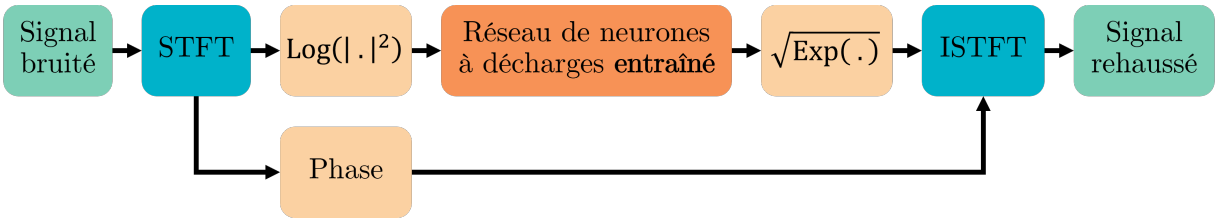

FIGURE 3.2 Étape de test d'un réseau de neurones à décharges pour le rehaussement de la parole.

Au cours de cette étape, le réseau est soumis à des données de test distinctes de celles utilisées lors de l'entraînement. L'objectif est d'évaluer la capacité du réseau à améliorer la qualité de la parole en réduisant le bruit de fond, et en augmentant l'intelligibilité des signaux vocaux.

Ces deux étapes constituent le processus central du système de rehaussement de la parole proposé. L'étape d'entraînement permet d'optimiser les paramètres du réseau neuronal

à décharges afin qu'il puisse apprendre à rehausser la parole, tandis que l'étape de test évalue les performances du réseau sur des données de test indépendantes. Le système de rehaussement de la parole ainsi développé peut être utilisé pour rehausser la qualité de la parole dans diverses applications, telles que la reconnaissance vocale, les systèmes de téléphonie ou les plateformes de communication audio.

3.2 Architecture du réseau de neurones à décharges

Le système de rehaussement de la parole implémenté repose sur un SNN en U-Net (voir section 2.2.2). La Figure 3.3 illustre l'architecture du réseau de neurones à décharges proposé.

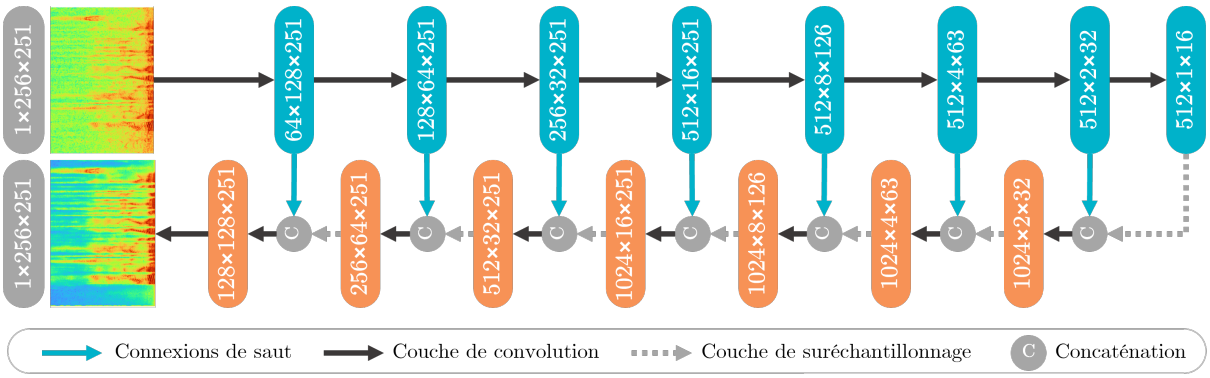

FIGURE 3.3 Architecture proposée pour le réseau de neurones à décharges constitué d'un encodeur (sections en bleues) et d'un décodeur (sections en orange).

Les dimensions des données au sein du réseau de neurones à décharges sont formellement exprimées selon la notation $c \times w \times h$, où c symbolise le nombre de canaux, w est associé à l'axe fréquentiel, et h reflète l'axe temporel. Cette convention permet de définir rigoureusement la structure des données en spécifiant le nombre de canaux, la largeur dans le domaine fréquentiel, et la hauteur dans le domaine temporel. Ces conventions sont couramment adoptées dans le domaine de l'apprentissage profond pour représenter des données multidimensionnelles.

Le réseau de neurones à décharges utilisé est basé sur l'architecture U-Net, un modèle fréquemment utilisé pour les tâches de segmentation et de reconstruction d'images médicales. L'architecture se compose de deux parties principales, à savoir l'encodeur et le décodeur, qui permettent d'extraire les caractéristiques pertinentes du signal bruité et de reconstruire le spectre de puissance logarithmique (*Logarithmic Power Spectrum* - LPS) rehaussé.

La première partie du réseau, l'encodeur (sections en bleues), traite la représentation en entrée qui est le spectre de puissance logarithmique du signal bruité. Ce spectre de puissance est soumis à des opérations de convolution, qui consistent à filtrer le signal pour détecter des motifs et des caractéristiques spécifiques. Chaque couche de convolution est suivie d'une couche de neurones à décharges LIF, qui génèrent des décharges binaires en réponse au courant synaptique d'entrée. Dans cette architecture, l'encodeur comprend huit couches de convolution, permettant ainsi une extraction progressive des caractéristiques du signal bruité. La réduction de dimension se réalise grâce aux paramètres de convolution, et au pas d'avancement, qui réduit progressivement la taille des caractéristiques extraites lorsque sa valeur est supérieure à 1.

La deuxième partie du réseau est le décodeur (sections en orange). Chaque couche du décodeur reçoit à la fois la sortie de la couche précédente et la sortie correspondante de la couche symétrique de l'encodeur, grâce à des connexions de sauts basées sur la concaténation. Cette connexion symétrique permet au décodeur d'accéder aux informations de bas niveau provenant de l'encodeur, aidant ainsi à reconstruire le spectre de puissance logarithmique rehaussé. La dernière couche du décodeur est une couche de neurones LIF qui génère une sortie continue correspondant au spectre de puissance logarithmique rehaussé. Cette couche ne génère pas de décharges, ce qui garantit une représentation continue et lisse du spectre de puissance rehaussé.

Le tableau 3.1 présente la configuration des différentes couches du modèle proposé, indiquant le nombre de noyaux, la taille des noyaux, le pas d'avancement et la taille de la sortie pour chaque couche. En effet, chaque couche de convolution est composée de C_{out} noyaux de convolution, où chaque noyau est représenté par une matrice de taille (k_1, k_2). Lors de la propagation avant, chaque noyau se déplace progressivement d'un pas (s_1, s_2), ce qui signifie qu'il se déplace de s_1 éléments dans la dimension verticale et de s_2 éléments dans la dimension horizontale. Ainsi, une entrée de taille (C_{in}, H_{in}, W_{in}) est soumise à une succession de noyaux de convolution, permettant ainsi d'obtenir une sortie de taille (C_{out}, H_{out}, W_{out}), avec :

- C_{in}, C_{out} représentent respectivement le nombre de canaux de l'image d'entrée et le nombre de canaux produits par la convolution. À la première couche du réseau, Cin est égal à 1 car la LPS est une représentation à deux dimensions.
- H_{in}, H_{out} représentent respectivement la hauteur de la carte d'activation d'entrée et la hauteur de la carte d'activation de sortie, correspondant ainsi à la dimension verticale. À la première couche du réseau, H_{in} est égal au nombre de bandes de fréquence de la LPS d'entrée.

- W_{in}, H_{out} représentent respectivement la largeur de la carte d'activation d'entrée et la largeur de la carte d'activation de sortie, correspondant ainsi à la dimension horizontale. À la première couche du réseau, W_{in} est égal au nombre de fenêtres temporelles de la LPS d'entrée.

Il est à noter que les dimensions de sortie sont calculées en fonction des dimensions d'entrée et des paramètres de la couche de convolution.

TABLEAU 3.1 Configuration des couches du modèle proposé.

Couche	**Nombre de noyaux**	**Taille de noyaux**	**Pas d'avancement**	**Taille de la sortie**
E_1	64	(7, 5)	(2, 1)	(64, 128, 251)
E_2	128	(7, 5)	(2, 1)	(128, 64, 251)
E_3	256	(7, 5)	(2, 1)	(256, 32, 251)
E_4	512	(5, 5)	(2, 1)	(512, 16, 251)
E_5	512	(5, 5)	(2, 2)	(512, 8, 126)
E_6	512	(3, 3)	(2, 2)	(512, 4, 63)
E_7	512	(3, 3)	(2, 2)	(512, 2, 32)
E_8	512	(3, 3)	(2, 2)	(512, 1, 16)
D_1	512	(3, 3)	(1, 1)	(512, 2, 32)
D_2	512	(3, 3)	(1, 1)	(512, 4, 63)
D_3	512	(3, 3)	(1, 1)	(512, 8, 126)
D_4	512	(5, 5)	(1, 1)	(512, 16, 251)
D_5	512	(5, 5)	(1, 1)	(256, 32, 251)
D_6	256	(7, 5)	(1, 1)	(128, 64, 251)
D_7	128	(7, 5)	(1, 1)	(64, 128, 251)
D_8	64	(7, 5)	(1, 1)	(1, 256, 251)

L'architecture proposée comprend huit couches d'encodage, E_1 à E_8, et huit couches de décodage, D_1 à D_8. Les couches d'encodage capturent les caractéristiques pertinentes de l'entrée, tandis que les couches de décodage reconstruisent le LPS rehaussé.

Les couches d'encodage ont des configurations similaires, avec une augmentation du nombre de noyaux et une réduction de la taille spatiale de la sortie à chaque couche successive. Les couches d'encodage initiales (E_1 à E_3) ont des noyaux de taille (7, 5), suivies des couches d'encodage subséquentes (E_4 à E_8) avec des noyaux de taille (5, 5) et (3, 3). Le pas de déplacement dans les deux directions est de (2, 1) pour les couches E_1 à E_4, et il est de (2, 2) pour les couches E_5 à E_8. Ce paramètre de pas de déplacement permet de sous-échantillonner les données en réduisant de moitié les dimensions correspondantes du produit de convolution.

Les couches de décodage ont des configurations similaires aux couches d'encodage correspondantes, mais en sens inverse. Les couches de décodage initiales, D_1 à D_3, ont des noyaux de taille (3, 3) et un pas de déplacement de (1, 1), tandis que les couches de décodage subséquentes, D_4 à D_8, utilisent des noyaux de taille (5, 5) et (7, 5) avec un pas de déplacement de (1, 1). Afin d'augmenter la taille des données d'une couche à l'autre, la méthode de suréchantillonnage par les plus proches voisins est employée.

Il est à noter que cette architecture du réseau de neurones à décharges est inspirée des travaux de Bulut *et al.* [13] qui portaient sur le rehaussement à l'aide de réseaux de neurones conventionnels. Nous nous sommes inspirés de ces travaux pour optimiser notre approche.

Cette section a présenté en détails l'architecture proposée ainsi que la configuration des couches du réseau. Les prochaines section présenteront la fonction de coût utilisé ainsi que la fonction de gradient de substitution pour la rétropropagation de l'erreur.

3.3 Fonction de coût

La fonction de coût utilisée est la distance spectrale logarithmique (*Log-Spectral Distance* - LSD), comme définie dans [13] :

$$L_{LSD} = \frac{1}{M} \sum_{m=0}^{M-1} \sqrt{\frac{1}{K} \sum_{k=0}^{K-1} (X[m,k] - \hat{X}[m,k])^2} \tag{3.1}$$

où $X[m,k]$ et $\hat{X}[m,k]$ représentent respectivement le spectre de puissance logarithmique propre et celui estimé, avec m l'indice de la fenêtre et k l'indice de la bande de fréquence.

Conclusion

Ce chapitre a exposé un système de rehaussement de la parole fondé sur un réseau de neurones à décharges, en fournissant une description détaillée de l'architecture du réseau proposé ainsi que de la configuration de ses différentes couches. La fonction de coût employée dans notre approche a également été indiquée.

Le chapitre suivant sera dédié à la description rigoureuse des conditions expérimentales établies pour évaluer et comparer l'efficacité de l'approche proposée.

CHAPITRE 4

Conditions expérimentales

Ce chapitre expose les conditions expérimentales relatives à l'entraînement et au test du modèle proposé, en complément de la méthodologie présentée dans le chapitre précédent. La première section se concentre sur une présentation détaillée de la base de données utilisée, mettant en évidence ses caractéristiques et son contenu. La deuxième section décrit en profondeur les étapes de prétraitement des données, expliquant les techniques appliquées pour préparer les données brutes en vue de l'entraînement du réseau de neurones à décharges. La configuration d'entraînement est ensuite présentée dans la troisième section, où les méthodes d'initialisation des paramètres du modèle et les stratégies d'optimisation sont détaillés. La quatrième section expose les métriques d'évaluation utilisées pour évaluer la performance du modèle, en soulignant leur pertinence et leurs objectifs. Enfin, la dernière section aborde les aspects d'implémentation, fournissant des informations sur les bibliothèques utilisées, ainsi que les spécifications du matériel informatique utilisé pour les expériences.

4.1 Base de données

Dans le cadre de ce projet, une base de données publiquement disponible [89, 88] a été utilisée. Cette base de données regroupe des signaux de parole propres ainsi que bruités. Les données de parole utilisées ont été extraites du corpus *Voice Bank Corpus* [91] et sont équitablement réparties en fonction du genre des locuteurs anglophones. Les enregistrements de phrases ont été échantillonnés à une fréquence de 48 kHz.

L'ensemble d'entraînement de cette base de données comprend environ 10 heures de signaux de parole provenant de 28 locuteurs. Les signaux propres ont été bruités avec 10 types de bruit distincts, associés à des valeurs de SNR de 15, 10, 5 et 0 dB. Parmi ces types de bruit figurent deux enregistrements de bruit artificiel, ainsi que huit enregistrements de bruit réel provenant de la base de données DEMAND (*Diverse Environments Multi-channel Acoustic Noise Database*) [85]. Pour constituer l'ensemble de validation, des échantillons ont été sélectionnés aléatoirement à partir de l'ensemble d'entraînement. Quant à l'ensemble de test, il se compose de 30 minutes de signaux de parole provenant de deux locuteurs, auxquels ont été ajoutés cinq types de bruit réel provenant de la base de

données DEMAND, avec des valeurs de SNR fixées à 17,5, 12,5, 7,5 et 2,5 dB. L'ensemble des données de test ne sont pas compris dans les données d'entraînement.

Cette base de données offre donc une diversité de signaux de parole bruités et propres, ainsi qu'une variation des niveaux de SNR. Cela permettra d'évaluer efficacement l'efficacité du réseau de neurones à décharges proposé pour le rehaussement de la parole dans différentes conditions expérimentales. Toutefois, il convient de noter qu'une série d'étapes de traitement est nécessaire pour préparer les données en vue de leur utilisation par un réseau de neurones. Ces étapes seront abordées dans la section suivante.

4.2 Prétraitement des données

Cette section aborde le prétraitement des données, une étape essentielle du processus de rehaussement de la parole à l'aide d'un réseau de neurones à décharges. Les données sont préalablement sous-échantillonnées à une fréquence de 16 kHz. Une durée prédéfinie de 4 secondes est attribuée aux signaux d'entrée [15, 103], tout en notant que cet hyperparamètre peut être optimisé. Si la durée du signal dépasse cette limite, seul le premier segment est sélectionné, tandis que si elle est inférieure, le signal est concaténé à lui-même jusqu'à atteindre la longueur prédéfinie.

Ensuite, une transformation de Fourier à court terme est appliquée en utilisant une fenêtre d'analyse de 30 ms avec un chevauchement de 16 ms. Cette transformation permet d'analyser le signal dans le domaine fréquentiel et d'obtenir une représentation en termes de puissance spectrale.

Par la suite, une transformation logarithmique est effectuée en calculant le logarithme de la puissance de l'amplitude résultant de la transformation de Fourier. La figure 4.1 illustre clairement l'effet bénéfique de cette transformation, en améliorant la représentation des variations d'amplitude du signal dans le domaine fréquentiel. Cette amélioration facilite la distinction des caractéristiques pertinentes pour la tâche de rehaussement de la parole.

Enfin, les données d'entrée sont normalisées en utilisant les métadonnées obtenues à partir des signaux bruités présents dans l'ensemble d'entraînement, à savoir la moyenne et l'écart-type. Cette normalisation vise tout d'abord à assurer la stabilité numérique durant la phase d'entraînement, évitant ainsi des problèmes liés aux gradients tels que l'explosion et la fuite des gradients. De plus, mettre à l'échelle les données pour qu'elles aient une variance unitaire, favorise une convergence plus rapide du modèle pendant la phase d'entraînement.

L'approche de prétraitement présentée assure que les signaux d'entrée ont une durée uniforme et sont représentés de manière appropriée pour être utilisés par le réseau de neurones.

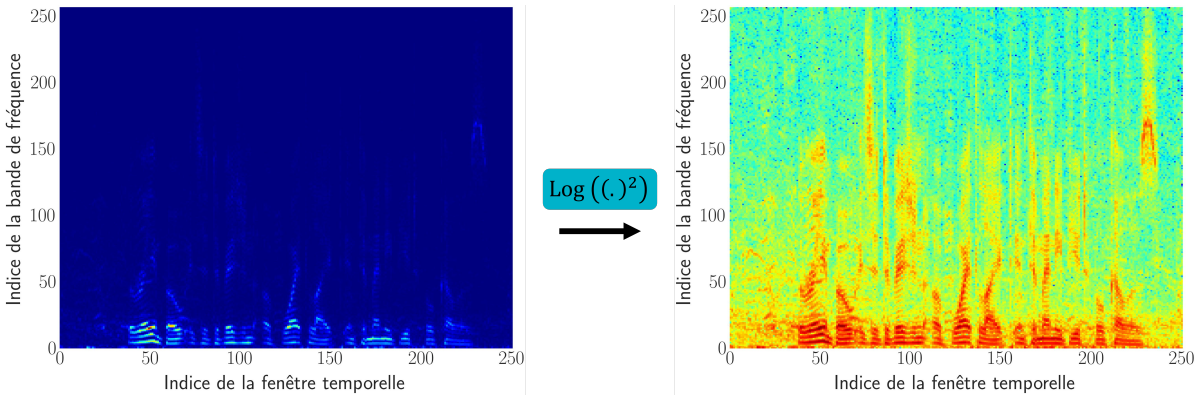

FIGURE 4.1 Exemple de calcul du logarithme de la puissance de l'amplitude de la STFT d'un signal de parole.

De plus, la normalisation des données atténue l'impact du bruit et garantit une convergence plus stable du modèle pendant la phase d'entraînement. La section suivante présente la configuration d'entraînement utilisée.

4.3 Configuration d'entraînement

Cette section présente la configuration utilisée pour entraîner le réseau de neurones à décharges proposé. Le choix des paramètres de configuration découle d'une série d'expérimentations préliminaires approfondies. L'objectif était de déterminer les valeurs les plus appropriées pour l'apprentissage du réseau de neurones à décharges implémenté, afin d'optimiser les performances de rehaussement de la parole.

Tout d'abord, la fonction du gradient de substitution utilisée est la dérivée de la fonction arc tangente (voir section 2.2.3). Rappelons que la méthode du gradient de substitution permet d'utiliser la rétropropagation de l'erreur pour l'entraînement d'un SNN. Aussi, l'algorithme d'optimisation choisi est Adam [39], un algorithme largement adopté dans le domaine de l'apprentissage profond. Les paramètres spécifiques suivants ont été choisis : un taux d'apprentissage de 0.002, un paramètre β égal à (0.5, 0.9), une taille de lot de 32 et un nombre d'itérations d'entraînement de 60.

Les poids des couches de convolution, qui jouent un rôle crucial dans la représentation des caractéristiques de la parole, sont initialisés en utilisant une distribution gaussienne. La distribution a une moyenne de 0 et un écart type de 0.2.

Concernant les neurones LIF utilisés pour modéliser le comportement neuronal, leurs taux de décroissance et leurs seuils de décharge sont également initialisés à l'aide d'une distribution gaussienne. Le taux de décroissance est initialisé avec une moyenne de 0.05 et un

écart type de 0.01, tandis que le seuil de décharge est initialisé avec une moyenne de 1 et un écart type de 0.01.

La section suivante présentera les métriques utilisées pour évaluer la performance obtenue.

4.4 Métriques d'évaluation

Le présent travail fait appel à trois différentes métriques objectives afin d'évaluer les performances de l'approche proposée : l'évaluation perceptive de la qualité de la parole (*Perceptual Evaluation of Speech Quality* - PESQ), l'intelligibilité objective à court terme (*Short-Time Objective Intelligibility* - STOI) et la note moyenne d'opinion de suppression de bruit profond (*Deep Noise Suppression Mean Opinion Score* - DNSMOS).

4.4.1 Évaluation perceptive de la qualité de la parole

L'évaluation perceptive de la qualité de la parole joue un rôle essentiel dans l'évaluation de l'efficacité des systèmes de rehaussement de la parole. Parmi les métriques largement utilisées dans ce domaine, on retrouve le PESQ [70]. Le PESQ est une métrique objective permettant de prédire la qualité perçue de la parole par un auditeur. Il se base sur une analyse psychoacoustique visant à évaluer les aspects perceptuels de la parole, tels que la fidélité et la clarté. Le PESQ attribue ensuite un score de qualité à la parole rehaussée, sur une échelle allant de -0.5 à 4.5, où une valeur plus élevée indique une meilleure qualité perçue.

Il convient de noter que bien que le PESQ soit un indicateur objectif couramment utilisé, il présente certaines limites. En effet, il se concentre principalement sur l'évaluation de la qualité de la parole dans un système de télécommunication, ce qui peut limiter sa représentativité en termes d'intelligibilité de la parole.

4.4.2 Intelligibilité objective à court terme

L'évaluation de l'intelligibilité de la parole revêt une importance primordiale lorsqu'il s'agit d'évaluer l'amélioration de la compréhensibilité de la parole dans un système de rehaussement. Parmi les métriques couramment utilisées à cet effet, le STOI [82] occupe une place prépondérante. Le STOI mesure la corrélation spectrale à court terme entre la parole rehaussée et la parole propre, fournissant ainsi une estimation de la proportion de contenu informationnel préservée après le rehaussement de la parole. Le STOI attribue un score de qualité à la parole rehaussée, sur une échelle allant de 0 à 1, où une valeur plus élevée indique une meilleure qualité en terme d'intelligibilité.

4.4.3 Note moyenne d'opinion de suppression du bruit profond

Une autre métrique employée dans le cadre de ce projet est le DNSMOS [68], un estimateur développé par Microsoft en 2022 dans le cadre de leur compétition sur la suppression du bruit (*Deep Noise Suppression* - DNS). Le DNSMOS est basé sur un réseau de neurones conventionnel. Il évalue la qualité du rehaussement de la parole en prédisant les notes basées sur les aspects suivants : la qualité du signal de parole (*speech quality* - SIG), le bruit de fond (*background noise quality* - BAK), et la qualité globale de la parole rehaussée (*overall audio quality* - OVRL). Les scores attribués varient de 1 à 5, une valeur plus élevée indiquant une performance supérieure de l'algorithme de rehaussement évalué.

Il est important de noter que cette métrique est relativement récente, ce qui limite le nombre d'études disponibles utilisant un réseau de neurones et évaluant leurs modèles à la fois avec la base de données décrite dans la section 4.1 et la métrique DNSMOS. Par conséquent, les résultats de cette étude seront précieux pour enrichir la littérature scientifique existante dans ce domaine et fournir de nouvelles perspectives sur l'évaluation des performances des modèles de rehaussement de la parole.

4.5 Implémentation

4.5.1 Bibliothèques de programmation

Le présent travail de recherche a été mis en œuvre en utilisant le langage de programmation Python [1]. La version utilisée est Python 3.9. Les bibliothèques de programmation utilisées, leurs noms respectifs et leurs versions correspondantes sont répertoriés dans le Tableau 4.1.

Ces bibliothèques offrent des fonctionnalités essentielles et des outils nécessaires pour la manipulation et l'analyse des données audio, l'entraînement et l'exécution de modèles de réseaux de neurones conventionnels ou à décharges, ainsi que pour l'évaluation des performances des algorithmes proposés.

4.5.2 Matériel utilisé

Dans le cadre de ce projet, les modèles de réseaux neuronaux utilisés ont été entraînés en exploitant les ressources du superordinateur Béluga [2], mis à disposition par l'Alliance de recherche numérique du Canada.

Le superordinateur Béluga est caractérisé par une architecture de pointe, comprenant des processeurs multi-cœurs et des unités de traitement graphique (*Graphics Processing*

1. `https://www.python.org/`
2. `https://docs.alliancecan.ca/wiki/Beluga`

TABLEAU 4.1 Bibliothèques de programmation.

Nom	**Version**
comet_ml	3.31.15
librosa	0.9.2
numpy	1.21.0
onnxruntime	1.12.1
pesq	0.0.4
pystoi	0.3.3
seaborn	0.12.2
soundfile	0.11.0
speechbrain	0.5.13
torch	1.13.1
torchaudio	0.13.1
tqdm	4.64.1

Unit - GPU) hautement performantes. Cette configuration matérielle permet d'exploiter pleinement le potentiel de calcul parallèle offert par les réseaux neuronaux, ce qui accélère significativement le processus d'entraînement.

Conclusion

Dans le présent chapitre, nous avons exposé en détail les conditions expérimentales adoptées pour l'entraînement et l'évaluation du modèle proposé. Différents aspects cruciaux ont été abordés, notamment les caractéristiques de la base de données utilisée, les étapes de prétraitement des données, la configuration de l'entraînement, les métriques d'évaluation, ainsi que les spécifications de mise en œuvre, telles que les bibliothèques Python et le matériel utilisé.

Les conditions expérimentales établies ont permis d'offrir un cadre rigoureux pour l'évaluation de l'approche proposée. Le chapitre subséquent sera consacré à la présentation des résultats obtenus au moyen de ces conditions. Ces résultats permettront d'évaluer l'efficacité de notre approche et de formuler des conclusions significatives quant à ses performances et à son potentiel pour résoudre le problème du rehaussement de la parole.

CHAPITRE 5

Résultats

Ce chapitre présente les résultats obtenus lors de l'évaluation de la performance du SNN proposé pour le rehaussement de la parole. Tout d'abord, une série d'expériences variées est entreprise afin d'analyser l'impact de différents paramètres sur les performances du modèle, offrant ainsi une compréhension approfondie de son comportement et de ses limites. Cette approche permet d'éclairer les facteurs clés qui influencent l'efficacité du rehaussement de la parole à l'aide d'un réseau de neurones à décharges. Par la suite, les avancées réalisées par notre approche sont mises en évidence en comparant nos résultats avec ceux de l'état de l'art.

5.1 Étude des effets de différents paramètres

Cette section vise à étudier les effets de divers paramètres sur le rehaussement de la parole en utilisant le réseau de neurones à décharges proposé. Les paramètres suivants sont examinés afin d'évaluer leur influence sur les performances du système de rehaussement proposé :

1. Fonction de coût : cette partie analyse l'impact de différentes fonctions de coût sur les performances du modèle proposé. Des fonctions de coût largement utilisées dans le domaine du rehaussement de la parole telles que la LSD, la MSE (*Mean Square Error*) et la SI-SNR (*Scale-Invariant Signal-to-Noise Ratio*) sont comparées afin de déterminer celle qui favorise un meilleur rehaussement de la parole en utilisant un SNN.
2. Entraînement des paramètres neuronaux : cette partie étudie l'effet de l'entraînement des paramètres des neurones LIF sur la performance du système proposé, à savoir les constantes de temps et le seuil de décharge.
3. Méthode de sous-échantillonnage : cette partie évalue l'effet des méthodes de sous-échantillonnage. Les méthodes de sous-échantillonnage par convolution, par valeur maximale et par valeur moyenne sont comparées.
4. Méthode de suréchantillonnage : cette partie examine l'impact des méthodes de suréchantillonnage. Les méthodes de suréchantillonnage par les plus proches voisins et par interpolation bilinéaire sont comparées.

5. Type de connexions de saut : cette partie étudie l'influence du type de connexions de saut, à savoir l'addition ou la concaténation.
6. Ajout d'un bloc résiduel : cette partie analyse l'effet de l'ajout d'un ou de deux blocs résiduels dans l'architecture du réseau de neurones.

5.1.1 Fonction de coût

Cette section vise à analyser les effets de trois fonctions de coût différentes, à savoir LSD, MSE et SI-SNR. Les résultats obtenus sont présentés dans le Tableau 5.1.

TABLEAU 5.1 Résultats de l'étude de l'effet de la fonction de coût.

			DNSMOS		
Système	**PESQ**	**STOI**	**OVRL**	**SIG**	**BAK**
LSD	2.66	0.92	2.81	3.13	3.85
MSE	2.59	0.92	2.79	3.11	3.84
SI-SNR	1.63	0.81	1.88	2.06	3.97

Les résultats de cette étude mettent en évidence une légère supériorité de la fonction de coût LSD par rapport à la fonction de coût MSE, pour toutes les métriques à l'exception du STOI où les deux fonctions de coût donnent des valeurs égales. En revanche, la fonction de coût SI-SNR affiche des résultats inférieurs à ceux obtenus avec LSD et MSE, à l'exception du score BAK qui présente une légère amélioration.

Il est important de souligner que la fonction de coût utilisée pour l'approche présentée dans le chapitre 3 et évaluée dans la section 5.2.2 est la fonction de coût LSD.

5.1.2 Entraînement des paramètres neuronaux

Cette section a pour objectif d'analyser l'effet de l'entraînement des paramètres neuronaux, à savoir les constantes de temps et le seuil de décharge (voir section 2.2.3 du chapitre 2), sur les performances du réseau. Les résultats obtenus sont présentés dans le Tableau 5.2.

TABLEAU 5.2 Résultats de l'étude de l'effet d'entraînement des paramètres neuronaux.

			DNSMOS		
Système	**PESQ**	**STOI**	**OVRL**	**SIG**	**BAK**
Avec entraînement des paramètres neuronaux	2.66	0.92	2.81	3.13	3.85
Sans entraînement des paramètres neuronaux	2.53	0.92	2.82	3.16	3.81

L'analyse des résultats révèle que l'optimisation des paramètres neuronaux conduit à des améliorations significatives en termes de PESQ et BAK, des valeurs similaires pour STOI, et une légère diminution pour SIG et OVRL.

Il est important de souligner que pour les résultats d'expérimentations présentées dans la section 5.2.2, l'entraînement des paramètres neuronaux est effectuée.

5.1.3 Méthode de sous-échantillonnage

Dans cette section, une étude est menée pour évaluer les effets associés à la méthode de sous-échantillonnage utilisée pour réduire la taille de l'ensemble de données au niveau de la partie de l'encodeur de l'architecture U-Net. Plus précisément, l'influence de deux approches distinctes est examinée : l'utilisation d'une couche de convolution avec un paramètre de pas d'avancement égal à 2, réduisant ainsi la taille des données d'entrée de moitié à chaque couche, ou l'emploi d'une couche de sous-échantillonnage basée sur la valeur maximale ou moyenne (voir section 2.2.2). Les couches de sous-échantillonnage ont pour objectif principal de réduire progressivement la résolution des caractéristiques en entrée, tout en conservant les informations les plus pertinentes. Cela est réalisé en calculant la valeur maximale ou moyenne de l'entrée à l'aide d'une fenêtre glissante. Les résultats obtenus sont présentés dans le Tableau 5.3.

TABLEAU 5.3 Résultats de l'étude de l'effet de la méthode de sous-échantillonnage.

Système	**PESQ**	**STOI**	**DNSMOS**		
			OVRL	**SIG**	**BAK**
Sous-échantillonnage par convolution	2.66	0.92	2.81	3.13	3.85
Sous-échantillonnage par valeur maximale	2.13	0.89	2.61	2.93	3.79
Sous-échantillonnage par valeur moyenne	1.82	0.86	2.43	2.73	3.74

L'analyse des résultats démontre que le sous-échantillonnage par convolution présente une performance supérieure à celle obtenue avec la couche de sous-échantillonnage basée sur la valeur maximale. De plus, le sous-échantillonnage par la valeur maximale se révèle plus performante que celui par la valeur moyenne, selon toutes les métriques évaluées.

Il est à noter que la méthode de sous-échantillonnage utilisée pour l'approche présentée dans le chapitre 3 et évaluée dans la section 5.2.2 est le sous-échantillonnage par convolution.

5.1.4 Méthode de suréchantillonnage

Cette section se focalise sur l'investigation de l'impact de la méthode de suréchantillonnage visant à augmenter la taille de l'ensemble de données au niveau de la partie du décodeur de l'architecture U-Net. Plus spécifiquement, la méthode basée sur les plus proches voisins et celle basée sur l'interpolation bilinéaire (voir section 2.2.2). La méthode basée sur les plus proches voisins consiste à dupliquer chaque élément de l'entrée en utilisant la valeur du pixel le plus proche. En d'autres termes, chaque pixel est répété pour augmenter la taille de l'ensemble de données. En revanche, la méthode basée sur l'interpolation bilinéaire calcule une moyenne pondérée des quatre éléments les plus proches dans l'entrée pour générer de nouveaux pixels. Il convient de noter que la méthode de convolution transposée n'a pas été testée, car elle est connue pour provoquer des artéfacts en damier (*checkerboard artifacts*) [59]. Les résultats de cette analyse sont présentés dans le Tableau 5.4.

TABLEAU 5.4 Résultats de l'étude de la méthode de suréchantillonnage.

			DNSMOS		
Système	**PESQ**	**STOI**	**OVRL**	**SIG**	**BAK**
Plus proche voisin	2.66	0.92	2.81	3.13	3.85
Interpolation bilinéaire	2.49	0.92	2.78	3.11	3.81

Les résultats obtenus mettent en évidence que la méthode de suréchantillonnage avec les plus proches voisins se distingue par des performances supérieures pour l'ensemble des métriques évaluées, à l'exception de STOI, qui est identique pour les deux approches.

Il est à souligner que la méthode de suréchantillonnage utilisée pour l'approche présentée dans le chapitre 3 et évaluée dans la section 5.2.2 est le suréchantillonnage par les plus proches voisins.

5.1.5 Type de connexions de saut

Cette section examine l'impact du type de connexions de saut (voir section 2.2.2), à savoir la concaténation ou l'addition, sur les performances du système. Les connexions de saut facilitent la transmission d'informations entre l'encodeur et le décodeur de l'architecture U-Net, favorisant ainsi une convergence plus rapide et une stabilité accrue de l'apprentissage. Les résultats obtenus sont présentés dans le Tableau 5.5.

L'analyse des résultats révèle que l'utilisation de connexions de saut basées sur la concaténation conduit à des améliorations significatives des performances. Cette observation suggère que la concaténation des caractéristiques extraites à différents niveaux du réseau enrichit les représentations et favorise une amélioration plus efficace de la parole rehaussée par rapport à l'addition.

TABLEAU 5.5 Résultats de l'étude de l'effet du type de connexions de saut.

Système	PESQ	STOI	DNSMOS		
			OVRL	SIG	BAK
Concaténation	2.66	0.92	2.81	3.13	3.85
Addition	2.51	0.91	2.74	3.05	3.83

Le type de connexions de saut utilisé pour l'approche présentée dans le chapitre 3 et évaluée dans la section 5.2.2 est la concaténation.

5.1.6 Ajout d'un bloc résiduel

Dans cette section, nous procédons à une analyse des effets de l'ajout d'un ou de deux blocs résiduels dans le modèle proposé. L'incorporation d'un bloc résiduel est une méthode largement utilisée dans la littérature pour favoriser une propagation efficace des informations et atténuer les problèmes de rétropropagation du gradient, tels que la disparition ou l'explosion du gradient [25]. La Figure 5.1 illustre l'architecture du réseau de neurones à décharges avec un bloc résiduel inséré entre l'encodeur et le décodeur.

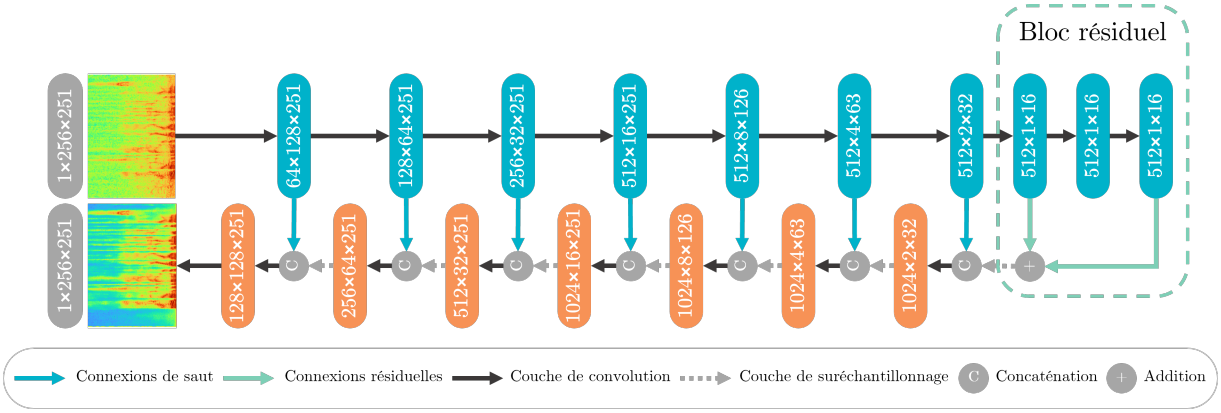

FIGURE 5.1 Architecture du réseau de neurones à décharges proposé avec un bloc résiduel.

Les résultats pertinents issus de cette étude sont présentés dans le Tableau 5.6.

TABLEAU 5.6 Résultats de l'étude de l'effet d'ajout d'un bloc résiduel.

Système	PESQ	STOI	DNSMOS		
			OVRL	SIG	BAK
Sans bloc résiduel	2.66	0.92	2.81	3.13	3.85
Avec un bloc résiduel	2.59	0.92	2.82	3.16	3.83
Avec deux blocs résiduels	2.65	0.92	2.79	3.11	3.85

Les résultats obtenus mettent en évidence que l'incorporation d'un bloc résiduel ne conduit pas à une amélioration significative de la performance, à l'exception d'une légère augmentation observée pour les métriques OVRL et SIG. De manière similaire, l'ajout de deux blocs résiduels ne présente aucune amélioration notable de la performance. Ces résultats suggèrent que l'ajout de blocs résiduels ne contribue pas de manière significative à l'amélioration de la qualité de la parole rehaussée pour le modèle proposé.

Il est à indiquer que l'architecture proposée dans le chapitre 3 et évaluée dans la section 5.2.2 ne comprend pas de bloc résiduel.

5.2 Comparaison à l'état de l'art

5.2.1 Systèmes de comparaison

Cette section met en évidence les systèmes de rehaussement de la parole utilisés pour la comparaison. Nous considérons le signal bruité comme référence pour ces évaluations.

Tout d'abord, nous considérons une approche classique, à savoir la méthode de Wiener [42]. Aussi, nous évaluons plusieurs modèles de rehaussement de la parole utilisant des réseaux de neurones conventionnels, tels que SEGAN [61], Wave-U-Net [46], MMSE-GAN [98], D+M [102], et U-Net [13]. La plupart de ces modèles ont été présentés dans la section 2.3.2 du chapitre 2. De plus, nous comparons les performances d'un réseau de neurones conventionnel ayant une architecture similaire à celle du réseau de neurones à décharges proposé. En outre, nous présentons les performances du SDNN [86], un réseau de neurones à décharges introduit par Intel et abordé dans la section 2.3.3 du chapitre 2. Les résultats expérimentaux de la comparaison sont ensuite présentés dans la section suivante.

5.2.2 Résultats expérimentaux

Les performances du système proposé ont été évaluées en le comparant à plusieurs modèles de l'état de l'art largement étudiés dans le domaine du rehaussement de la parole. Il convient de rappeler que les résultats expérimentaux ont été obtenus en utilisant différentes métriques d'évaluation, notamment le PESQ, le STOI, ainsi que les estimations de scores subjectifs obtenus par les métriques objectives DNSMOS : OVRL, SIG et BAK (voir section 4.4).

Les résultats de la comparaison entre le système proposé et les autres modèles sont présentés dans le Tableau 5.7. Il convient de noter que la plupart des résultats de l'état de l'art proviennent des travaux de Tran *et al.* [87]. Par ailleurs, il est important de souligner que le tableau présenté se révèle incomplet principalement parce que certaines des études citées n'ont pas recouru aux métriques spécifiées, soit pour d'autres motifs d'ordre méthodolo-

gique, soit parce qu'elles n'étaient pas encore disponibles à l'époque de leurs recherches. Par exemple, DNSMOS, un estimateur qui prédit les notes OVRL, SIG et BAK, n'a été intégré qu'en 2022, justifiant ainsi son absence dans les travaux antérieurs.

TABLEAU 5.7 Résultats de la comparaison du SNN propsé avec différents modèles de l'état de l'art. Les cellules vides indiquent que la métrique correspondante n'a pas été rapportée ou évaluée pour le modèle concerné.

			DNSMOS		
Système	**PESQ**	**STOI**	**OVRL**	**SIG**	**BAK**
Signal bruité	1.97	0.92	2.69	3.34	3.12
Wiener [42]	2.22	-	-	-	-
SEGAN [61]	2.16	-	-	-	-
Wave-U-Net [46]	2.40	-	-	-	-
MMSE-GAN [98]	2.53	-	-	-	-
D+M [102]	2.73	-	-	-	-
U-Net [13]	2.90	0.93	-	-	-
ANN équivalent	2.89	0.94	2.92	3.23	3.90
SDNN [86]	2.00	0.91	2.44	3.05	3.09
SNN proposé	2.66	0.92	2.81	3.13	3.85

Tout d'abord, il est observé que le système proposé présente des performances supérieures sur la plupart des métriques d'évaluation par rapport aux signaux de parole bruités non traités. Cette amélioration significative met en évidence l'efficacité de l'approche proposée pour le problème de rehaussement de la parole.

En outre, les résultats expérimentaux démontrent que l'approche proposée surpasse plusieurs modèles de l'état de l'art, y compris la méthode de Wiener ainsi que des modèles basés sur des réseaux de neurones conventionnels tels que SEGAN, Wave-U-Net, et MMSE-GAN, en termes de PESQ. En ce qui concerne le STOI, l'approche proposée atteint une performance comparable au modèle U-Net.

Aussi, une comparaison de l'architecture SNN proposée avec une architecture équivalente basée sur un réseau de neurones conventionnel a été effectuée. Les résultats indiquent que l'approche SNN proposée obtient des performances légèrement inférieures, mais toujours comparables en termes de DNSMOS. Cela démontre ainsi l'efficacité de l'approche SNN pour le rehaussement de parole, tout en utilisant moins de ressources computationnelles que l'approche basée sur les ANN.

Une évaluation détaillée a été réalisée pour comparer le modèle SNN proposé avec une approche récemment publiée basée sur les SNN, spécifiquement le modèle SDNN de référence introduit dans l'article [86] pour la compétition d'Intel, N-DNS. Afin d'assurer une

comparaison cohérente, le modèle SDNN a été entraîné et évalué en utilisant le même ensemble de données que le modèle proposé. Les résultats montrent que le SNN basé sur U-Net proposé surpasse significativement le modèle SDNN de référence sur toutes les métriques d'évaluation. Il convient de souligner que les auteurs de l'article [86] rapportent de meilleurs résultats dans leur étude originale. Cependant, il est important de noter que le modèle SDNN a été entraîné sur une base de données plus vaste, comprenant 500 heures de données audio, tandis que notre étude utilise un ensemble de données plus restreint, avec environ 10 heures de données audio. L'approche proposée semble moins sensible à ces limitations de taille et de variabilité de l'ensemble de données que le modèle SDNN.

En conclusion, les résultats expérimentaux confirment l'efficacité du modèle basé sur un SNN proposé pour le rehaussement de la parole, démontrant des performances supérieures par rapport à plusieurs modèles de l'état de l'art, évalués selon diverses métriques. Bien que le modèle proposé présente une performance légèrement inférieure à une architecture équivalente basée sur un ANN, il demeure néanmoins très compétitif. De plus, le modèle proposé surpasse de manière significative le modèle d'Intel, SDNN.

Conclusion

Une série d'expérimentations a été menée pour étudier l'influence de divers paramètres sur les performances du modèle. Les conclusions tirées de ces expérimentations soulignent que la meilleure performance est obtenue lorsque la fonction de coût LSD est utilisée, les paramètres neuronaux sont entraînés, le sous-échantillonnage est effectué par convolution, le suréchantillonnage est réalisé à l'aide de la méthode du plus proche voisin, les connexions de saut sont basées sur la concaténation, et enfin, une architecture U-Net sans bloc résiduel entre l'encodeur et le décodeur est adoptée. Cette architecture a été conservée pour la suite de l'étude en raison de ses performances supérieures.

En outre, ce chapitre a exposé les performances expérimentales du modèle proposé pour le rehaussement de la parole, en les comparant à celles d'une approche classique, des modèles de l'état de l'art, d'un réseau de neurones conventionnel d'une architecture similaire, ainsi que d'un réseau de neurones à décharges. Les résultats obtenus démontrent l'efficacité du modèle proposé, qui a surpassé les modèles de l'état de l'art dans plusieurs métriques. Bien que légèrement inférieure à celle de l'ANN d'architecture similaire, la performance du modèle proposé reste néanmoins comparable. De plus, la performance du modèle proposé dépasse significativement celle du SDNN.

En somme, ces résultats confirment la pertinence du modèle proposé pour le rehaussement de la parole et mettent en évidence l'importance des paramètres et des choix architecturaux dans l'obtention de performances optimales.

CHAPITRE 6

Conclusion

Dans cette section, nous présenterons un résumé concis des principaux éléments abordés tout au long de ce mémoire. Ensuite, nous mettrons en évidence les contributions majeures de ce travail. Enfin, nous examinerons les perspectives de recherche futures qui émergeront naturellement de cette étude approfondie, ouvrant ainsi de nouvelles voies pour des avancées significatives dans le domaine du rehaussement de la parole à l'aide de réseau de neurones à décharges.

6.1 Sommaire

Le rehaussement de la parole est un problème qui a été largement étudié en utilisant les réseaux de neurones conventionnels. Toutefois, l'émergence des réseaux de neurones à décharges offre une alternative prometteuse, caractérisée par des performances similaires dans plusieurs domaines tout en consommant moins d'énergie. Cependant, peu d'études ont exploité une approche neuromorphique pour le rehaussement de la parole. Par conséquent, l'objectif principal de cette étude était d'évaluer les capacités des réseaux de neurones à décharges pour le rehaussement de la parole, et de déterminer s'ils peuvent atteindre des performances comparables à celles des réseaux de neurones conventionnels.

L'approche proposée repose sur une architecture en U-Net intégrant des neurones LIF afin de rehausser la parole. Les expérimentations menées dans le cadre de cette étude ont démontré que les réseaux de neurones à décharges peuvent atteindre des performances légèrement inférieures, mais comparables à celles des réseaux de neurones conventionnels. Ces résultats suggèrent que les SNN peuvent constituer une solution viable pour le rehaussement de la parole, offrant des avantages en termes de consommation d'énergie sans compromettre de manière significative les performances.

6.2 Contributions

La présente étude a apporté plusieurs contributions significatives dans le domaine du rehaussement de la parole en utilisant un réseau de neurones à décharges. Les principales contributions sont les suivantes :

1. **Estimation directe :** À notre connaissance, il s'agit de la première architecture utilisant une approche d'estimation directe pour le rehaussement de la parole en utilisant un SNN.

2. **Architecture profonde :** Contrairement aux travaux précédents qui se sont concentrées sur des architectures peu profondes pour le rehaussement de la parole avec un SNN, ce travail présente une architecture profonde basée sur U-Net, et spécifiquement conçue pour le rehaussement de la parole.

3. **Encodage direct :** Les approches antérieures de rehaussement faisant appel à des SNN ont recours à des méthodes d'encodage pour convertir les données d'entrée continues en décharges. Ce travail propose d'utiliser la méthode d'encodage directe qui permet à la première couche du SNN d'apprendre à encoder l'entrée tout en éliminant simultanément le bruit indésirable.

4. **Entraînement de paramètres neuronaux :** Alors que l'entraînement traditionnel des réseaux de neurones conventionnels se limite généralement à l'optimisation des poids du réseau, la méthode proposée, exploitant un réseau de neurones à décharges, étend cette approche en incluant l'entraînement simultané des poids ainsi que des paramètres intrinsèques aux neurones LIF, à savoir les constantes de temps et le seuil de décharge. Cette démarche enrichit le modèle en lui conférant une capacité d'adaptation plus fine aux caractéristiques inhérentes des données, tout en consolidant sa faculté de généralisation aux tâches complexes.

5. **Étude d'ablation :** Une étude d'ablation a été menée pour évaluer l'impact de différents paramètres du modèle proposé sur la performance du réseau.

6. **Comparaison avec des ANN d'architecture similaire :** Une comparaison est effectuée entre le SNN proposé et un ANN présentant une architecture similaire permettant de garantir une comparaison adéquate entre l'approche par SNN et celle par ANN.

Les résultats obtenus dans cette étude ouvrent des perspectives prometteuses pour améliorer l'efficacité énergétique tout en maintenant des performances comparables à celles des systèmes n'utilisant pas des réseaux de neurones à décharges, notamment les réseaux de neurones conventionnels. Cependant, il convient de noter que cet aspect n'a pas été examiné dans le cadre de ce mémoire.

6.3 Travaux futurs

6.3.1 Base de données

L'un des aspects clés pour améliorer davantage le rehaussement de la parole à l'aide d'un réseau de neurones à décharges réside dans l'utilisation d'une base de données à grande échelle. Traditionnellement, une base de données plus vaste et représentative a été associée à une meilleure généralisation des modèles et à des performances plus robustes dans l'apprentissage supervisé utilisant la rétropropagation, qui repose sur des statistiques. Cependant, il convient de noter que certains réseaux de neurones à décharges adoptent des approches non statistiques, et dans de tels cas, l'impact de la taille de la base de données sur les performances peut différer.

6.3.2 Extraction des caractéristiques

Dans ce travail, l'utilisation de la transformée de Fourier à court terme a été considérée, en raison de sa large adoption et de son efficacité. Cependant, afin de tirer pleinement parti des capacités des SNN, il est nécessaire d'explorer des méthodes d'extraction de caractéristiques plus compatibles et adaptées à ce type de réseau telles que l'algorithme localement compétitif (*Locally Competitive Algorithm*) [73].

6.3.3 Architecture du système

Dans le cadre de ce travail, une architecture de réseau convolutionnel U-Net a été utilisée pour concevoir le système. Cependant, il existe d'autres architectures prometteuses qui pourraient être explorées dans les travaux futurs, notamment les transformateurs [90]. Les transformateurs sont des modèles de réseau de neurones récemment introduits qui ont montré des performances impressionnantes dans de nombreux domaines [43]. Il est toutefois impératif de noter que le choix entre ces architectures doit être guidé par la nature spécifique de la tâche et par l'inclusion ou non de mécanismes d'attention dans le cadre de réseaux récurrents, car cela peut influencer la pertinence des transformateurs pour la problématique en question.

En outre, cette étude s'est concentrée sur l'utilisation de neurones LIF. Cependant, il existe plusieurs autres modèles de neurones à décharges qui peuvent être étudiés et comparés afin de mieux comprendre leurs caractéristiques et leur impact sur le problème de rehaussement de la parole.

Il convient également de noter que l'optimisation de l'architecture du système proposé peut être abordée en considérant d'autres facteurs, tels que l'ajout de couches supplémentaires, l'ajustement des hyperparamètres, ainsi que l'exploration d'autres méthodes d'initialisation.

6.3.4 Déploiement du modèle sur un processeur neuromorphique

Pour pouvoir valider l'avantage d'efficience énergétique des SNN en comparaison avec les ANN, il serait nécessaire d'implémenter le SNN proposé sur un processeur neuromorphique. Dans le cadre des travaux futurs, une direction prometteuse consisterait donc à valider l'avantage en termes d'efficience énergétique des réseaux de neurones à décharges par rapport aux réseaux de neurones conventionnels à l'aide d'un processeur neuromorphique.

LISTE DES RÉFÉRENCES

[1] L. F. Abbott. Lapicque's introduction of the integrate-and-fire model neuron (1907). *Brain Research Bulletin*, 50(5-6) :303–304, 1999.

[2] O. I. Abiodun, A. Jantan, A. E. Omolara, et al. State-of-the-art in artificial neural network applications : A survey. *Heliyon*, 4(11) :e00938, 2018.

[3] J. Benesty, S. Makino, and J. Chen. *Speech Enhancement*. Springer, 2005.

[4] B. V. Benjamin, P. Gao, E. McQuinn, et al. Neurogrid : A mixed-analog-digital multichip system for large-scale neural simulations. *Proceedings of the IEEE*, 102(5) :699–716, 2014.

[5] G. Bi and M. Poo. Synaptic modifications in cultured hippocampal neurons : Dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience*, 18(24) :10464–10472, 1998.

[6] C.M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, USA, 1995.

[7] S. Boll. A spectral subtraction algorithm for suppression of acoustic noise in speech. In *ICASSP '79. IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 200–203, 1979.

[8] L. Bottou, C. Cortes, J.S. Denker, et al. Comparison of classifier methods : a case study in handwritten digit recognition. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition, Vol. 3 - Conference C : Signal Processing*, volume 2, pages 77–82, 1994.

[9] A. S. Bregman. *Auditory Scene Analysis : The Perceptual Organization of Sound*. The MIT Press, 05 1990.

[10] A. W. Bronkhorst. The cocktail party phenomenon : A review of research on speech intelligibility in multiple-talker conditions. *Acta Acustica united with Acustica*, 86(1) :117–128, 2000.

[11] G. J. Brown and M. Cooke. Computational auditory scene analysis. *Computer Speech Language*, 8(4) :297–336, 1994.

[12] T. Brown, B. Mann, N. Ryder, et al. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020.

[13] A. E. Bulut and K. Koishida. Low-latency single channel speech enhancement using U-Net convolutional neural networks. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6214–6218, 2020.

[14] M. Cooke, P. Green, L. Josifovski, et al. Robust automatic speech recognition with missing and unreliable acoustic data. *Speech Communication*, 34(3) :267–285, 2001.

[15] F. Dang, H. Chen, and P. Zhang. Dpt-fsnet : Dual-path transformer based full-band and sub-band fusion network for speech enhancement. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6857–6861, 2022.

[16] M. Davies, N. Srinivasa, T. Lin, et al. Loihi : A neuromorphic manycore processor with on-chip learning. *IEEE Micro*, 38(1) :82–99, 2018.

[17] M.E. Deisher and A.S. Spanias. Hmm-based speech enhancement using harmonic modeling. In *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 1175–1178 vol.2, 1997.

[18] J. Devlin, M. Chang, K. Lee, et al. BERT : Pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[19] S. Dhar, J. Guo, J. Liu, et al. On-device machine learning : An algorithms and learning theory perspective. *CoRR*, abs/1911.00623, 2019.

[20] P. U. Diehl, D. Neil, J. Binas, et al. Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2015.

[21] M. Drozdzal, E. Vorontsov, G. Chartrand, et al. The importance of skip connections in biomedical image segmentation. In *Deep Learning and Data Labeling for Medical Applications*, pages 179–187, Cham, 2016. Springer International Publishing.

[22] S. Dupond. A thorough review on the current advance of neural network structures. *Annual Reviews in Control*, 14 :200–230, 2019.

[23] Y. Ephraim and D. Malah. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(6) :1109–1121, 1984.

[24] S. K. Esser, P. A. Merolla, J. V. Arthur, et al. Convolutional networks for fast, energy-efficient neuromorphic computing. *Proceedings of the national academy of sciences*, 113(41) :11441–11446, 2016.

[25] W. Fang, Z. Yu, Y. Chen, et al. Deep residual learning in spiking neural networks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 21056–21069. Curran Associates, Inc., 2021.

[26] R. FitzHugh. Impulses and physiological states in theoretical models of nerve membrane. *Biophysical journal*, 1(6) :445–466, 1961.

[27] S. B. Furber, F. Galluppi, S. Temple, et al. The spinnaker project. *Proceedings of the IEEE*, 102(5) :652–665, 2014.

[28] I. Goodfellow, J. Pouget-Abadie, M. Mirza, et al. Generative adversarial networks. *Commun. ACM*, 63(11) :139–144, oct 2020.

[29] J. L. Hindmarsh, R. M. Rose, and A. F. Huxley. A model of neuronal bursting using three coupled first order differential equations. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 221(1222) :87–102, 1984.

[30] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8) :1735–1780, 1997.

[31] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4) :500–544, 1952.

[32] Y. Hu, Y. Liu, S. Lv, et al. DCCRN : Deep complex convolution recurrent network for phase-aware speech enhancement, 2020.

[33] E.M. Izhikevich. Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, 14(6) :1569–1572, 2003.

[34] E.M. Izhikevich. Which model to use for cortical spiking neurons ? *IEEE Transactions on Neural Networks*, 15(5) :1063–1070, 2004.

[35] X. Jia, E. Gavves, B. Fernando, et al. Guiding the long-short term memory model for image caption generation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[36] N. Kasabov. Deep learning in spiking neural networks for brain-inspired artificial intelligence. In *Proceedings of the 19th International Conference on Computer Systems and Technologies*, CompSysTech'18, New York, NY, USA, 2018.

[37] M. B. Kennedy. Synaptic signaling in learning and memory. *Cold Spring Harbor perspectives in biology*, 8(2) :a016824, 2016.

[38] S. Khan, M. H. Javed, E. Ahmed, et al. Facial recognition using convolutional neural networks and implementation on smart glasses. In *2019 International Conference on Information Science and Communication Technology (ICISCT)*, pages 1–6, 2019.

[39] D. P. Kingma and J. Ba. Adam : A method for stochastic optimization. *arXiv :1412.6980*, 2014.

[40] T. Kounovsky and J. Malek. Single channel speech enhancement using convolutional neural network. In *2017 IEEE International Workshop of Electronics, Control, Measurement, Signals and their Application to Mechatronics (ECMSM)*, pages 1–5, 2017.

[41] C. KyungHyun, V. M. Bart, B. Dzmitry, et al. On the properties of neural machine translation : Encoder-decoder approaches. *CoRR*, abs/1409.1259, 2014.

[42] J. Lim and A. Oppenheim. All-pole modeling of degraded speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(3) :197–210, 1978.

[43] T. Lin, y. Wang, X. Liu, et al. A survey of transformers. *AI Open*, 3 :111–132, 2022.

[44] G. Litjens, T. Kooi, B. E. Bejnordi, et al. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42 :60–88, 2017.

[45] S. Lu and A. Sengupta. Exploring the connection between binary and spiking neural networks. *Frontiers in Neuroscience*, 14 :535, 2020.

[46] C. Macartney and T. Weyde. Improved speech enhancement with the Wave-U-Net. *CoRR*, abs/1811.11307, 2018.

[47] D. S. Maitra, U. Bhattacharya, and S. K. Parui. CNN based common approach to handwritten character recognition of multiple scripts. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1021–1025, 2015.

[48] R. McAulay and M. Malpass. Speech enhancement using a soft-decision noise suppression filter. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(2) :137–145, 1980.

[49] K. Meier. A mixed-signal universal neuromorphic computing system. In *2015 IEEE International Electron Devices Meeting (IEDM)*, pages 4.6.1–4.6.4, 2015.

[50] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, et al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, 345(6197) :668–673, 2014.

[51] M. Minsky and S. Papert. *Perceptrons : An Introduction to Computational Geometry*. MIT Press, Cambridge, MA, USA, 1969.

[52] N. Mohammadiha, T. Gerkmann, and A. Leijon. A new approach for speech enhancement based on a constrained nonnegative matrix factorization. In *2011 International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS)*, pages 1–5, 2011.

[53] S. K. Moore. Intel's neuromorphic system hits 8 million neurons, 100 million coming by 2020. `https://spectrum.ieee.org/intels-neuromorphic-system-hits-8-million-neurons-100-million-coming-by-2020`, Jun 2021.

[54] C. Morris and H. Lecar. Voltage oscillations in the barnacle giant muscle fiber. *Biophysical Journal*, 35(1) :193–213, 1981.

[55] E. O. Neftci, H. Mostafa, and F. Zenke. Surrogate gradient learning in spiking neural networks : Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6) :51–63, 2019.

[56] S. A. Nossier, J. Wall, M. Moniri, et al. Mapping and masking targets comparison using different deep learning based speech enhancement architectures. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2020.

[57] P. O'Connor and M. Welling. Sigma delta quantized networks. *CoRR*, abs/1611.02024, 2016.

[58] B. O. Odelowo and D. V. Anderson. A study of training targets for deep neural network-based speech enhancement using noise prediction. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5409–5413, 2018.

[59] A.s Odena, V. Dumoulin, and C. Olah. Deconvolution and checkerboard artifacts. *Distill*, 1(10) :e3, 2016.

[60] A. Paolo, B. Adriano, B. Maide, et al. Image processing for medical diagnosis using CNN. *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, 497(1) :174–178, 2003. First International Symposium on Functional Breast Imaging with Advanced Detectors.

[61] S. Pascual, A. Bonafonte, and J. Serrà. SEGAN : speech enhancement generative adversarial network. *CoRR*, abs/1703.09452, 2017.

[62] R. Patil. Noise reduction using wavelet transform and singular vector decomposition. *Procedia Computer Science*, 54 :849–853, 12 2015.

[63] R. Pichevar and J. Rouat. Monophonic sound source separation with an unsupervised network of spiking neurones. *Neurocomputing*, 71(1) :109–120, 2007. Dedicated Hardware Architectures for Intelligent Systems Advances on Neural Networks for Speech and Audio Processing.

[64] F. Pineda. Generalization of back propagation to recurrent and higher order neural networks. In D. Anderson, editor, *Neural Information Processing Systems*. American Institute of Physics, 1987.

[65] F. Ponulak and A. Kasinski. Introduction to spiking neural networks : Information processing, learning and applications. *Acta Neurobiologiae Experimentalis*, 71(4) :409–433, 2011.

[66] J. W. Rae, S. Borgeaud, T. Cai, et al. Scaling language models : Methods, analysis & insights from training gopher. *CoRR*, abs/2112.11446, 2021.

[67] N. Rathi and K. Roy. DIET-SNN : A low-latency spiking neural network with direct input encoding and leakage and threshold optimization. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–9, 2021.

[68] C. K. A. Reddy, V. Gopal, and R. Cutler. DNSMOS p.835 : A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 886–890, 2022.

[69] A. Riahi and É. Plourde. Single channel speech enhancement using U-Net spiking neural networks. In *2023 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, 5 p., 2023, soumis pour publication.

[70] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra. Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings*, volume 2, pages 749–752, 2001.

[71] O. Ronneberger, P. Fischer, and T. Brox. U-Net : Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

[72] F. Rosenblatt. The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6) :386, 1958.

[73] C. J. Rozell, D. H. Johnson, R. G. Baraniuk, et al. Sparse coding via thresholding and local competition in neural circuits. *Neural Computation*, 20(10) :2526–2563, 2008.

[74] B. Rueckauer, I. Lungu, Y. Hu, et al. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Frontiers in Neuroscience*, 11 :682, 2017.

[75] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088) :533–536, 1986.

[76] B. Schrauwen and J. Van Campenhout. BSA, a fast and accurate spike train encoding scheme. In *Proceedings of the International Joint Conference on Neural Networks, 2003.*, volume 4, pages 2825–2830 vol.4, 2003.

[77] S. Sengupta, S. Basak, P. Saikia, et al. A review of deep learning with special emphasis on architectures, applications and recent trends. *Knowledge-Based Systems*, 194 :105596, 2020.

[78] A. Shewalkar, D. Nyavanandi, and S. A. Ludwig. Performance evaluation of deep neural networks applied to speech recognition : RNN, LSTM and GRU. *Journal of Artificial Intelligence and Soft Computing Research*, 9(4) :235–245, 2019.

[79] S. Srinivasan, N. Roman, and D. Wang. Binary and ratio time-frequency masks for robust speech recognition. *Speech Communication*, 48(11) :1486–1501, 2006. Robustness Issues for Conversational Interaction.

[80] E. Strubell, A. Ganesh, and A. McCallum. Energy and policy considerations for deep learning in NLP. *CoRR*, abs/1906.02243, 2019.

[81] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[82] C. H. Taal, R. C. Hendriks, R. Heusdens, et al. An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7) :2125–2136, 2011.

[83] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, et al. Deep learning in spiking neural networks. *Neural Networks*, 111 :47–63, 2019.

[84] A. Tealab. Time series forecasting using artificial neural networks methodologies : A systematic review. *Future Computing and Informatics Journal*, 3(2) :334–340, 2018.

[85] J. Thiemann, N. Ito, and E. Vincent. The Diverse Environments Multi-channel Acoustic Noise Database (DEMAND) : A database of multichannel environmental noise recordings. *Proceedings of Meetings on Acoustics*, 19(1), 06 2013. 035081.

[86] J. Timcheck, S. B. Shrestha, D. B. D. Rubin, et al. The Intel neuromorphic DNS challenge. *arXiv :2303.09503*, 2023.

[87] D. N. Tran and K. Koishida. Single-channel speech enhancement by subspace affinity minimization. In *Proc. Interspeech 2020*, pages 2447–2451, 2020.

[88] C. Valentini-Botinhao. Noisy speech database for training speech enhancement algorithms and tts models. *University of Edinburgh. School of Informatics. Centre for Speech Technology Research (CSTR)*, 2017.

[89] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi. Speech Enhancement for a Noise-Robust Text-to-Speech Synthesis System Using Deep Recurrent Neural Networks. In *Proc. Interspeech 2016*, pages 352–356, 2016.

[90] A Vaswani, N Shazeer, N Parmar, et al. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[91] C. Veaux, J. Yamagishi, and S. King. The voice bank corpus : Design, collection and data analysis of a large regional accent speech database. In *2013 International Conference Oriental COCOSDA held jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)*, pages 1–4, 2013.

[92] J. Wall, C. Glackin, N. Cannings, et al. Recurrent lateral inhibitory spiking networks for speech enhancement. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 1023–1028, 2016.

[93] D. Wang. *On Ideal Binary Mask As the Computational Goal of Auditory Scene Analysis*, pages 181–197. Springer US, Boston, MA, 2005.

[94] D. L. Wang and G.J. Brown. Separation of speech from interfering sounds based on oscillatory correlation. *IEEE Transactions on Neural Networks*, 10(3) :684–697, 1999.

[95] T. Wang, X. Xu, J. Xiong, et al. ICA-UNet : ICA inspired statistical unet for real-time 3D cardiac cine MRI segmentation. In *2020 Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 447–457, Cham, 2020. Springer International Publishing.

[96] Y. Wang, A. Narayanan, and D. Wang. On training targets for supervised speech separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(12) :1849–1858, 2014.

[97] P. Werbos. Beyond regression : new tools for prediction and analysis in the behavioral sciences. 1974.

[98] D. S. Williamson and D. Wang. Time-frequency masking in the complex domain for speech dereverberation and denoising. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(7) :1492–1501, 2017.

[99] D. S. Williamson, Y. Wang, and D. Wang. Complex ratio masking for monaural speech separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(3) :483–492, 2016.

[100] Y. Xing, W. Ke, G. Di Caterina, et al. Noise reduction using neural lateral inhibition for speech enhancement. *International Journal of Machine Learning and Computing*, 2019.

[101] Y. Xu, J. Du, L. R. Dai, et al. A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(1) :7–19, 2015.

[102] J. Yao and A. Al-Dahle. Coarse-to-fine optimization for speech enhancement. *CoRR*, abs/1908.08044, 2019.

[103] M. Ye and H. Wan. Improved transformer-based dual-path network with amplitude and complex domain feature fusion for speech enhancement. *Entropy*, 25(2), 2023.

[104] F. Zenke and S. Ganguli. Superspike : Supervised learning in multilayer spiking neural networks. *Neural Computation*, 30(6) :1514–1541, 2018.

[105] F. Zenke and T. P. Vogels. The remarkable robustness of surrogate gradient learning for instilling complex function in spiking neural networks. *Neural Computation*, 33(4) :899–925, 2021.

[106] X. L. Zhang and D. L. Wang. A deep ensemble learning method for monaural speech separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(5) :967–977, 2016.