Université du Québec
à Rimouski

CLASSIFICATION DES SONS RESPIRATOIRES PAR RÉSEAUX DE NEURONES À APPRENTISSAGE PROFOND

Mémoire présenté

dans le cadre du programme de maîtrise en ingénierie

en vue de l'obtention du grade de maître ès sciences appliquées (M. Sc. A.)

PAR

© Hassen Chanane

[Août 2022]

Composition du jury :

Yacine Yaddaden (PhD.), président du jury, Université du Québec à Rimouski

Mohammed Bahoura (PhD.), directeur de recherche, Université du Québec à Rimouski

Hassan Ezzaidi (PhD.), examinateur externe, Université du Québec à Chicoutimi

Dépôt initial le 27 juin 2022 Dépôt final le 16 août 2022

Je dédie ce travail à mes parents,
mes frères et sœurs, et mes proches.

REMERCIEMENTS

Au terme de ce travail, je tiens à remercier mon directeur de recherche. Monsieur Mohammed Bahoura, professeur au département de mathématiques, d'informatique et de génie (DMIG) de l'Université du Québec à Rimouski (UQAR), pour m'avoir donné l'opportunité d'effectuer cette recherche. Le grand intérêt qu'il a démontré, sa disponibilité ainsi que son évaluation des différentes versions de mes articles qui m'a permis de mener ce projet à terme. Je désire exprimer ma reconnaissance et remercier toutes les autres personnes qui de près ou de loin m'ont aidé dans la réalisation de mes travaux.

J'exprime toute ma gratitude au professeur Yacine Yaddaden d'avoir accepté d'être le président du jury pour l'évaluation de mon mémoire. ´

J'adresse également mes plus vifs remerciements au professeur Hassan Ezzaidi d'avoir accepté d'examiner mon travail.

Cette étude a été rendue possible grâce au support financier du Conseil de Recherches en Sciences Naturelles et en Génie (CRSNG).

AVANT-PROPOS

Ce mémoire rentre dans le cadre de l'obtention du diplôme de maître en sciences appliquées, en ingénierie. Cette étude a pour but de contribuer au développement de systèmes intelligents de reconnaissance des sons respiratoires afin de faciliter la procédure de diagnostic des maladies pulmonaires. Le système proposé utilise l'apprentissage profond pour l'analyse des sons respiratoires adventices présentés dans les cycles. Ces sons symptomatiques indiquent souvent des anomalies présentes dans le système respiratoire. Ils sont généralement le signe d'une maladie pulmonaire potentielle.

L'idée derrière l'approche utilisée dans ce système de reconnaissance des sons respiratoires est motivée par le succès remarquable des algorithmes modernes d'apprentissage profond. Ces derniers qui ont démontré une réelle amélioration par rapport aux algorithmes traditionnels d'apprentissage automatique basés sur l'extraction de caractéristiques dans de nombreuses applications et domaines, y compris le domaine biomédical, pour la classification d'images à grande échelle.

L'utilisation d'une telle approche n'a été rendue possible pour classifier les sons respiratoires qu'à partir de 2017. En effet toutes les études liées à l'analyse automatique des sons respiratoires de deux dernières décennies utilisaient de petites bases de données avec seulement quelques enregistrements en raison d'absence d'une grande base de données, ce qui limitait l'utilisation des algorithmes basés sur l'apprentissage profond. Depuis la présentation de la nouvelle base de données de sons respiratoires en 2017, lors de la conférence internationale sur l'informatique de santé biomédicale (ICBHI), de nombreuses approches basées sur l'apprentissage profond ont été considérées.

RÉSUMÉ

L'auscultation des sons respiratoires à l'aide d'un stéthoscope reste essentielle pour l'examen clinique pulmonaire en raison de sa nature non invasive. Bien que son efficacité et sa simplicité fournissent des informations essentielles aux médecins, cette technique demeure limitée par certains inconvénients, notamment par sa subjectivité qui dépendra toujours de la perception auditive des médecins, de leur expérience et de leur capacité à différencier les caractéristiques des différents sons.

Selon l'organisation mondiale de la santé (OMS), les maladies respiratoires représentent la troisième cause de décès dans le monde. La détection de sons adventices tels que des crépitants ou des sibilants pendant l'auscultation est un aspect essentiel de l'examen médical pour diagnostiquer les maladies respiratoires. Dans ce projet de recherche, nous avons développé un système basé sur l'apprentissage profond pour la classification des sons pulmonaires adventices qui peut être intégré dans un outil de diagnostic médical intelligent.

Une méthodologie en trois phases est adoptée pour atteindre l'objectif de cette étude. La première consiste à réaliser une étude comparative pour évaluer l'impact de plusieurs types de représentations temps-fréquence sur les performances de classification de sons respiratoires. Cette étude se base sur l'hypothèse selon laquelle les représentations temps-fréquence sont riches en caractéristiques temporelles et fréquentielles servant à différencier chaque classe de sons respiratoires.

La deuxième phase consiste à trouver une architecture optimale d'un réseau de neurones convolutif (CNN) qui tienne compte du compromis entre les contraintes de mémoire et les performances de classification de sons respiratoires. Pour ces raisons, nous avons développé notre propre modèle. À partir des résultats obtenus, nous avons pu

constater deux points essentiels. Premièrement, un modèle avec plus de couches sera capable d'apprendre plus de caractéristiques abstraites. Cependant, ce modèle sera également plus vulnérable au surapprentissage, car la nature de nos images n'est pas complexe en termes de caractéristiques, mais très sensible en raison de la présence de bruit de fond. Deuxièmement, étant donné le nombre limité d'échantillons de l'apprentissage dans certaines classes de données, le processus de classification devient plus compliqué.

Enfin, la troisième étape explore l'influence de l'ajustement des hyperparamètres de réseau CNN, de l'augmentation des données et des techniques de régularisation des données sur la diminution des erreurs de généralisation afin d'éviter le surapprentissage. Pour comparer les résultats des différentes expériences, nous avons adopté l'usage de la matrice de confusion comme critère spécifique requis par le défi ICBHI 2017 pour évaluer les performances de la classification. Les résultats expérimentaux montrent que notre approche surpasse les méthodes concurrentes dans la tâche de classification de sons respiratoires en quatre classes.

Mots-clés : Apprentissage profond, Réseau CNN, Sons respiratoires, Classification des sons respiratoires, Sibilants, Crépitants, Représentations temps-fréquence, Décomposition en modes empiriques.

ABSTRACT

Auscultation of lung sounds using a stethoscope remains vital for clinical assessment due to its non-invasive nature. Nevertheless, its efficiency and simplicity provide essential information for physicians, yet it still has some drawbacks since this subjective method will always depend on the auditory perception among physicians, experience, and ability to differentiate sound patterns.

According to the world health organization (WHO), respiratory diseases represent the third globally leading cause of death. The detection of abnormal sounds such as crackles or wheezes during auscultation is an essential aspect of the medical examination to diagnose respiratory diseases. In this research project, we have developed a deep learning-based system for abnormal lung sounds classification that may contribute to designing an advanced medical-assistance system to diagnose lung sounds.

A three-phase methodology is adopted to achieve this objective. The first phase consists of a comparative study to evaluate the impact of several types of time-frequency representations on the classification performances of respiratory sounds. This study is based on the hypothesis that the time-frequency representations are rich in both frequency and temporal features serving to differentiate each class of breath sounds.

The second phase involves finding a suitable convolutional neural network (CNN) architecture that considers the trade-off between memory constraints and classification performance. For those reasons, we have developed our proper model. From the obtained results, we were able to identify two key points. Firstly, a model with more layers will be able to learn more abstract features. However, such a model is likely to become vulnerable to overfitting since the nature of our data is not complex in terms of features, but very sensitive due to the presence of background noise. Secondly, given the limited number of

training samples in some data classes, the classification process becomes more complicated.

Finally, the third step explores the influence of CNN network's hyper-parameters, data augmentation, and data regularization techniques on decreasing generalization errors to prevent overfitting. To compare the results of the different experiments, we have adopted the use of the confusion matrix as specific criteria required by the ICBHI 2017 challenge to evaluate the classification performances. The experimental results show that our approach outperforms competing methods in the task of classifying breath sounds into four classes.

Keywords: Deep Learning, CNN network, Respiratory sounds, Classification of respiratory sounds, Wheezes, Crackles, Time Frequency Representations, Empirical Mode Decomposition.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

LISTE DES ABRÉVIATIONS, DES SIGLES ET DES ACRONYMES

2D	Two-Dimensional
ADAM	Adaptive Moment estimation.
ATS	American Thoracic Society
BIGRU	Bidirectional Gated Recurrent Unit
BILSTM	Bidirectional Long Short-Term Memory
BN	Batch Normalization
BPCO	Broncho Pneumopathie Chronique Obstructive
CNN	Convolutional Neural Network
CORSA	Computerized Respiratory Sound Analysis
CQT	Constant-Q Transform
DL	Deep Learning
DNN	Deep Neural Network
EMD	Empirical Mode Decomposition
FFT	Fast Fourier Transform
FM	Frequency Modulation
GMM	Gaussian Mixture Model
GPU	Graphics Processing Unit

GRU	Gated Recurrent Unit
HMM	Hidden Markov Model
IA	Intelligence Artificielle
ICBHI	International Conference on Biomedical and Health Informatics
IF	Instantaneous Frequency
IMF	Intrinsic Mode Functions
KNN	K-Nearest Neighbour
LDA	Linear Discriminant Analysis
LPC	Linear Predictive Coding
LSTM	Long Short-Term Memory
MFCC	Mel-Frequency Cepstral Coefficients
ML	Machine Learning
MPOC	Maladie Pulmonaire Obstructive Chronique
MWSCAS	Midwest Symposium on Circuits and Systems
OMS	Organisation mondiale de la santé
RELU	Rectified Linear Unit
RMSPROP	Root Mean Square Propagation
RNN	Recurrent Neural Network
RSE	Random Subspace Ensembles
RTF	Représentation Temps-Fréquence

SBC	Sub-Band Coding
SE	Sensitivity
SP	Specificity
ST	Stockwell transform
STFT	Short-Time Fourier Transform
SVM	Support Vector Machine
TF	Transformée de Fourier
VGG	Visual Geometry Group
VTLP	Vocal Tract Length Perturbation
WPT	Wavelet Packet Transform
WT	Wavelet Transform
WVD	Wigner-Ville distribution

LISTE DES SYMBOLES

$\Delta \mathbf{f}$	Frequency resolution
$\Delta \mathbf{t}$	Time resolution
$\mathbf{CO_2}$	Carbon dioxide
GB	Gigabyte
GHz	Gigahertz
Hz	Hertz
k	Frequency index
ms	Millisecond
n	Time index
$\mathbf{O_2}$	Oxygen
$\mathcal{L}$	Loss
$\boldsymbol{\eta}$	Learning rate
$\boldsymbol{\lambda}$	Weight loss

INTRODUCTION

Les maladies pulmonaires constituent la troisième cause de mortalité dans le monde selon l'organisation mondiale de la santé (WHO, 2020). Cela inclut des maladies tels que le cancer du poumon, la tuberculose, l'asthme, la maladie pulmonaire obstructive chronique (MPOC), et l'infection des voies respiratoires inférieures. Au Canada (Statistique Canada, 2018), 3.8 millions de personnes vivent avec l'asthme, et 2 millions de personnes vivent avec une maladie pulmonaire obstructive chronique.

Les crépitants et les sibilants sont des sons respiratoires généralement liés à des maladies respiratoires chroniques. Les crépitants sont des sons respiratoires souvent présents dans les cas des maladies pulmonaires obstructives chroniques (MPOC) et de troubles restrictifs, tels que l'insuffisance cardiaque, la fibrose pulmonaire et la pneumonie (Rees et Calverley, 2002). Alors que les sibilants sont des sons respiratoires fréquemment présents dans les cas des maladies pulmonaires chroniques courantes comme l'asthme (Lehrer, 2002). Dans la plupart des cas, le moyen le plus efficace de lutter contre la mortalité de ces maladies est la détection précoce, car elle permet de limiter la propagation et d'accroitre l'efficacité du traitement.

Bien que la méthode conventionnelle d'auscultation avec le stéthoscope fournisse des informations utiles aux médecins, mais elle reste toujours subjective et dépend de la perception auditive des médecins, leurs expériences et de leurs capacités à détecter la présence ou l'absence des sons symptomatiques (Bahoura et Pelletier, 2004). Grâce au progrès technologique, les chercheurs d'aujourd'hui ont la possibilité d'enregistrer les sons respiratoires tout autant que d'exploiter des techniques de traitement des signaux numériques avancées pour mieux les analyser. Le développement d'un outil intelligent d'aide au diagnostic automatique est très envisageable. Il facilitera l'accès aux soins et aux services de santé pour les communautés établies en régions éloignées, où il n'y a pas assez de médecins pour diagnostiquer chaque patient à temps. Par conséquent, l'utilisation d'un stéthoscope électronique doté d'un système de classification basé sur l'apprentissage profond peut aider à surmonter les limites de l'auscultation conventionnelle.

Les algorithmes d'apprentissage automatique basés sur l'extraction de caractéristiques pour la classification d'objets ont été proposés dans la littérature durant les trois dernières décennies. Grâce à ces nombreux algorithmes basés sur l'apprentissage machine (Machine Learning), les systèmes de classification développés ont permis de faire progresser et améliorer les résultats obtenus dans la classification des sons pulmonaires pour de petits bases de données.

L'étude de la littérature des dernières décennies a montré que les algorithmes d'apprentissage profond surpassent les algorithmes d'apprentissage automatique traditionnels basés sur l'extraction de caractéristiques pour la classification d'objets à grande échelle, soit l'utilisation de grandes bases de données (Shuvo *et al*., 2020; Gairola *et al*., 2020; Demir *et al*., 2020a; Hinton *et al*., 2012; Mohamed *et al*., 2009; Qawaqneh *et al*., 2017; Jácome *et al*., 2019; Bardou *et al*., 2018).

Dans ce contexte, ce mémoire présente une approche d'apprentissage profond basée sur l'architecture CNN pour la classification d'anomalies présentées dans les cycles respiratoires. Le premier chapitre introduit de manière générale les signaux respiratoires, et expose la problématique de cette recherche, les objectifs à atteindre, ainsi que la méthodologie. Dans le second chapitre, nous présentons les techniques de représentation temps-fréquence utilisés pour transformer les sons respiratoires en images bidimensionnels. Le troisième chapitre présente les techniques d'apprentissage profond qui permettront, par la suite, l'extraction de caractéristiques et la classification des différentes classes de sons respiratoires. Dans le quatrième chapitre, nous présenteront en détail la méthodologie utilisée et les différentes étapes de cette démarche ainsi que les résultats obtenus. Finalement, la conclusion permet de faire le bilan de ce travail et proposer des perspectives.

CHAPITRE 1
PRINCIPES FONDAMENTAUX DE L'ANALYSE DES SONS RESPIRATOIRES

Dans ce chapitre, nous présentons les définitions et les caractéristiques des différents sons respiratoires. Ensuite, la problématique de cette recherche, les objectifs, les hypothèses et à la fin nous présentons la démarche adoptée pour atteindre ces objectifs.

1.1 NOMENCLATURE DES SONS RESPIRATOIRES

L'auscultation par stéthoscope est la technique la plus répandue et la moins couteuse pour évaluer l'état de santé du système respiratoire d'un patient en écoutant les sons produits par la respiration. Les sons respiratoires se divisent en deux grandes catégories: les sons respiratoires normaux et les sons respiratoires adventices. Si la personne respire naturellement, le flux d'air dans le système respiratoire produit des sons respiratoires normaux. Cependant, si les sons générés par ce système respiratoire produisent des sons bruyants superposés aux sons normaux, on les appelle des sons adventices. Ces sons sont souvent associés à des maladies respiratoires graves, comme l'asthme, la broncho pneumopathie chronique obstructive (BPCO) et la pneumonie (Lehrer, 2002; Pramono *et al*., 2017). L'anatomie du système respiratoire est illustrée à la figure 1.1.

On peut subdiviser les sons respiratoires normaux en sons bronchovésiculaires, trachéobronchiques, vésiculaires et trachéaux selon le site de prise de mesure, au niveau de la trachée ou du torse (Lehrer, 2002; Bahoura, 2009). Les sons adventices se divisent en sons respiratoires continus et discontinus en fonction de la durée de leur apparition. Les sons adventices continus peuvent être subdivisés en respiration sibilante aigüe et en respiration sibilante grave en fonction du nombre de vibrations par unité de temps, en Hertz. Un grand nombre de vibrations donne une respiration sibilante aigüe et vice versa.

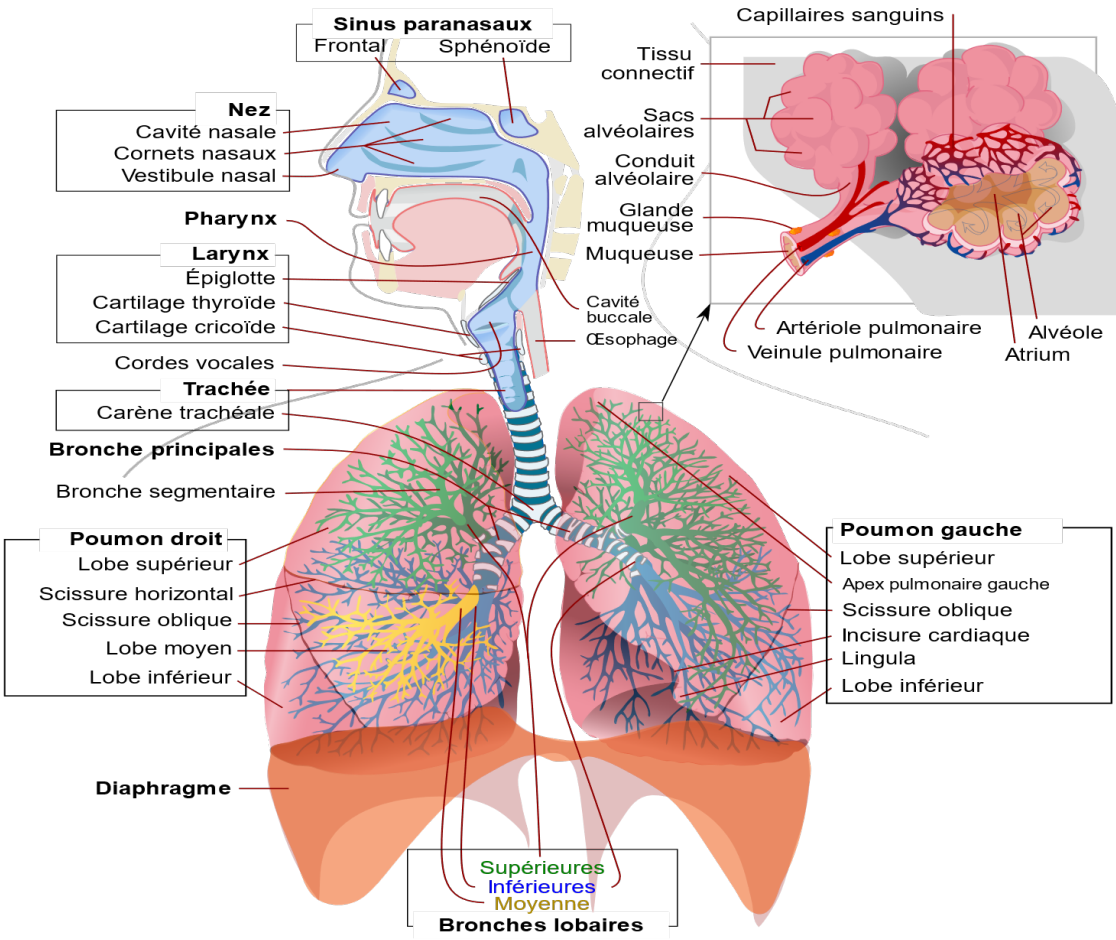


Figure 1.1. L'anatomie du système respiratoire (Netter, 2018).

Selon la définition antérieure de l'ATS (American Thoracic Society), les sons respiratoires adventices sont considérés comme continus si leur durée est supérieure à 250 ms, sinon, ils sont considérés comme discontinus (Bahoura, 2009; Sovijärvi *et al*., 2000; Reichert *et al*., 2008).

Les sibilants sont des sons respiratoires adventices continus de nature musicale (Sovijärvi *et al*., 2000). L'ATS définit les sibilants comme des sons respiratoires continus aigus avec une fréquence dominante supérieure à 400 Hz, et les rhonchus comme des sons continus graves avec fréquence dominante de 200 Hz ou moins.

Selon les nouvelles directives de CORSA (Computerized Respiratory Sound Analysis), la fréquence dominante des sibilants est généralement supérieure à 100 Hz et sa durée dépasse 100 ms (Sovijärvi *et al*., 2000). De nombreuses circonstances peuvent être à l'origine d'une respiration sibilante. Elles comprennent tous les mécanismes qui restreignent les voies aériennes, telles que le bronchospasme, l'œdème muqueux, la compression externe par une masse tumorale ou l'obstruction dynamique des voies aériennes (Lehrer, 2002). Ainsi, les sibilants peuvent être entendus dans le cas de plusieurs maladies, et non seulement dans le cas de l'asthme (Moussavi, 2006). En ce qui concerne les applications cliniques de l'analyse des sibilants, des études antérieures ont montré que le volume ou l'intensité des sibilants, le caractère monophonique ou polyphonique, le moment de l'inspiration ou de l'expiration n'étaient pas corrélés à l'obstruction des voies respiratoires (Loudon, 1993).

En revanche, les sons respiratoires adventices discontinus ont un caractère explosif répétitif, avec une durée d'apparition inférieure à 20 ms (Reichert *et al*., 2008). Nous pouvons les classer en deux catégories : gros crépitants et crépitants fins. Les gros crépitants sont caractérisés par une fréquence faible de 350 Hz, et une durée d'apparition de 15 ms. Ces sons sont surtout audibles au début de l'inspiration, mais peuvent aussi être perçus à l'expiration. Ils peuvent être entendus chez les patients atteints de bronchite chronique, bronchiectasie et de broncho-pneumopathie chronique obstructive (BPCO), mais sont généralement associés à une obstruction grave des voies respiratoires (Rees et Calverley, 2002). Les crépitants fins sont provoqués par l'ouverture soudaine des petites voies respiratoires. Ces sons adventices sont classés comme aigus quand leur plage de fréquences est dans les 650 Hz, et leur durée ne dépasse pas les 5 ms. Les crépitants fins ne sont audibles qu'à la fin des phases inspiratoires, et sont généralement causés à une pneumonie, une insuffisance cardiaque congestive et une fibrose pulmonaire (Pramono *et al*., 2017). La comparaison et le résumé des types et des caractéristiques des sons respiratoires sont présentés dans les tableaux 1.1 et 1.2.

Tableau 1.1. Types de sons respiratoires et leurs caractéristiques.

Sons respiratoires						
Normaux			**Adventices**			
Vésiculaires	Bronchiques	Trachéaux	Continus > 250 ms		Discontinus < 250 ms	
Plage de fréquences						
100-1000 Hz	100-5000 Hz	100-5000 Hz	Sibilant > 400 Hz	Rhonchus < 200 Hz	Gros crépitant 350 Hz	Crépitant fin 650 Hz

Les figures 1.2-1.4 suivantes présentent des cas de sons respiratoires, normaux et adventices (sibilants et crépitants), avec leurs phonopneumographies temporelles et spectrales. Les phonopneumographies spectrales présentées permettent de déterminer la plage de fréquences pour chaque type de sons concernés. La figure 1.2 (B) représente un spectre de fréquences avec une largeur de bande supérieure à 10% du maximum du signal qui s'étendant jusqu'à 400 Hz dans une respiration normale, 590 Hz dans une respiration crépitante (voir figure 1.3 (B)) et 712 Hz dans une respiration sibilante (figure 1.4 (B)).

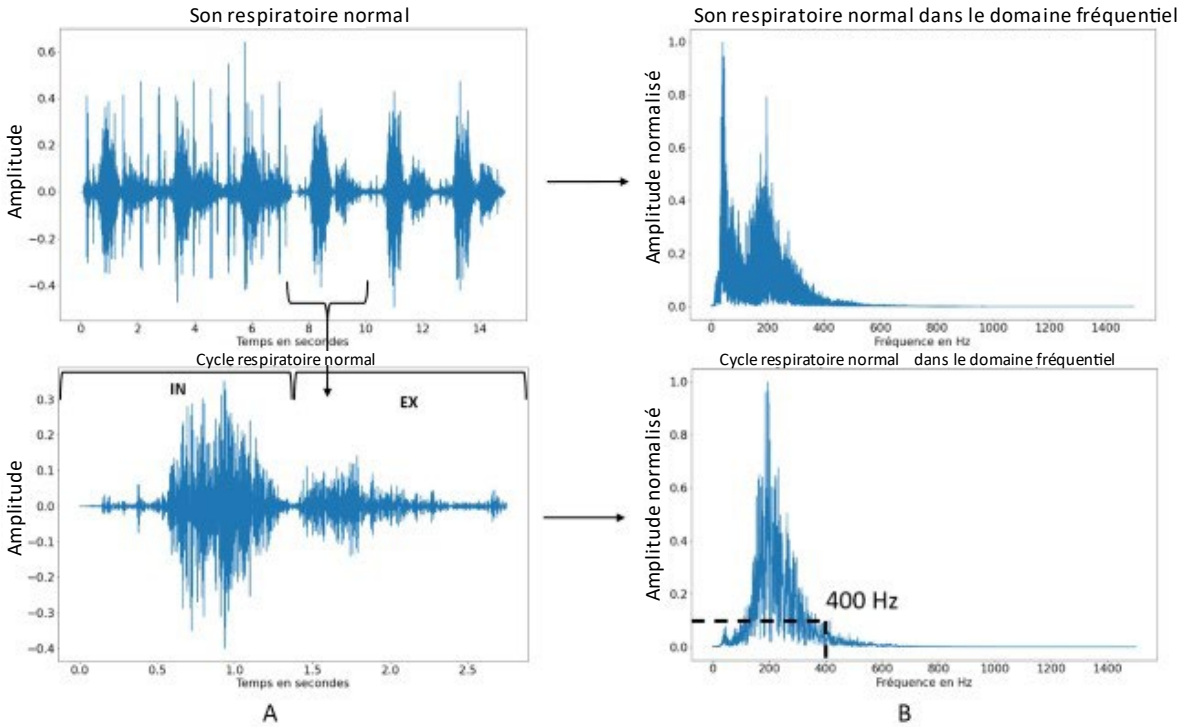


Figure 1.2. Phonopneumographie temporelle (A) et spectrale (B) d'un son normal.

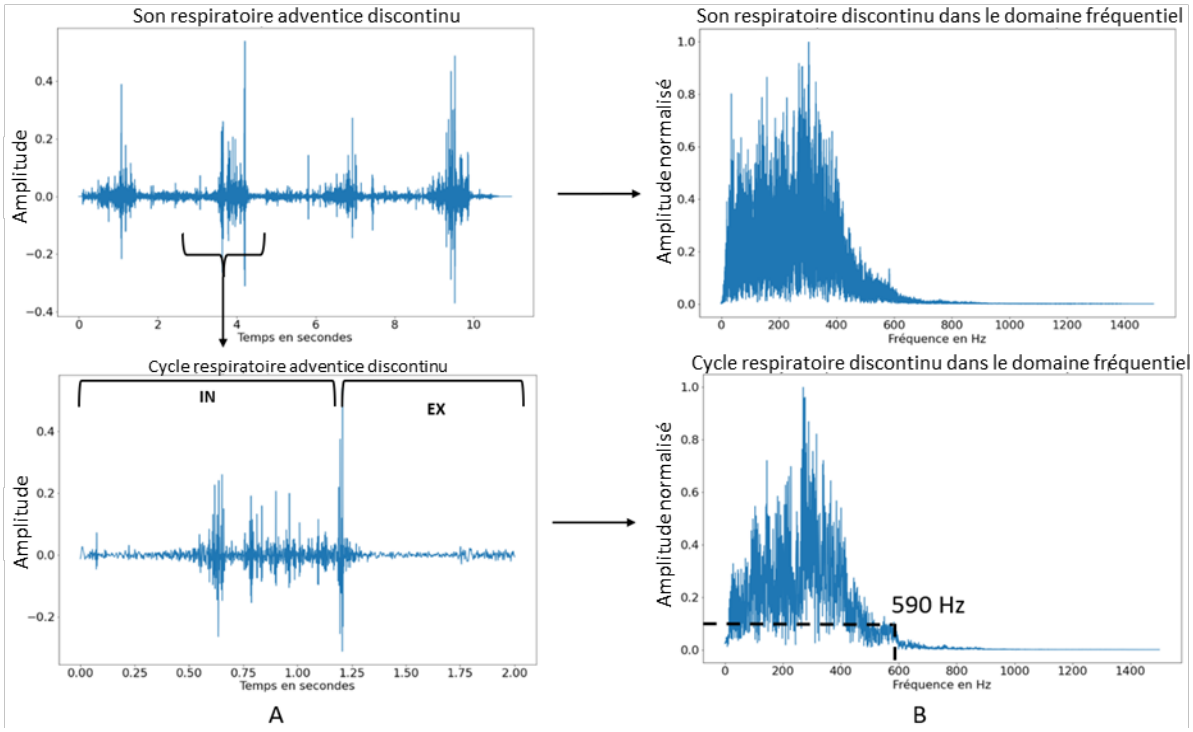


Figure 1.3. Phonopneumographie temporelle (A) et spectrale (B) d'un crépitant.

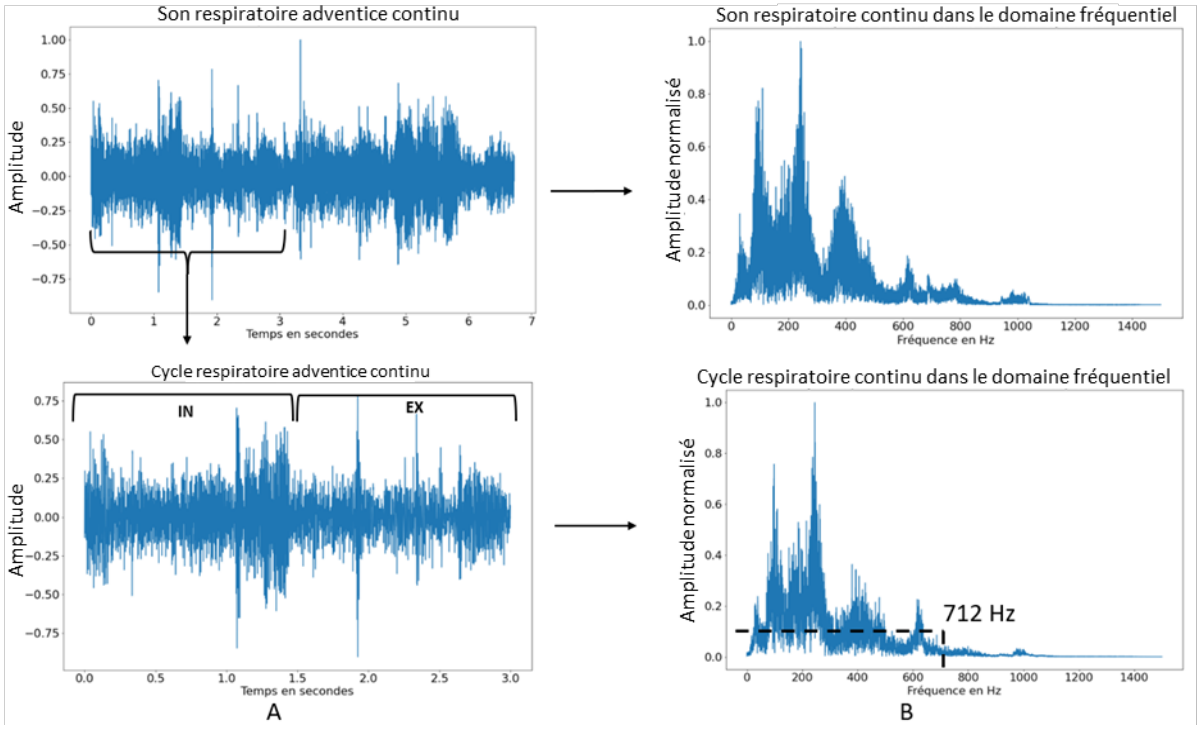


Figure 1.4. Phonopneumographie temporelle (A) et spectrale (B) d'un sibilant.

Tableau 1.2. Définition et appellation des sons adventices selon l'American Thoracic Society (Tableau provenant de (Mikami *et al.*, 1987))

	G.B. et U.S.A.	Allemagne	France	Proposition de modification	Forme d'onde en temps étendu
Discontinous (<250 ms)					
-Fine (high pitched, low amplitude, short duration, 1DW*=0.92 ms, 2CD¶ = (6.02 ms)	Fine crackles	Feines Rasseln	Râles crépitants	Crépitements fins	
-Coarse (high pitched, low amplitude, long duration, 1DW= 1.25 ms, 2CD = (9.32 ms)	Coarse crackles	Grobes Rasseln	Râles bulleux ou sous-crépitants	Gros crépitements	
Continous (>250 ms)					
-High pitched (dominant frequency > 400 Hz)	Wheezes	Pfeifen	Râles sibilants	Sifflements	
-Low pitched (dominant frequency < = 200 Hz)	Ronchi	Brummen	Râles ronflants	Ronchus	

* Initial Deflection Width (largeur de déflection initiale)

¶ Two Cycle Duration (durée des 2 premiers cycles)

1.2 BASE DE DONNEES UTILISÉE

Dans le cadre de cette étude, nous utilisons la base de données du défi scientifique ICBHI 2017 (Rocha *et al.*, 2019) qui est actuellement considérée comme la plus grande base de données de sons respiratoires disponible publiquement. La figure 1.5 présente la répartition des cycles par classe de sons respiratoires dans cette base de données. Elle contient 920 fichiers audio annotés, provenant de 126 participants et collectés indépendamment par deux équipes de recherche de deux pays différents. La création de cette base de données a nécessité de nombreuses années de travail, ce qui a permis d'obtenir une durée totale de 5 heures et 30 minutes. Elle représente donc une référence dans ce domaine.

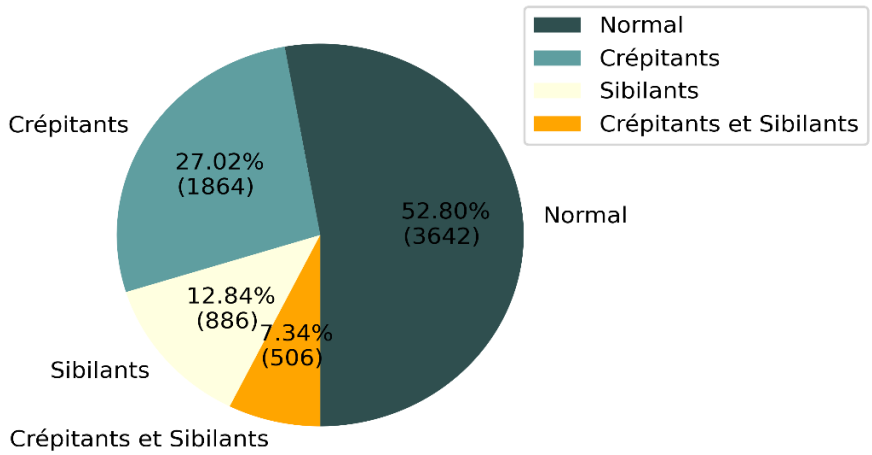
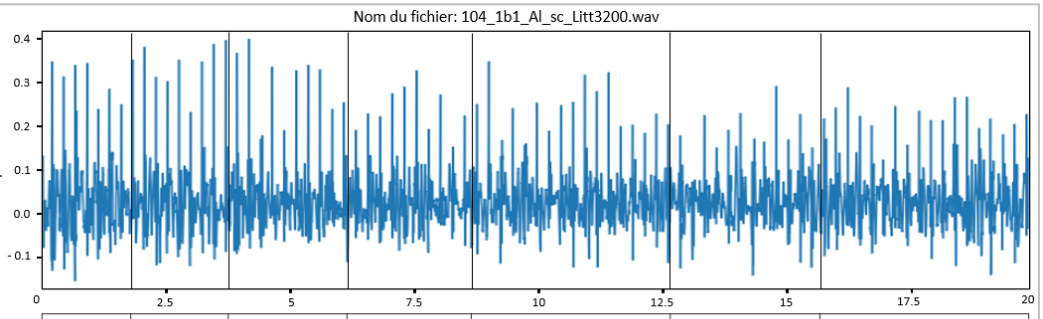


Figure 1.5. Distribution des cycles par classe dans la base de données ICBHI.

L'ICBHI 2017 est une base de données riche et en même temps complexe. On trouve dans chaque fichier audio différents types de cycles appelés crépitants, sibilants, crépitants et sibilants, et normaux accompagnés d'étiquettes temporelles de début et de fin. Ces cycles présentent des durées d'enregistrement différentes, allant de 0.2 s à 16.2 s, et le nombre de cycles dans chaque type de sons respiratoires est déséquilibré, avec respectivement 1864, 886, 506 et 3642 cycles (voir figure 1.5). Les fichiers audio ont des durées variables de 10 à 90 secondes et utilisent une variété de fréquences d'échantillonnage allant de 4 kHz à 44.1 kHz (Rocha *et al.*, 2019). Cette base de données comprend des fichiers au format

(.WAV) et les fichiers d'annotations correspondants. Comme indiqué dans la figure 1.5, chaque fichier d'annotation comprend quatre colonnes, indiquant le début et la fin de chaque cycle, suivis de deux colonnes indiquant la présence ou l'absence de sibilant et/ou de crépitant (voir figure 1.6). La figure 1.7 représente la distribution de la longueur des cycles dans toute la base de données.



	1	2	3	4	5	6	7
Début du cycle (s)	0	1.8771	3.7543	6.1071	8.2502	12.618	15.858
Fin du cycle (s)	1.8771	3.7543	6.1071	8.2502	12.618	15.858	20
Crépitant	0	0	1	0	0	1	0
Sibilant	0	0	0	1	0	1	0
Nature du cycle	Normal	Normal	Crépitant	Sibilant	Normal	Cr. & Sib.	Normal

Figure 1.6. Exemple de fichiers d'annotation et son utilisation dans la segmentation des fichiers audio correspondants.

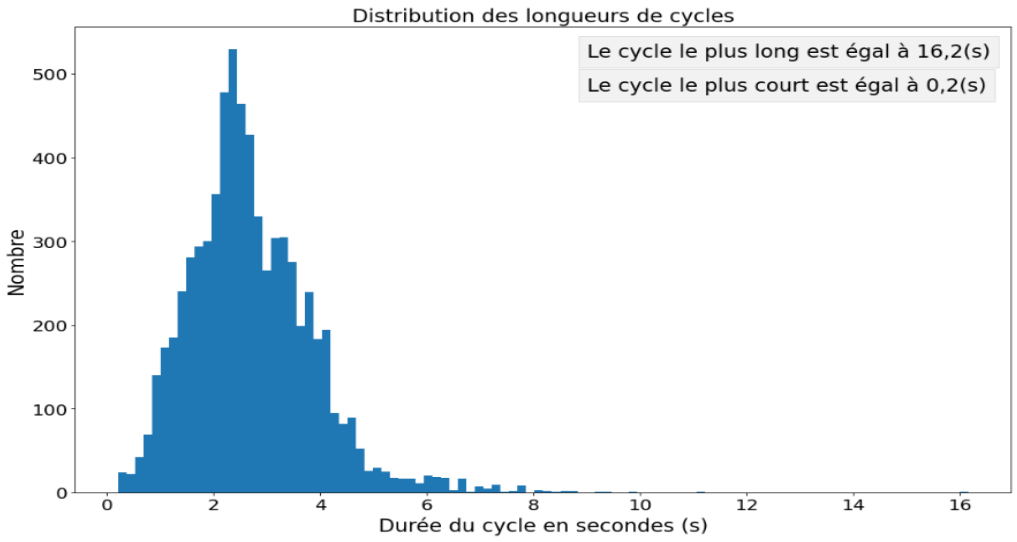


Figure 1.7. Distribution de la longueur des cycles à travers les enregistrements.

1.3 PROBLÉMATIQUE

L'auscultation des sons respiratoires à l'aide d'un stéthoscope demeure la méthode la plus utilisée par les médecins pour déterminer la présence d'une maladie au niveau des poumons, non seulement parce qu'il s'agit d'une technique non invasive et moins couteuse, mais aussi parce que son efficacité et sa simplicité sont connues pour permettre à un médecin praticien d'interpréter la présence de signes pathologiques (Moussavi, 2006). Si on considère les pays à faible qualité de vie, où il n'y a tout simplement pas assez de médecins pour diagnostiquer chaque patient à temps, le développement d'un système automatique de classification de sons respiratoires peut aider à surmonter ces limitations.

Bien que la méthode conventionnelle d'auscultation à l'aide du stéthoscope fournisse des informations utiles aux médecins, elle présente des limites de subjectivité liées à la perception auditive des médecins, leur expérience et leur capacité à différencier et évaluer les sons respiratoires (Bahoura, 2009). D'où le besoin d'une approche quantitative objective de reconnaissance de sons respiratoires.

Au cours des trois dernières décennies, plusieurs méthodes d'apprentissage machine ont été proposées, basées sur l'extraction de caractéristiques par transformée de Fourier (FFT) (Abbas et Fahim, 2010), transformée d'ondelettes (WT) (Pesu *et al*., 1998) ou par extraction de coefficients cepstraux à l'échelle de Mel (MFCC) (Bahoura et Pelletier, 2004). Tandis que les réseaux de neurones multicouches (MLP) (Forkheim *et al*., 1995), machine à vecteurs de support (SVM) (Boujelben et Bahoura, 2018), modèle de mélange gaussien (GMM) (Bahoura et Pelletier, 2004) et méthode des k plus proches voisins (KNN) (Palaniappan *et al*., 2014) ont été principalement utilisés pour la classification des sons respiratoires. Cependant, la revue de la littérature récente sur les systèmes automatiques d'analyse de sons respiratoires montre l'utilisation de l'apprentissage profond (Liu *et al*., 2019; García *et al*., 2020; Perna, 2018; Aykanat *et al*., 2017; Shuvo *et al*., 2020; Liu *et al*., 2019; Rocha *et al*., 2020; Kim *et al*., 2021).

La problématique de recherche consiste à mettre en place un système capable d'identifier et de classer les sons symptomatiques qui servent généralement à détecter une éventuelle maladie pulmonaire. Cet outil va permettre de poser un bon diagnostic, de suivre l'évolution de la maladie et d'évaluer l'efficacité des traitements prescrits.

La complexité de ce type de signaux, le manque de grandes bases de données accessibles et l'absence d'une méthode standard d'évaluation entre les travaux des équipes de recherche constituent un obstacle considérable dans la recherche sur les signaux respiratoires (Pelletier, 2006).

1.4 OBJECTIFS

L'objectif de cette étude est de développer un système intelligent permettant de distinguer entre les quatre classes de sons respiratoires. Nous proposons une architecture basée sur les réseaux de neurones convolutifs (CNN), notre approche pour la classification utilise des techniques de représentation temps-fréquence pour créer des images à partir de sons respiratoires enregistrés. Ce choix est motivé par la disponibilité d'une base de donnes relativement large.

1.5 HYPOTHESES

Pour atteindre les objectifs de ce projet de recherche, nous faisons appel aux techniques de traitement de signaux liées à la problématique de la reconnaissance de formes. Les sons respiratoires sont des sons acoustiques qui partagent quelques propriétés avec les sons de parole. Cette ressemblance (source, production, conduction, etc.) permet de supposer que les techniques qui performent en reconnaissance de la parole fonctionneraient également pour les sons acoustiques respiratoires.

La disponibilité d'une base de données relativement large comme celle utilisée dans ce projet de recherche (ICBHI), donne la possibilité d'exploiter les algorithmes de classification les plus avancés et de les comparer objectivement aux systèmes existants.

1.6 MÉTHODOLOGIE

La figure 1.8 présente une vue d'ensemble de notre approche concernant cette étude. Nous décrivons les étapes de prétraitement, du rééchantillonnage jusqu'au découpage et au fenêtrage, suivies par la transformation temps-fréquence et la création de spectrogrammes qui permettent de décrire le contenu fréquentiel des sons respiratoires utilisés. Pour résoudre les problèmes de déséquilibre de classe et de minimums locaux, nous incorporons plusieurs techniques de manipulation des données.

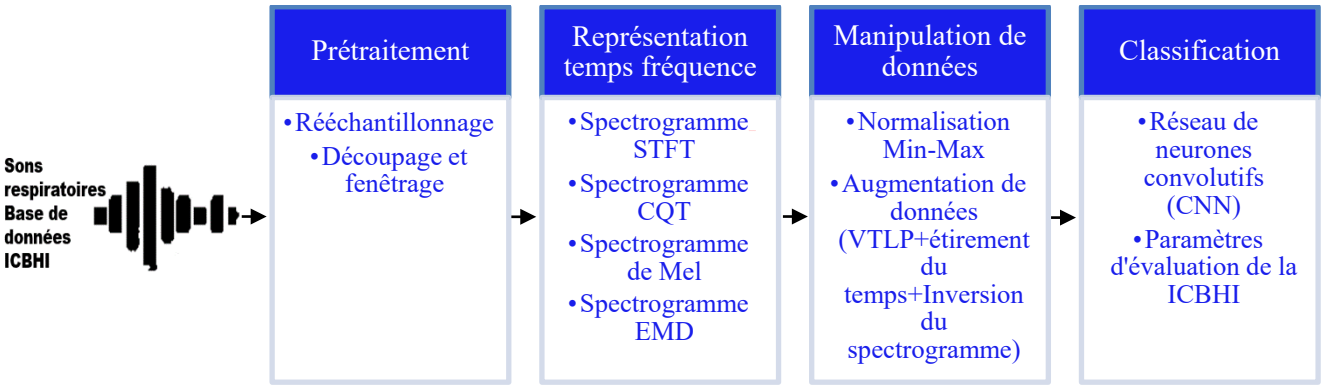


Figure 1.8. Descriptif de l'approche proposée. L'étape de rééchantillonnage et de segmentation des sons respiratoires est suivie par la création des représentations temps-fréquence (RTF). Ensuite l'augmentation et la normalisation de données seront utilisées avant de passer les images créées par les RTF au réseau de neurones convolutif (CNN).

Notre approche pour mettre en place un tel système de classification des sons respiratoires se résume en quatre étapes :

1.5.1 Prétraitement de données

La première étape consiste à amener les différentes fréquences d'échantillonnages utilisées dans les différents fichiers audio à une fréquence d'échantillonnage unique, suivie d'une séquence de découpage et de segmentation. Dans la phase de découpage, chaque fichier audio sera divisé en plusieurs cycles en tenant compte des fichiers d'annotation correspondants. Étant donné la durée variable de chaque cycle (voir figure 1.7), l'étape de segmentation est inévitable tant que les CNN ne prennent que des images d'entrée de taille unique. L'étape de segmentation consiste donc à unifier la longueur des cycles par

troncatures ou par remplissage. Ce dernier est réalisé par ajout de zéros ou par réflexion. Il faut note que la longueur de chaque cycle est bien définie dans les fichiers d'annotation tandis que la longueur des segments est un paramètre qui sera défini dans le dernier chapitre.

1.5.2 Représentation temps-fréquence

Cette étape consiste à générer des images basées sur la technique de représentation temps-fréquence (RTF) pour visualiser à la fois le contenu temporel et fréquentiel des sons respiratoires. Trois représentations différentes ont été utilisées : la transformée de Fourier à court terme (STFT), la transformée à l'échelle de Mel et la transformée à Q constant (CQT). De plus, nous avons aussi proposé d'utiliser ces trois techniques en combinaison avec la technique de décomposition en modes empiriques (EMD). Avec 6 IMFs (fonctions de mode intrinsèque), cela amène à un total de 18 RTFs à comparer. Toutes les images obtenues avec les différentes méthodes ont été transformées en niveaux de gris.

1.5.3 Manipulation de données

Cette étape vient juste après la génération des spectrogrammes, elle se compose de deux étapes principales :

1.5.2.1 Normalisation de données

Le redimensionnement des entrées et des sorties utilisées dans le processus d'apprentissage est un facteur important lorsqu'on travaille dans le domaine d'apprentissage machine. Les variables d'entrée non mises à l'échelle peuvent entrainer un processus d'apprentissage lent ou instable. Il est souvent possible d'améliorer les performances d'un modèle en transformant d'abord les données, notamment par la normalisation et/ou la standardisation. (Géron, 2019). Plusieurs auteurs proposent l'utilisation de cette technique d'une façon directe (Kochetov *et al*., 2018; Perna et Tagarelli, 2019) ou indirecte (Fraiwan *et al*., 2021).

1.5.2.2 Augmentation de données

Dans le domaine de l'apprentissage profond, la performance d'un modèle s'améliore souvent avec la quantité de données disponibles pour son entrainement. L'augmentation des données peut également aider à prévenir le phénomène de surentrainement (overfitting), qui se produit généralement lors de l'entrainement d'un grand modèle sur un petit ensemble d'images (Géron, 2019; Shorten et Khoshgoftaar, 2019; Mushtaq et Su, 2020; Kochetov *et al*., 2018). Malheureusement, de nombreux domaines d'application tels que le domaine médical n'a pas accès aux données massives (Big Data). De plus, ils souffrent d'un déséquilibre de classes dans les bases de données disponibles, ce qui est le cas avec la base de données ICBHI (Gairola *et al*., 2020; Nguyen et Pernkopf, 2020). Dans cette étude, nous avons utilisé trois techniques d'augmentation de données qui nous permettent de générer des images afin d'augmenter la quantité de données en ajoutant des versions légèrement modifiées des images déjà existantes ou des images synthétiques nouvellement créées à partir de données existantes. Cette méthode sera appliquée directement aux représentations temps-fréquence résultantes.

1.5.4 Classification par réseau de neurones convolutif (CNN)

Dans la dernière étape, nous utiliserons les réseaux CNN qui permettent à la fois d'extraire automatiquement les caractéristiques des images créées et de les classer en 4 catégories, incluant les sibilants, crépitants, normaux, et crépitants et sibilants ensemble de l'autre. Les réseaux de neurones convolutifs (CNN) sont très similaires aux réseaux de neurones standards, où chaque neurone reçoit une entrée et exécute ensuite une opération pour avoir une sortie. La principale distinction entre les deux architectures est que les CNN sont un type de réseau spécialisé dans le traitement des données ayant une topologie en forme de grille, comme les images. L'architecture des CNN est composée de trois types de couches : les couches de convolutions, les couches de pooling et les couches entièrement connectées.

Afin d'atteindre cet objectif, nous chercherons dans la littérature les meilleures architectures CNN proposées dans des domaines connexes, en vue de les utiliser dans la classification des sons respiratoires.

1.7 CONTRIBUTIONS

Ce mémoire présente le travail de recherche réalisé dans le cadre de la maitrise en ingénierie. Les résultats obtenus ont mené à la rédaction de deux publications scientifiques dont un accepté et publié et un autre article en phase de finalisation pour une soumission à un journal de référence dans les prochaines semaines.

- La première contribution intitulée "Convolutional Neural Network based Model for Lung Sounds Classification". C'est un article qui a été accepté le 24 avril 2021, présenté le 10 Aout 2021 à la conférence "64th IEEE International Midwest Symposium on Circuits and Systems (MWSCAS) 2021" et publié le 13 September 2021.

Chanane, H., & Bahoura, M. (2021). Convolutional Neural Network-based Model for Lung Sounds Classification. Midwest Symposium on Circuits and Systems, 555–558. https://doi.org/10.1109/MWSCAS47672.2021.9531887.

- La deuxième contribution s'intitule "Lung Sounds Classification System based on Time-Frequency Representation and Convolutional Neural Network". Cet article est une version étendue de la première contribution, il sera soumis prochainement à un journal de référence dans le domaine.

CHAPITRE 2
REPRÉSENTATION TEMPS FRÉQUENCE DES SONS RESPIRATOIRES

Ce chapitre traite de l'étape de la représentation temps-fréquence des signaux. Il comprend les techniques rencontrées dans la littérature et celles que nous avons proposées dans ce projet.

2.1 PRINCIPE DE LA TRANSFORMATION TEMPS FREQUENCE

Un inconvénient du phonopneumogramme temporel et spectral présenté dans les figures 1.2, 1.3 et 1.4; est que les sons normaux ne peuvent pas être facilement distingués des sons adventices, tels que les crépitants et les sibilants par une simple analyse temporelle ou fréquentielle. Compte tenu du fait que les méthodes d'analyses conventionnelles, comme la transformée de Fourier (TF), qui est inadéquate pour l'analyse des signaux non stationnaires comme les sons respiratoires, une solution évidente consiste à adopter une représentation du signal dans le domaine bidimensionnel temps-fréquence.

L'analyse temps-fréquence permet de représenter simultanément des informations importantes relatives aux caractéristiques temporelles et fréquentielles. Étant donné la nature non stationnaire des signaux rencontrés dans le monde réel, il est nécessaire d'utiliser la représentation temps-fréquence pour analyser leurs caractéristiques fréquentielles variables dans le temps (Bahoura, 2019). On dispose dans le domaine du traitement de signal de différentes méthodes pour représenter un signal en temps-fréquence comme la transformée de Fourier à court-terme (STFT), la transformée en ondelettes (WT), la transformée par paquets d'ondelettes (WPT), la distribution de Wigner-Ville (WVD) et la transformée en S (ST) (Bahoura, 2019; Boashash, 2015). Chaque méthode a ses avantages et ses inconvénients.

Malgré de nombreuses améliorations des outils de représentation temps-fréquence (RTF) de haute résolution, les spectrogrammes restent un outil important pour représenter le contenu temporel et fréquentiel des signaux dans plusieurs systèmes et applications. La simplicité, l'efficacité, la robustesse et les bonnes performances de la transformée de Fourier à court-terme (STFT) en font l'outil principal pour l'analyse temps-fréquence.

Le choix de l'outil de représentation du signal dépend du degré de complexité de ce dernier. En réalité, pour toute application, on ne dispose pas d'une solution unique. Au contraire, chaque représentation temps-fréquence (RTF) pourrait être adaptée à une classe particulière de signaux en fonction de leurs types et caractéristiques. Dans cette étude, nous présentons et comparons différentes méthodes de RTF pour l'analyse des sons respiratoires. Il s'agit de spectrogramme à base de transformée de Fourier (STFT), le spectrogramme à base de transformée CQT et le spectrogramme à base de la transformée de Fourier à l'échelle de Mel (Figure 2.1)

2.2 TRANSFORMÉE DE FOURIER À COURT TERME (SPECTROGRAMME STFT)

La transformée de Fourier à court terme (STFT) est utilisée pour construire des représentations qui capturent à la fois le contenu temporel et fréquentiel local du signal. Pour chaque instant, la transformée de Fourier est calculée sur un intervalle limité, d'où vient le terme court terme. En utilisant une fonction de fenêtre glissante dans le temps $w(t)$, cette méthode propose une extension de la transformée de Fourier afin de traiter la non-stationnarité du signal en appliquant des fenêtres pour une analyse segmentée. Le but de l'utilisation de la fenêtre est d'avoir des caractéristiques fréquentielles stationnaires sur le signal par rapport à la transformée de Fourier (Fulop, 2011).

La formulation mathématique de la transformée de Fourier à court terme (STFT) d'un signal discret x(n), où n est l'indice temporel total discret variant de zéro au nombre total d'échantillons, est donnée par l'équation 2.1. Ce signal va être divisé en trames superposées qui sont multipliées par une fenêtre de pondération discrète $w(n)$ afin d'être traitées séparément (Bahoura, 2019).

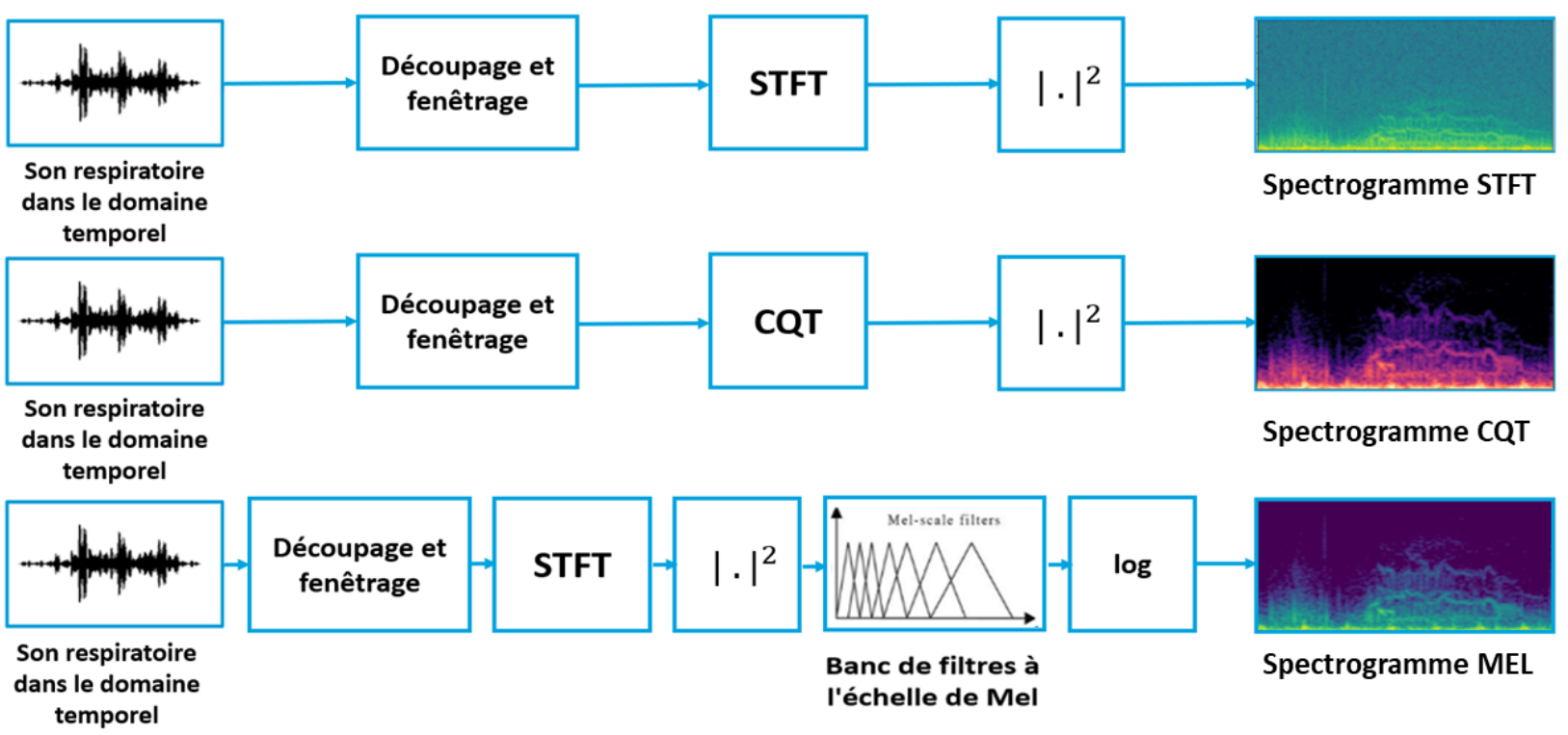


Figure 2.1. Diagramme à blocs du processus de construction des RTF utilisant respectivement la STFT, la CQT et le spectrogramme Mel.

$$X(m,k) = \sum_{n=0}^{N-1} x(n)w(n-mL)e^{-2j\pi kn/N} \quad (2.1)$$

où n et k sont respectivement les indices du temps et de fréquence, L correspond au pas de déplacement entre deux fenêtres successives et m est l'indice de la trame. Ainsi, nous définissons le taux de chevauchement entre deux trames successives selon l'équation 2.2 (Bahoura, 2019) :

$$Taux\ de\ chevauchement\ (\%) = \frac{N-L}{N} \times 100 \quad (2.2)$$

La fenêtres de pondération $w(n)$, d'une longueur de N échantillons dans l'équation 2.3, est supposée être non nulle uniquement dans l'intervalle $[0, N-1]$. La fenêtre de pondération $w(n)$ sert notamment à diminuer les effets de bords causés par les discontinuités d'amplitude aux limites. Plusieurs fenêtres de pondération sont proposées dans la littérature, mais on utilise souvent les fenêtres de Hamming, Blackman ou de Hanning.

Pour une fréquence d'échantillonnage F_e, les résolutions temporelle et fréquentielle d'un spectrogramme basé sur des fenêtres de pondération non chevauchantes, sont respectivement données par les équations 2.3 et 2.4 :

$$\Delta t = N/\ F_e \quad (2.3)$$

$$\Delta f = F_e/N \quad (2.4)$$

Dans le cas d'un spectrogramme basé sur des fenêtres de pondération chevauchantes. La résolution temporelle et fréquentielle sont données dans les équations 2.5 et 2.6 :

$$\Delta t = L/\ F_e \quad (2.5)$$

$$\Delta f = F_e/N \quad (2.6)$$

Il existe un grand compromis entre la résolution temporelle et la résolution fréquentielle dans la STFT, ce qui nécessite de bien choisir la largeur de la fenêtre utilisée.

Par conséquent, si nous utilisons une fenêtre de largeur étroite, nous serons en mesure d'obtenir une meilleure résolution temporelle au détriment de la résolution fréquentielle, et vice versa. C'est le principe d'incertitude d'Heisenberg (Fulop, 2011; Kehtarnavaz, 2008; Boashash, 2015).

Le calcul de la transformée STFT sur plusieurs segments donne une matrice de nombres complexes $X(m,k)$ appelée matrice STFT. Cependant, comme le DFT est une fonction complexe, il est difficile de visualiser ces arguments en tant que tels. Une première solution serait de tracer le spectre de magnitude ou le spectre de puissance. La représentation graphique du spectre de puissance en fonction du temps constitue le spectrogramme, qui peut être calculé par l'équation 2.7 :

$$S(m,k) = \frac{1}{N}|X(m,k)|^2 \tag{2.7}$$

où $X(n,k)$ désigne la matrice STFT du son respiratoire, alors que $S(n,k)$ représente la densité spectrale de puissance à deux dimensions (temps et fréquence). Le spectrogramme décrit la concentration d'énergie en fréquence relative au son respiratoire en fonction du temps; par conséquent, il reflète les propriétés de la forme d'onde du son respiratoire qui varient dans le temps. Plus l'énergie dans le spectre est importante à une fréquence spécifique, plus le dégrée de luminosité est important dans le spectrogramme (Loizou, 2007).

La figure 2.2 montre les spectrogrammes des sons respiratoires de quatre classes provenant de la base de données ICBHI. Dans la figure 2.2(A), nous pouvons voir que la présence d'un sibilant se traduit par une rayure dans le spectrogramme correspondant à un pic assez fort ayant une certaine durée, la présence d'un crépitant se traduit par un pic assez fort dans le spectrogramme ayant une très courte durée (figure 2.2(B)), alors que la présence des deux se traduit par ces deux caractéristiques ensemble (figure 2.2(C)).

Les caractéristiques, illustrées à la figure 2.2, seront par la suite utilisées par le réseau de neurones convolutif (CNN) pour son apprentissage. Une fois l'entrainement est fini, des phases de validation et de test seront menées pour évaluer les performances de ce réseau.

2.3 TRANSFORMÉE À Q CONSTANT (SPECTROGRAMME CQT)

Une autre technique d'analyse temps-fréquence qui transforme un signal $x(n)$ du domaine temporel vers le domaine temps-fréquence est la transformée à facteur de qualité Q constant ou plus couramment appelée transformée à Q constant (CQT). Dans de nombreux cas comme celui des signaux musicaux, la CQT fournit une meilleure représentation des données spectrales que la transformée de Fourier (FFT), en plus d'être plus efficace sur le plan des calculs (Brown et Puckette, 1992). La transformée CQT, dénotée par $X^{CQ}(n,k)$, du signal discret $x(n)$ peut être exprimée par l'équation 2.8 suivante:

$$X^{CQ}(m,k) = \sum_{i=m-\left\lfloor \frac{N_k}{2} \right\rfloor}^{i=m+\left\lfloor \frac{N_k}{2} \right\rfloor} x(i) a_k^* \left(i - m + \frac{N_k}{2} \right) \tag{2.8}$$

où $\lfloor . \rfloor$ signifie l'arrondi, $*$ désigne le conjugué, N_k représente la taille variable des fenêtres, n étant l'indice du temps et k comme indice de fréquence de la CQT. Les fonctions de base $a_k(n)$ sont des formes d'onde à valeurs complexes, également appelées atomes temps-fréquence, définies par l'équation 2.9 suivante:

$$a_k(n) = \frac{1}{N_k} w\left(\frac{n}{N_k}\right) e^{-j2\pi m \frac{f_k}{f_s}} \tag{2.9}$$

où $w(n,k)$ représente la fenêtre de pondération (Hanning, Hamming ou Blackman), f_s la fréquence d'échantillonnage et f_k la fréquence de la k^{ieme} composante spectrale. La CQT définit la taille de la fenêtre N_k selon l'équation 2.10:

$$N_k = \left\lceil \frac{f_s}{f_k} Q \right\rceil \tag{2.10}$$

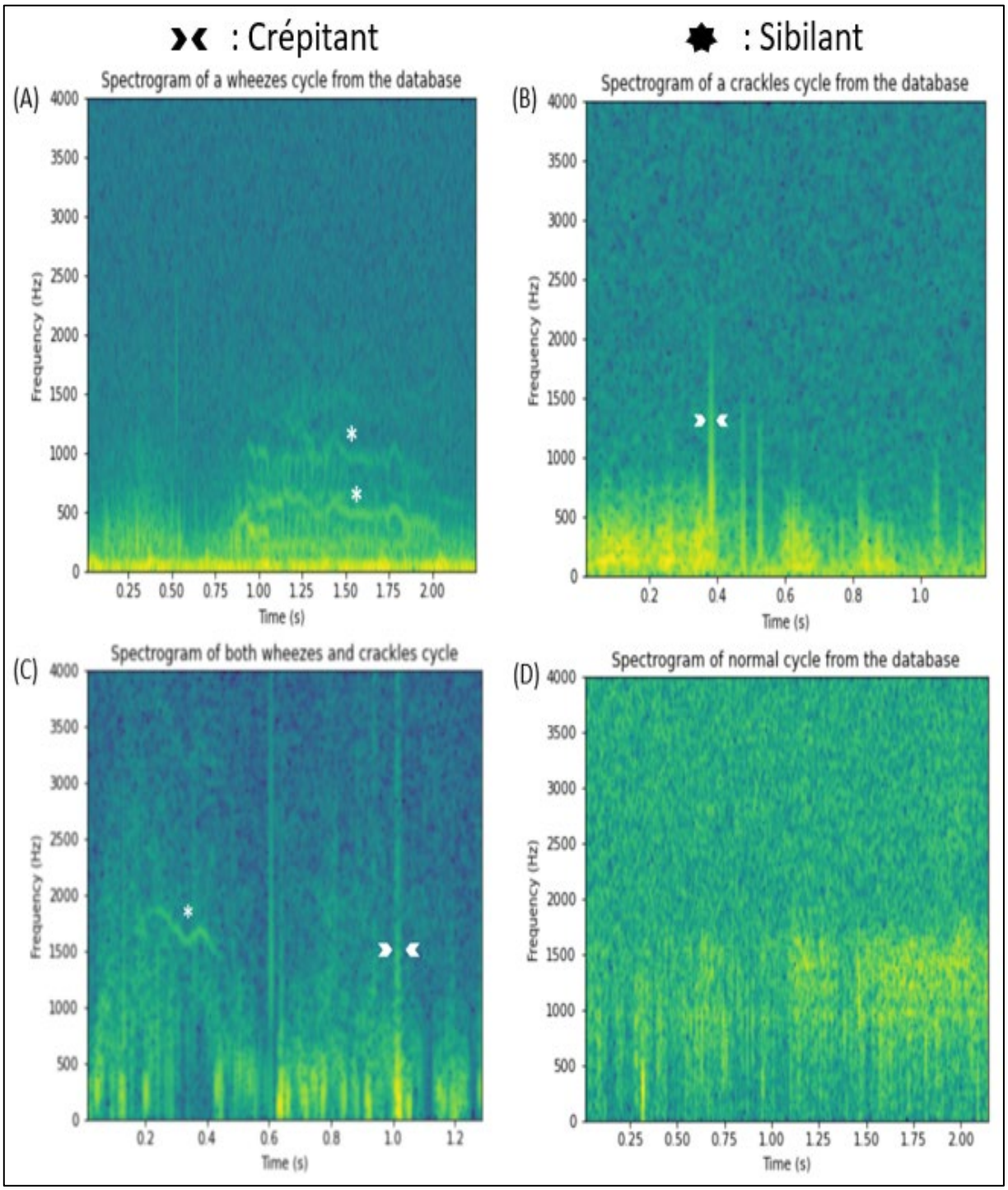


Figure 2.2. Exemples de spectrogrammes à base de STFT obtenus à partir de sons pulmonaires des quatre classes de la base de données ICBHI. (A) sibilants, (B) crépitants, (C) crépitants et sibilants, et (D) normaux.

De ce fait, la longueur de la fenêtre N_k est toujours inversement proportionnelle à la fréquence de la k^{ieme} composante spectrale f_k, ce qui permet d'avoir le même facteur Q pour tous les segments analysés. Considérons la variable n dans l'équation 2.11 comme étant :

$$n = i - m + \frac{N_k}{2} \tag{2.11}$$

Par conséquent, l'équation 2.8 s'écrit maintenant comme suit :

$$X^{CQ}(m,k) = \frac{1}{N_k} \sum_{n=0}^{N_k-1} x\left(m - \frac{N_k}{2} + n\right) w(\frac{n}{N_k}) e^{-j\frac{2\pi Q n}{N_k}} \tag{2.12}$$

Comme son nom l'indique, le coefficient de qualité Q dans l'équation 2.13 qui représente le rapport entre la fréquence centrale et la largeur de bande, doit absolument rester constant.

$$Q = \frac{f_k}{\Delta_k^{CQ}} = \frac{f_k}{f_{k+1} - f_k} = (2^{\frac{1}{b}} - 1)^{-1} \tag{2.13}$$

où b est le nombre de composantes fréquentielles par octave.

L'avantage de l'utilisation de la CQT par rapport à une transformée de Fourier discrète standard, est que cette dernière utilise une taille de fenêtre constante pour toutes les fréquences. Elle ne peut donc pas satisfaire les exigences de résolution temporelle et fréquentielle variables sur la large gamme de fréquences audibles, ce qui entraine certains problèmes lorsque nous transformons la fréquence sur une échelle logarithmique. Une transformée à Q constant (CQT), en revanche, cherche à résoudre ce problème en augmentant la taille de la fenêtre pour les basses fréquences et la réduisant pour les hautes fréquences. Cette caractéristique permet d'obtenir une meilleure résolution fréquentielle pour les basses fréquences et une meilleure résolution temporelle pour les hautes fréquences (Schörkhuber, 2010).

2.4 SPECTROGRAMME DE MEL

Le spectrogramme de Mel est une représentation temps-fréquence (RTF) qui a suscité un grand intérêt dans le domaine du traitement des signaux audio et les applications d'apprentissage profond. Un spectrogramme de Mel est un spectrogramme où les fréquences sont converties à l'échelle Mel à l'aide d'une transformation non linéaire de l'échelle des fréquences. Ceci afin d'imiter le système auditif qui ne perçoit pas les fréquences sur une échelle linéaire.

Il a été observé que l'oreille humaine agit comme un filtre lorsqu'elle se concentre uniquement sur des composantes de fréquence spécifiques. La relation entre l'échelle de Mel et l'échelle linéaire des fréquences est définie par les équations 2.14 et 2.15 :

$$f_{Mel} = 2595 \times log_{10}(1 + \frac{f}{700}) \tag{2.14}$$

$$f = 700 \times (10^{\frac{f_{Mel}}{2595}} - 1) \tag{2.15}$$

où f est la fréquence dans l'échelle linéaire et f_{Mel} la fréquence perçue. L'échelle de Mel est construite de telle sorte que les sons situés à une distance égale les uns des autres sur l'échelle fréquentielle de Mel, résonnent également pour les humains comme s'ils étaient à égale distance les uns des autres. Contrairement à l'échelle fréquentielle linéaire, où la différence entre 500 Hz et 1000 Hz est évidente, alors que la différence entre 7500 Hz et 8000 Hz est à peine perceptible.

Les bancs de filtres sont les éléments les plus importants lors de la création de cette RTF. Ils sont construits à l'aide de l'algorithme 1 qui propose la décomposition de l'échelle linéaire de Hertz en plusieurs intervalles, puis il transforme chaque intervalle en un intervalle correspondant dans l'échelle de Mel, en utilisant des filtres triangulaires superposés.

Algorithm 1 Génération de bancs de filtres

for Toutes les fréquences $\in [0, Fs/2]$ **do**
    Convertir les fréquences en (Hz) en échelle de Mel en utilisant l'eq.(10)
end for
Trouver le minimum et le maximum de la fréquence Mel
Choisir le nombre de bandes du banc de filtres
Créer des bandes de fréquences équidistantes dans l'intervalle $[Mel_{min}, Mel_{max}]$
for Toutes les points de fréquences $\in [Mel_{min}, Mel_{max}]$ **do**
    Convertir les fréquences en (Hz) en utilisant l'eq.(11)
    Arrondir au entier le plus proche des fréquences
end for
Générer et retourner les filtres triangulaires =0

Tout d'abord, l'entrée audio d'une taille N_x est devisé en segments entrelacés de taille N présentent un chevauchement de N_c échantillons. La fenêtre de pondération (fenêtre de Hanning dans notre cas) est par la suite appliquée à chaque trame, puis la trame est convertie en représentation dans le domaine de la fréquence avec une longueur FFT égale à N. Ensuite, la matrice du spectrogramme obtenue, d'une taille de $\left[\frac{N}{2}+1\ ,\ \frac{N_x-N_c}{N-N_c}\right]$ est passée à travers un banc de filtres Mel de taille $\left[N_{bandes},\ \frac{N}{2}+1\right]$ en utilisant une multiplication matricielle en multipliant les valeurs de la matrice du domaine des fréquences de la STFT par un banc de filtres de Mel. Ensuite, les valeurs spectrales en sortie du banc de filtres de Mel sont additionnées, et les canaux sont concaténés de sorte que la taille correspondante de la matrice de sortie est donnée par $\left[N_{bandes},\ \frac{N_x-N_c}{N-N_c}\right]$. L'étape finale consiste à appliquer le logarithme standard $\log_{10}$ à la matrice de sortie pour obtenir le spectrogramme de Mel.

Dans (Huzaifah, 2017), l'auteur a comparé l'impact de plusieurs techniques de représentation temps-fréquence sur les performances de classification des sons environnementales avec les réseaux de neurones convolutifs (CNN). Bien que les

spectrogrammes STFT linéaires et CQT aient donné de bons résultats sur certains modèles, les spectrogrammes Mel se sont avérés les plus efficaces dans toutes les variations testées.

La figure 2.3 présente le résultat de l'application des trois différentes techniques sur un cycle respiratoire sibilant extrait de la base de données ICBHI. Les figures 2.3(A), 2.3(B) et 2.3(C) correspondent aux spectrogrammes obtenus respectivement avec la transformée de Fourier à temps court (STFT), la transformée à Q constant (CQT) et la transformée de Mel (STFT-MEL).

À partir de la figure 2.3, on remarque que les résultats des représentations temps-fréquence présentées dans les figures 2.3(A) et 2.3(B) permettent une meilleure discrimination visuelle entre les fréquences basses et les fréquences hautes et celles-ci ont été obtenues respectivement par le spectrogramme Mel et CQT. La représentation basée sur la STFT figure 2.3(C) résulte en une mauvaise séparation des fréquences adjacentes, et ceci est dû à l'utilisation de fenêtres de longueur fixe contrairement aux représentations basées sur le spectrogramme Mel et le spectrogramme CQT qui utilisent des fenêtres de longueur variable. Les différentes représentations des signaux audio présentées dans la figure 2.3 ont été effectuées principalement à l'aide des bibliothèques Librosa et Matplotlib. Dans la section suivante, ces techniques seront utilisées en combinaison avec la méthode de décompositions en modes empiriques (EMD).

2.5 DÉCOMPOSIONS EN MODES EMPIRIQUES

La décomposition en modes empiriques (EMD) a été proposée par (Huang *et al*., 1998) pour analyser les signaux non stationnaires en composantes de fonctions modales intrinsèques (IMF). Cependant, l'EMD n'a pas été largement utilisée pour l'analyse des sons respiratoires. Le principe de la décomposition suppose que tout signal est constitué de différents modes d'oscillation intrinsèques. Chaque IMF représente une oscillation qui :

1. Possède le même nombre d'extrema, de minima et de passages par zéro.
2. Possède une symétrie par rapport à la moyenne locale.

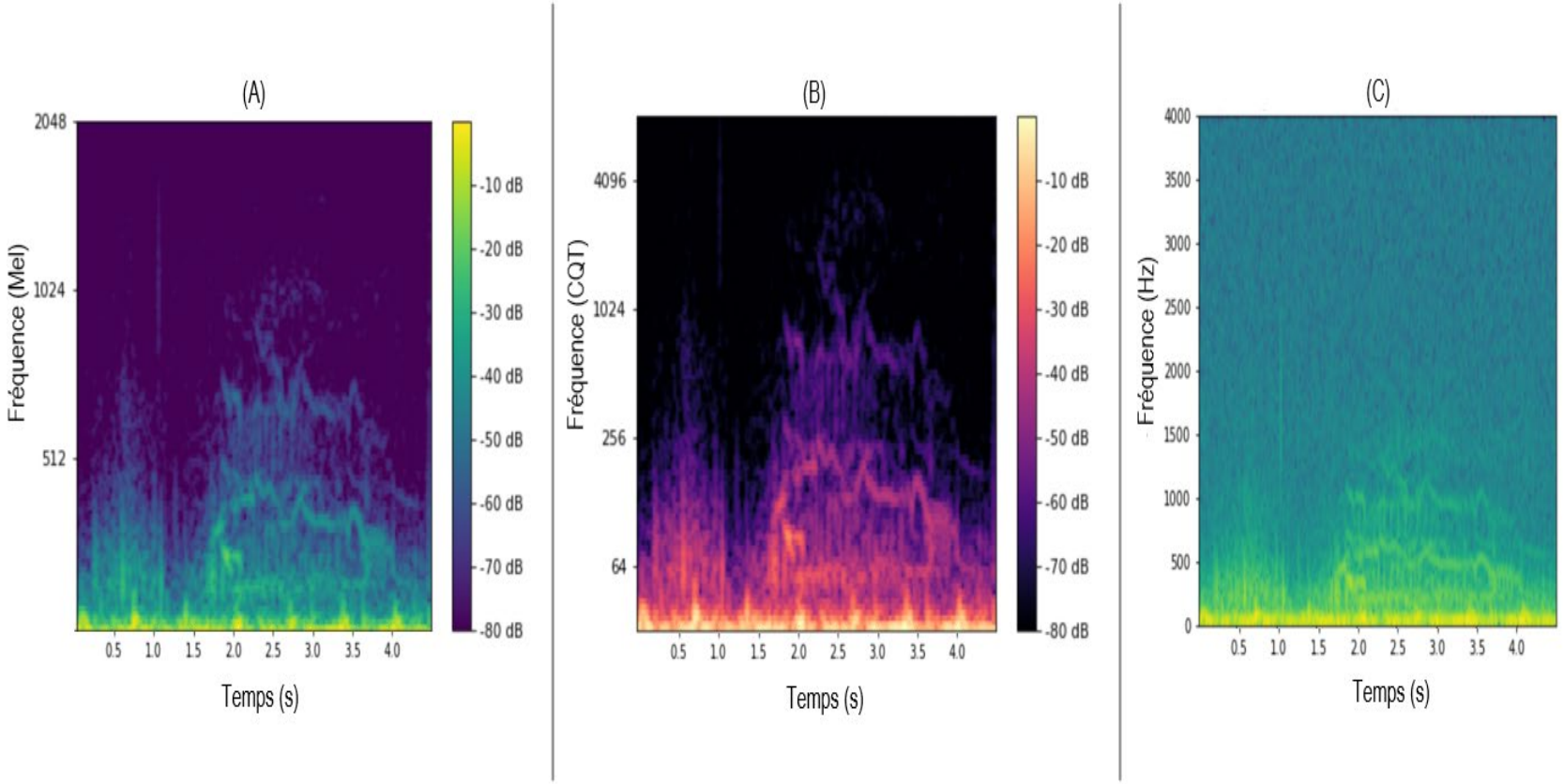


Figure 2.3. Le spectrogramme d'un sibilant obtenu par la transformée à l'échelle de Mel (A), la transformée à Q constant (B) et la transformée STFT (C).

Le principe de la méthode consiste à identifier empiriquement ces IMF en fonction de différentes échelles, puis à décomposer les données en conséquence par un processus appelé tamisage (sifting). Chaque itération de tamisage consiste en plusieurs étapes menant à l'extraction d'une IMF. En partant du signal à décomposer, à chaque étape, la moyenne des enveloppes supérieures et inférieures, obtenue par interpolation cubique des sommets locaux (maximums locaux) et des creux locaux (minimums locaux), respectivement, est soustraite du signal courant jusqu'à l'obtention d'une IMF comme montré dans la figure 2.4.

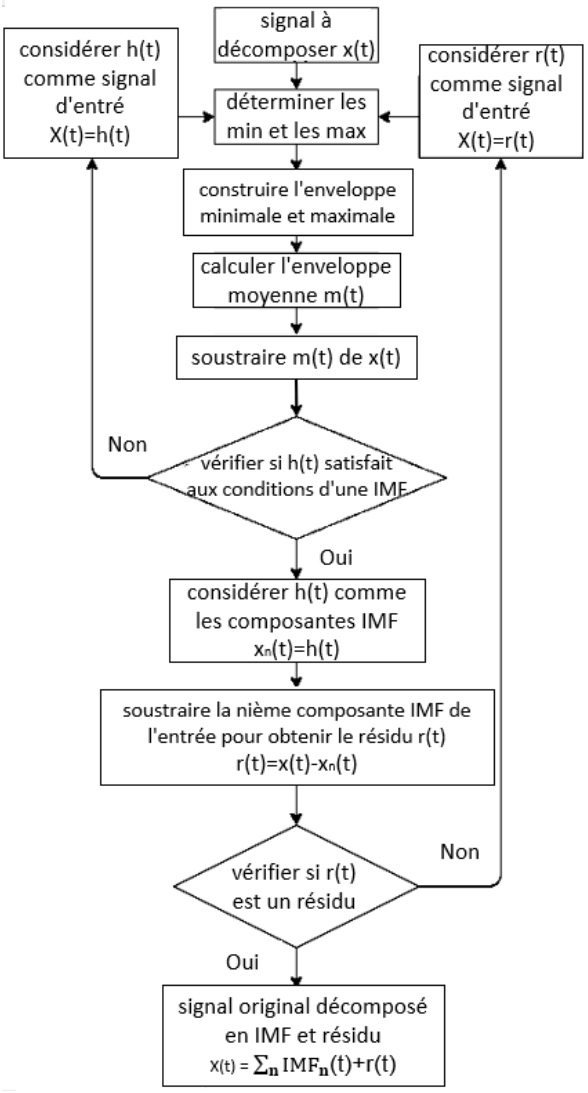


Figure 2.4. Diagramme de l'algorithme EMD (Zeiler *et al*., 2010).

Ensuite, un nouveau processus de tamisage est répété et ainsi de suite jusqu'à l'obtention de l'équation de décomposition qui modélise $x(t)$. Cette décomposition peut-être est exprimée par l'équation 2.16 (Zeiler *et al.*, 2010) :

$$x(t) = \sum_{n} IMF_n(t) + r(t) \tag{2.16}$$

Le processus de tamisage se poursuit tant que la condition d'arrêt n'est pas satisfaite. Toute la procédure s'arrête lorsque le résidu $r(t)$ est soit une constante, une pente monotone ou ne contient qu'un seul extrême.

Pour faciliter la compréhension de l'algorithme, un signal $x(t)$ obtenu par le mélange de deux composantes sinusoïdales de 15 Hz et 50 Hz a été utilisé. Il est défini par l'équation 2.17 et illustré dans la figure 2.5.

$$x(t) = x_1(t) + x_2(t) = 2\sin(2\pi \times 15 \times t) + 4\cos(2\pi \times 50 \times t) \tag{2.17}$$

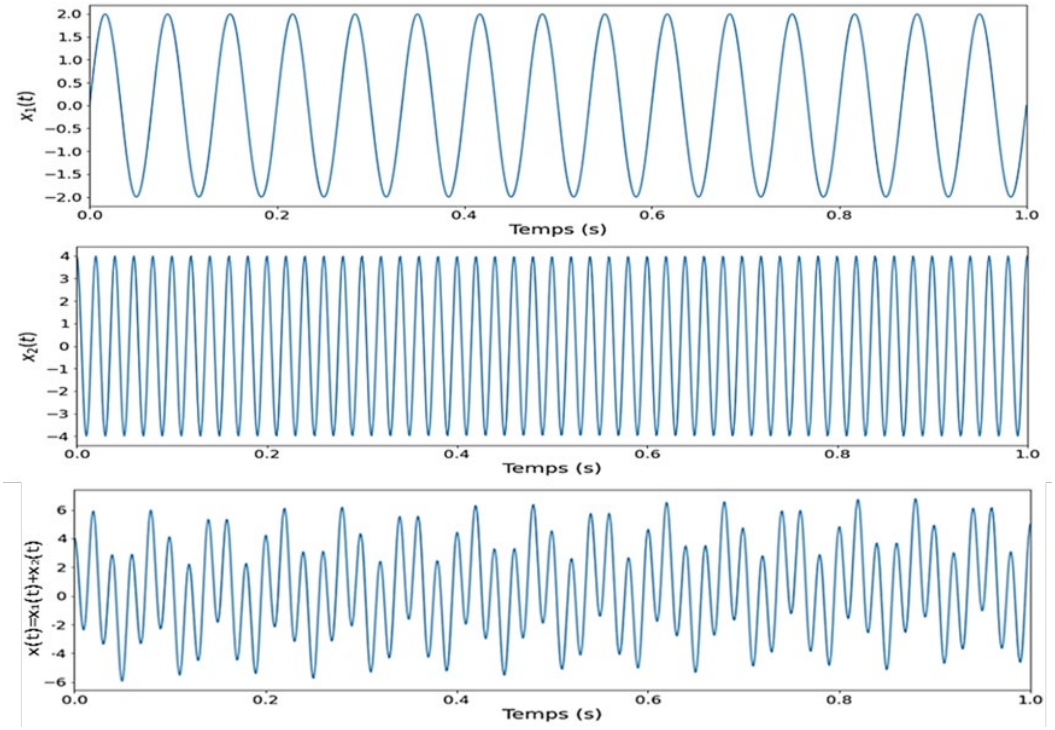


Figure 2.5. Obtention du signal x(t) par fusion des signaux x_1 (t) et x_2 (t).

Étape 1, Déterminer les valeurs minimales et maximales du signal x(t), tel qu'illustré à la figure 2.6

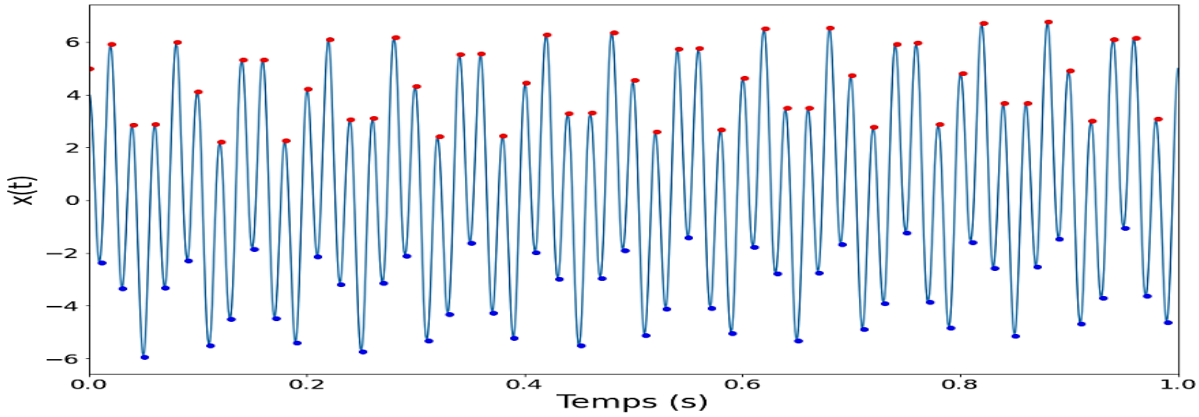


Figure 2.6. Détection des minimums et les maximums du signal x(t).

Étape 2, Interpoler les valeurs des minimums pour créer une enveloppe inferieure $e_{min}(t)$. Interpoler les valeurs des maximums pour créer une enveloppe supérieure $e_{max}(t)$, comme illustré dans la figure 2.7.

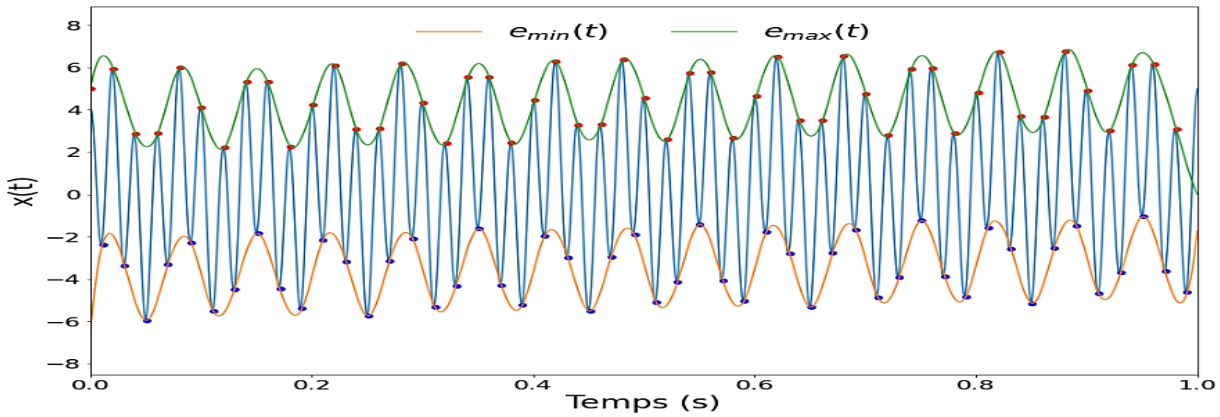


Figure 2.7. Création de l'enveloppe des minima et maxima.

Étape 3, À partir de l'enveloppe des minimums et des maximums, nous obtenons l'enveloppe moyenne $e_{moy}(t)$, tel qu'illustré à la figure 2.8.

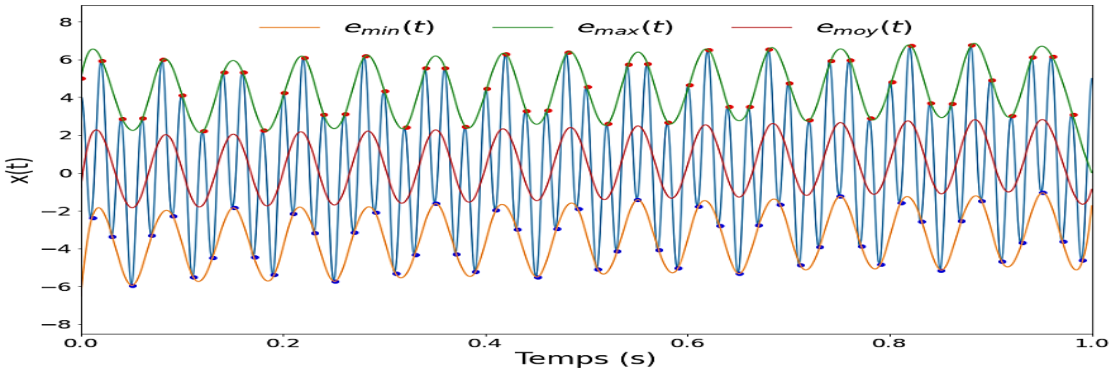


Figure 2.8. Création de l'enveloppe moyenne.

Étape 4, Soustraire l'enveloppe moyenne du signal orignal pour obtenir le signal $h(t)$, présenté dans la figure 2.9.

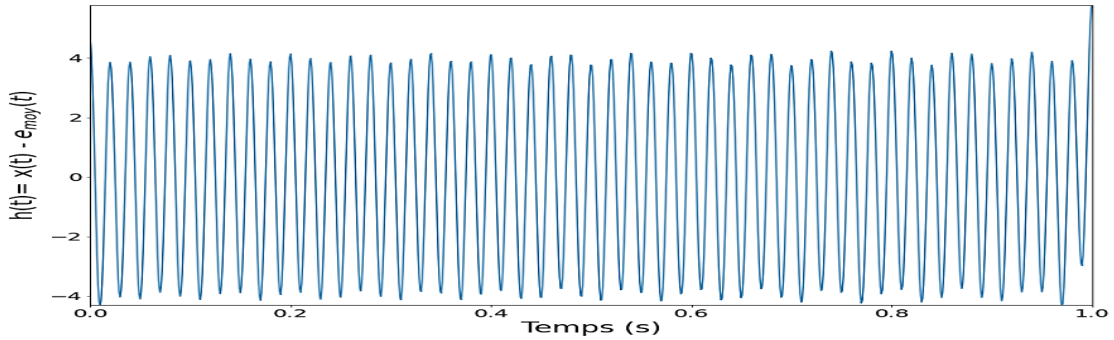


Figure 2.9. Signal récupéré après la soustraction.

Étape 5, Vérifier si le signal $h(t)$ extrait est une IMF. Selon les deux critères de l'oscillation, la moyenne est proche de zéro. Par conséquent, nous pouvons l'ignorer et l'arrondir à zéro. De plus, $h(t)$ possède le même nombre de minimums et de maximums remplissant les conditions requises. Ainsi, nous pourrions le considérer comme une IMF.

Étape 6, Soustraire la première composante IMF du signal original pour obtenir le signal $r(t)$.

Étape 7, Vérifier si $r(t)$ est un résidu selon les conditions d'arrêt, sinon considérer $x(t) = r(t)$. Répéter le processus (de l'étape 1 à l'étape 7) jusqu'à ce que nous obtenions une constante, une pente monotone ou un signal qui ne contient qu'un seul extrême.

À la suite de la décomposition, le signal $x(t)$ peut-être décrit par l'équation 2.18 :

$$x(t) = IMF_1 + IMF_2 + IMF_3 + IMF_4 + IMF_5 + IMF_6 + IMF_7 + r(t) \qquad (2.18)$$

Étant donné que $IMF_2 = 4\cos(2\pi \times 50 \times t)$ et $IMF_4 = 2\sin(2\pi \times 15 \times t)$. À partir de la figure 2.10 on peut constater que la reconstruction parfaite des signaux n'est pas garantie en utilisant cette décomposition, car elle dépend fortement du choix des paramètres de l'algorithme comme le critère d'arrêt, les conditions aux limites et la méthode d'interpolation (Zeiler *et al*., 2010). Ces paramètres ne font pas l'objet de cette étude.

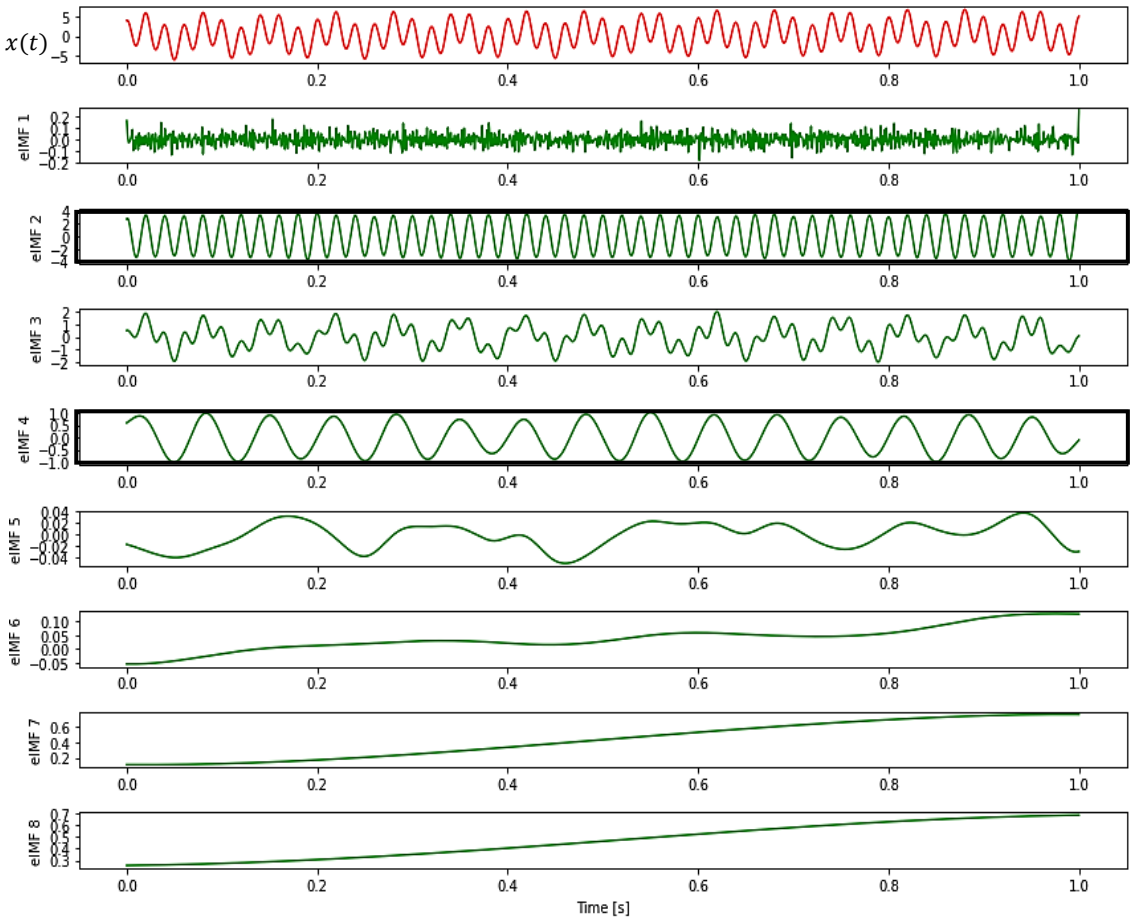


Figure 2.10. Décomposition du signal test par EMD.

Le principal avantage de cette méthode est que la fréquence instantanée (FI), estimée par EMD fournit des informations sur le contenu fréquentiel des signaux audio à chaque instant (Lozano *et al*., 2013), ce qui la rend similaire à la transformée STFT. Cette dernière suppose que le signal est périodique et que ses composantes de base sont constituées de diverses ondes sinusoïdales simples. En revanche, la principale différence avec l'EMD est que la sortie de cette dernière reste dans le domaine temporel et ne suppose pas la périodicité du signal.

Dans ce projet de recherche, nous avons décomposé les signaux respiratoires en 6 IMFs. Cependant, seulement la première IMFs a été retenue (IMF_0), car elle fournit plus d'informations pour ce type spécifique de sons. Le cadre proposé est illustré à la figure 2.11. Tout d'abord, les sons pulmonaires originaux sont passés à travers l'algorithme EMD pour extraire toutes les IMF possibles. Ensuite, nous examinons ces IMF en utilisant séparément la STFT, la CQT et le spectrogramme de Mel.

La figure 2.12 montre un exemple de la méthode du spectrogramme basé sur l'analyse EMD appliquée à un cycle respiratoire. Ce cycle a été précédemment défini comme un cycle de respiration sifflante. On peut constater depuis cette figure que la plupart des composantes du sifflement se trouvent dans les deux premières composantes.

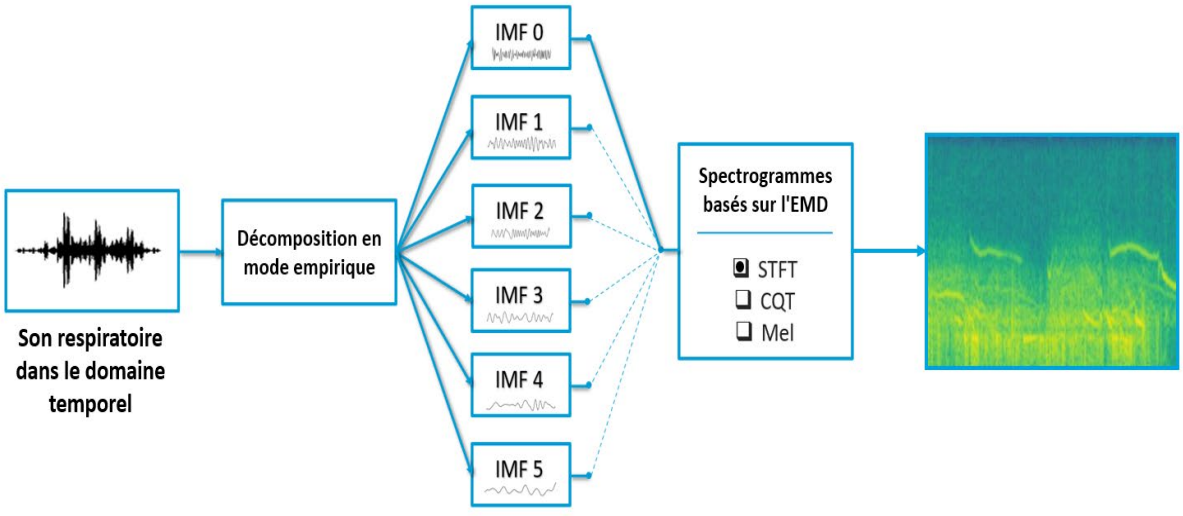


Figure 2.11. Structure proposée pour la création de spectrogrammes basée sur l'EMD.

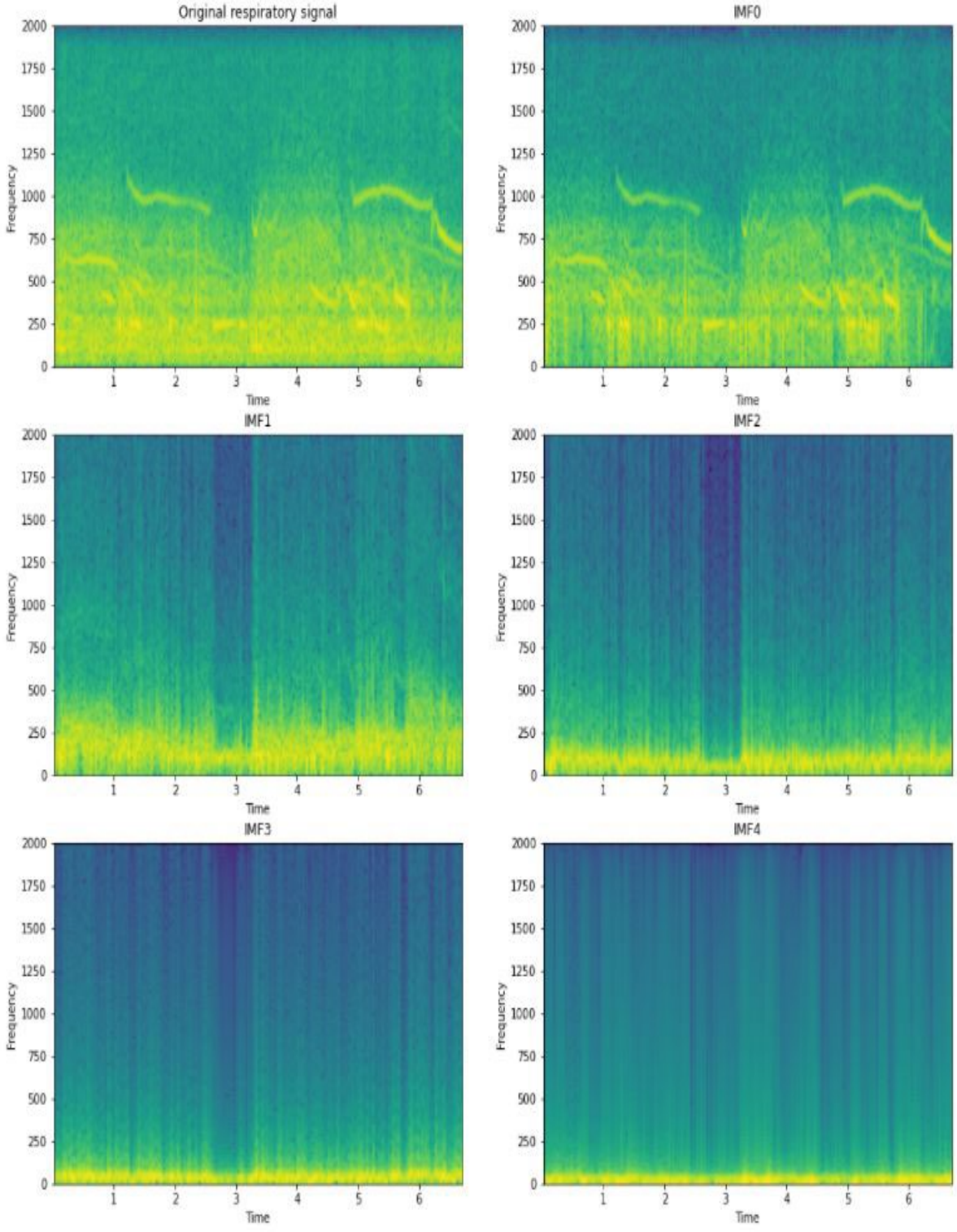


Figure 2.12. Spectrogramme basé sur l'EMD des IMF0, IMF1, IMF 2, IMF 3, IMF 4, et IMF 5 et son équivalent original pour un son respiratoire sibilant.

CHAPITRE 3
MODÉLISATION ET CLASSIFICATION DES SONS RESPIRATOIRES

Ce chapitre décrit la méthode de classification des sons respiratoires utilisée dans ce projet. Cette méthode fonctionne en deux étapes : l'apprentissage et le test. Lors de la phase d'apprentissage, un ensemble des images provenant des représentations temps-fréquence (RTF) est modélisé puis un discriminant est établi pour délimiter ces classes. Cette méthode de classification est une méthode courante d'apprentissage supervisé. Dans la phase de test, le réseau va recevoir de nouvelles données sous forme d'images qui n'ont jamais été rencontrées par le classifieur lors de la phase d'apprentissage. Par la suite, ce dernier va déterminer automatiquement la classe à laquelle appartient chaque nouvelle donnée.

3.1 REVUE DE LA LITTÉRATURE

La modélisation d'un processus de classification consiste à représenter son comportement à l'aide d'un modèle mathématique paramétré. Étant donné la problématique de recherche que nous désirons résoudre, nous devrons investiguer et obtenir des données de qualité que nous utiliserons pour améliorer notre système. La qualité et la quantité d'informations à l'entrée sont des facteurs très importants, car ils auront un impact direct sur le fonctionnement le modèle conçu. Plusieurs techniques ont été utilisées pour concevoir des modèles de classification. Parmi les techniques utilisées ces dernières années et qui permettent de résoudre des problèmes très complexes, on trouve les techniques d'apprentissage profond.

De nombreux travaux ont été menés récemment pour développer des systèmes de classification de sons respiratoires en utilisant la base de données ICBHI. La première tentative a été menée par deux équipes de recherche de deux pays différents (Rocha *et al*., 2019), qui ont proposé d'utiliser à la fois une étape de prétraitement, suivie d'un

rééchantillonnage puis d'une étape de débruitage. Ensuite, ces auteurs se sont basés sur la transformée de Fourier à court terme (STFT), les coefficients d'ondelettes et les coefficients cepstraux à l'échelle de Mel (MFCC) pour la phase d'extraction des caractéristiques. La dernière étape consistait à classer les différentes catégories de sons respiratoires à l'aide de méthodes de classification telles que les machines à vecteurs de support (SVM) et les modèles de Markov cachés (HMM). Depuis 2018, plusieurs chercheurs utilisent des réseaux de neurones à apprentissage profond pour la classification de sons respiratoires en utilisant la base de données ICBHI (Liu *et al*., 2019; García *et al*., 2020; Perna, 2018; Aykanat *et al*., 2017; Shuvo *et al*., 2020; Liu *et al*., 2019; Rocha *et al*., 2020; Kim *et al*., 2021).

L'extraction de spectrogrammes à partir des enregistrements audio et leur utilisation comme images d'entrée pour les réseaux CNN ont été largement utilisées dans ces systèmes, car ces représentations temps-fréquence (RTF) sont capables de fournir des informations temporelles et fréquentielles ainsi que de proposer un contexte temporel beaucoup plus large que l'analyse d'une seule fenêtre à l'aide des techniques d'extraction de caractéristiques traditionnelles.

Plusieurs représentations temps-fréquence ont été utilisées, comme le spectrogramme à base de STFT (Demir *et al*., 2020a; Demir *et al*., 2020b; Liu *et al*., 2019; Chen *et al*., 2019; Nakano *et al*., 2019; Jácome *et al*., 2019; Bardou *et al*., 2018), le spectrogramme Log-Mel (Pham *et al*., 2020), la transformée à Q constant (CQT) (Pham *et al*., 2020), le spectrogramme à base de filtre gammatone (Pham *et al*., 2020), les Mel-spectrogrammes (Gairola *et al*., 2020), le scalogramme hybride basé sur la décomposition en modes empiriques (EMD) en utilisant des ondelettes continues (Shuvo *et al*., 2020) et la S-Transformée optimisée (Chen *et al*., 2019). La figure 3.1 présente un graphique en anneau des méthodes de RTF répertoriées à partir de 30 publications consacrées à la classification des sons respiratoires.

Les systèmes de classification actuels d'apprentissage profond qui utilisent les spectrogrammes sont principalement basés sur des réseaux de neurones convolutifs (CNN), des réseaux neuronaux récurrents (RNN) ou des architectures hybrides.

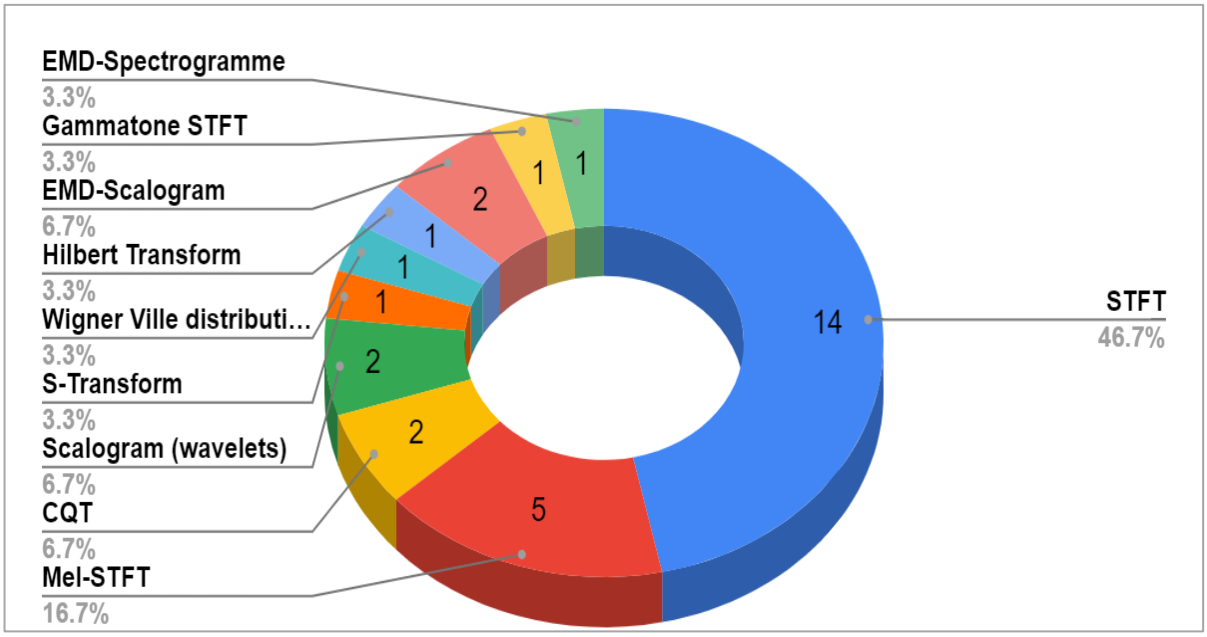


Figure 3.1. Résumé des méthodes de RTF utilisées dans la littérature liées à analyse de sons respiratoires.

Dans Perna et Tagarelli (2019), les auteurs ont utilisé des coefficients MFCC comme vecteur de caractéristiques avec quatre architectures différentes de réseaux de neurones récurrents (RNN), à savoir le réseau de neurones récurrent à longue mémoire à court terme (LSTM), réseau de neurones récurrent à portes (GRU), LSTM bidirectionnel (BiLSTM) et le GRU bidirectionnel (BiGRU). Les méthodes de classification qui sont basées sur des réseaux de neurones convolutifs (CNN) ont été utilisées dans (Liu *et al*., 2019; García *et al*., 2020; Perna, 2018; Aykanat *et al*., 2017; Shuvo *et al*., 2020; Liu *et al*., 2019; Rocha *et al*., 2020; Kim *et al*., 2021).

La plupart des travaux qui utilisent cette base de données ICBHI se sont concentrés sur la classification binaire (sons normaux et sons adventices) (Aykanat *et al*., 2017), la classification chronique ternaire (en bonne santé, en mauvaise santé ou avec maladies chroniques) (Perna et Tagarelli, 2019) et la prédiction multiclasses basée sur la pathologie. Pour la prédiction de la pathologie, les auteurs ont utilisé les informations fournies par la base de données pour chaque enregistrement permettant de déterminer le cas de chaque

patient par rapport à sept classes des maladies respiratoires suivantes : Broncho pneumopathie chronique obstructive (BPCO), Bronchiectasie, Asthme, Infection des voies respiratoires supérieures et inférieures, Pneumonie et Bronchiolite. Cependant, la classification basée sur les anomalies (classification en quatre classes de sons respiratoires) constitue la tâche la plus difficile (Rocha *et al*., 2020).

Compte tenu de la classification des sons respiratoires en quatre classes à savoir, les sons normaux, les sibilants, les crépitants et la combinaison des crépitants et sibilants, nous pouvons trouver plusieurs architectures proposées comme l'architecture C-DNN qui est basée sur l'architecture VGG-7 et le model CNN-MoE qui rassemble la combinaison de l'architecture des réseaux CNN avec un ensemble de mélange d'experts dans (Pham *et al*., 2020). Ces architectures ont été proposés notamment pour apprendre les caractéristiques spatiales et temporelles à partir de différentes combinaisons de spectrogrammes qui ont été mélangés ensemble.

Le modèle VGG-16 (Visual Geometry Group) qui représente une architecture très efficace à base d'un réseau de neurones convolutif a été utilisé par Demir *et al*. (2020b). Dans son architecture CNN, une couche de pooling moyen et une couche de Max Pooling sont connectées en parallèle afin d'améliorer les performances de classification. Les caractéristiques sont utilisées comme entrée du classifieur d'analyse discriminante linéaire (LDA) en utilisant la méthode des ensembles de sous-espaces aléatoires (RSE). Une autre méthode qui utilise un ensemble de réseaux de neurones convolutifs (CNN) a obtenu un score de 78.4 % pour la tâche de prédiction des quatre classes (Nguyen et Pernkopf, 2020). Pham *et al*. (2021), ont proposé une amélioration de leur modèle précédent en collaborant avec Nguyen et Pernkopf (2020) afin d'utiliser une combinaison des deux techniques. Au stade final, ils ont utilisé un ensemble de CNN-MoE avec un réseau de neurones récurrents de type continu (C-RNN) pour la classification, ce qui a donné un score de 80% pour la tâche de prédiction des quatre classes. L'architecture RespireNet a été proposée par une équipe de Microsoft en se basant sur l'architecture de ResNet-43. Elle utilise pratiquement la même méthodologie que celle utilisée dans les travaux précédents avec quelques

techniques de réglage et d'ajustement spécifiques au dispositif, d'augmentation de données basée sur la concaténation, découpage des régions vides et de remplissage intelligent afin d'obtenir une même durée dans cycles respiratoires (Gairola *et al*., 2020).

Après cette étude, nous avons pu identifier quelques architectures d'apprentissage profond et tirer quelques conclusions sur les méthodes qui pourraient améliorer le résultat de cette classification de sons respiratoires en quatre classes. À notre connaissance, le meilleur système qui a abordé la classification des sons respiratoires en quatre classes a été publié dans (Pham *et al*., 2021).

3.2 RÉSEAU DE NEURONES CONVOLUTIF (CNN)

Les réseaux de neurones convolutifs (CNN) sont des concepts basés sur l'apprentissage profond pour le traitement des images. Inspiré de la structure du cortex visuel chez les animaux, les CNN sont conçus pour apprendre automatiquement et de manière adaptative, passant des caractéristiques de bas niveau, comme les contours, jusqu'à des caractéristiques de haut niveau telles que la classification d'une scène dans son ensemble ou la détection et la reconnaissance d'objets (Lecun *et al.*, 1998). En effet, une critique courante des réseaux de neurones profonds est qu'ils se comportent comme des boites noires et que personne ne peut vraiment comprendre pourquoi et comment ils fonctionnent si bien (Zeiler et Fergus, 2013). Les travaux de Zeiler et Fergus (2013) constituent un premier pas important vers l'ouverture des réseaux de neurone convolutifs pour montrer leurs processus internes, et comment ils finissent par réagir à des caractéristiques particulières ainsi comment ils apprennent des concepts plus abstraits à chaque fois qu'ils approfondissent. Après la révolution de l'apprentissage profond en 2012, l'année où Alex Krizhevsky (Krizhevsky *et al*., 2012) a utilisé des réseaux de neurones convolutifs pour remporter le concours ImageNet de cette année-là avec le réseau AlexNet, en réduisant l'erreur de classification de 26 % à 15 %. Depuis ce temps-là, la recherche de systèmes de classification plus performants basés sur les réseaux CNN a pris un nouvel élan.

3.3 ARCHITECTURE DE BASE DES RÉSEAUX CNN

Les réseaux de neurones convolutifs (CNN) ont été introduits pour résoudre certaines lacunes des réseaux de neurones traditionnels. Ces derniers présentent deux limitations majeures dont le nombre élevé de paramètres et le manque de capacité de calcul spatial. En général, les architectures CNN sont composées de plusieurs couches de convolutions suivies d'une couche d'activation d'unité linéaire rectifiée (ReLU), puis d'une couche de mise en commun (Pooling), puis de nouvelles couches de convolutions, et ainsi de suite. Contrairement aux réseaux neuronaux traditionnels, où chaque caractéristique d'entrée est associée à des paramètres distincts, les paramètres des réseaux CNN sont partagés entre les neurones de la même carte de caractéristiques, ce qui permet au réseau de réduire le nombre de paramètres à former (voir figure 3.2)

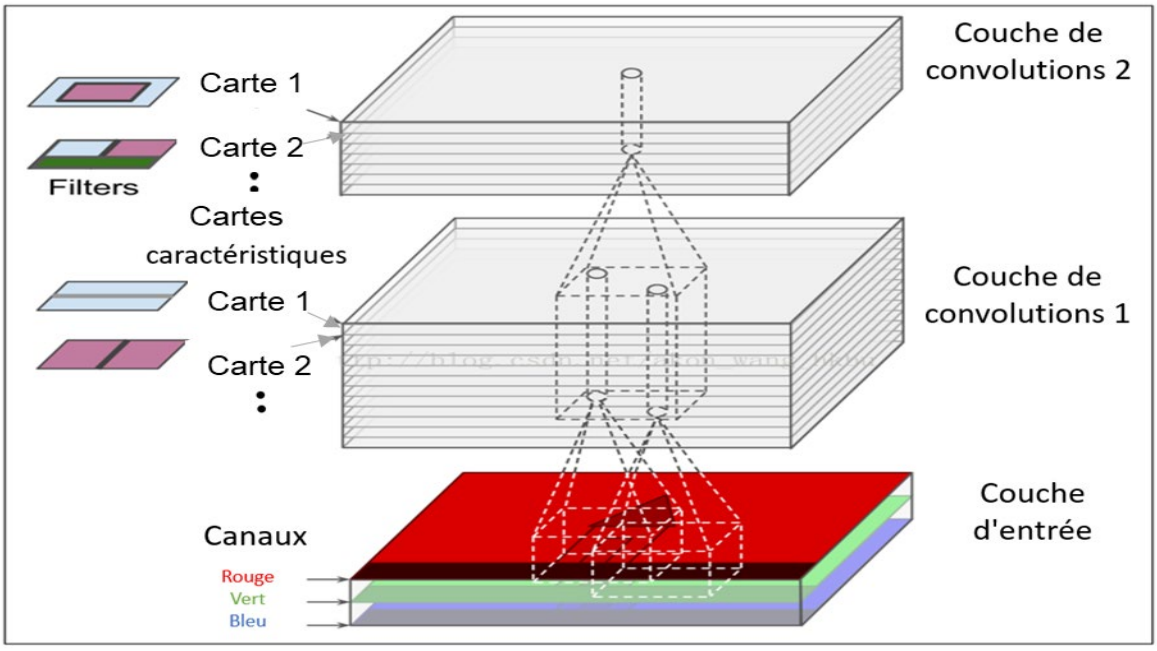


Figure 3.2. Des couches de convolutions avec plusieurs cartes de caractéristiques. Chaque couche de convolution possède des sorties de filtres multiples, soit une carte de caractéristiques par filtre. Par conséquent, les neurones de différentes cartes de caractéristiques utilisent des paramètres différents (Géron, 2019).

Chaque neurone dans un réseau de neurones convolutif (CNN) n'a accès qu'à certains éléments de la région voisine de la couche précédente. Cette région (généralement carrée)

est appelée champ réceptif des neurones (taille du filtre). De cette façon, les réseaux CNN peuvent apprendre les caractéristiques locales essentielles et ainsi de préserver la localisation des caractéristiques de l'image (Géron, 2019).

Pour les images, un réseau de neurones convolutif (CNN) prend en entrée des données tridimensionnelles (hauteur H × largeur W × profondeur D), chaque neurone z de la première couche de convolution n'est connecté qu'aux pixels de son champ réceptif de taille (K_H, K_W) de l'image d'entrée. À son tour, chaque neurone situé dans la position (i, j) de la couche suivante est connectée aux sorties des neurones de la couche précédente dans un petit rectangle qui s'étend de rangées $(i \times S_H)$ $jusqu'$à $(i \times S_H + K_H - 1)$, et de colonnes $(j \times S_W)$ $jusqu'$à $(i \times S_W + K_w - 1)$, où S_H et S_w sont respectivement les pas vertical et horizontal.

Les réseaux de neurones convolutifs (CNN) sont nommés d'après les couches de convolution, qui sont au cœur de leur architecture. Chaque couche de convolution dans un réseau CNN regroupe plusieurs cartes de caractéristiques de profondeur D, et dans chaque carte de caractéristiques données de la couche de convolution l, nous avons les mêmes poids et termes de biais partagés, tandis que les neurones de différentes cartes de caractéristiques utilisent des paramètres différents. Tous les neurones situés à la position (i, j), provenant de différentes cartes de caractéristiques sont connectés à la même région de neurones de la couche précédente. De point de vue mathématique, la sortie $z_{i,j}$ d'un neurone dans une couche de convolution, peut être exprimée par l'équation 3.1 (Planche et Andres, 2019) :

$$z_{i,j} = \varphi\left(b_{i,j} + \sum_{u=0}^{K_H-1}\sum_{v=0}^{K_W-1}\sum_{w=0}^{D-1}\omega_{u,v,w} \cdot x_{i+u,j+v,w}\right) \tag{3.1}$$

où $x_{u,v,w}$ est la sortie du neurone situé dans la couche précédente $(l-1)$, $\omega_{u,v,w} \in \mathbb{R}^{K_H \times K_W \times D}$ représente les poids du neurone, $b_{i,j} \in \mathbb{R}$ est le terme de biais, et φ est la

fonction d'activation telle que, la sigmoïde, la tangente hyperbolique ou l'unité linéaire rectifiée (ReLU), qui sont les plus couramment appliquées à ce type de réseau.

La construction d'un modèle avec plus de couches permet d'apprendre des caractéristiques plus abstraites, mais en même temps, il sera aussi plus susceptible de devenir plus vulnérable quant au surentrainement. Ajoutons que la nature de nos données n'est pas complexe, mais très sensible, et que le nombre de données disponibles pour l'entrainement est limité, ce qui rend la classification plus difficile.

La couche de pooling réalise une opération typique de sous-échantillonnage qui réduit la dimensionnalité des cartes de caractéristiques. Cette partie de l'architecture permet d'atteindre une invariance en translation aux petits décalages et de distorsions, de diminuer le nombre de paramètres apprenables et de réduire le temps de calcul (Géron, 2019; Yamashita *et al*., 2018). Il existe deux types de couches de Pooling souvent utilisés, à savoir le Max Pooling et le Average Pooling. Avec le Max Pooling, seule la valeur la plus élevée d'une région spécifique sera prise en compte. Cette région est souvent utilisée dans des noyaux 2x2. D'autre part, le Average Pooling est fondé sur le même principe que le Max Pooling, mais cette fois, elle renvoie la moyenne de toutes les valeurs de la partie de l'image couverte par le noyau 2x2.

La dernière couche de l'architecture CNN est appelée la couche entièrement connectée (Full connected layer). L'objectif de cette couche est de combiner les caractéristiques multidimensionnelles de haut niveau apprises par les couches de convolution en un vecteur unidimensionnel. Le vecteur colonne obtenu sera transmis à la couche de sortie, où un classificateur softmax ou SVM est utilisé pour prédire l'étiquette de la classe en entrée (Demir *et al*., 2020a). Il faut noter que ces couches, entièrement connectées sont souvent appelées couches denses, ou simplement denses.

3.4 PRÉSENTATION DES ARCHITECTURES CNN AVANCÉES

Dans cette partie, nous présentons deux architectures d'apprentissage profond les plus répandues comme AlexNet et VGG, en expliquant les raisons de leurs développements et

l'importance de leurs contributions. Parce qu'au final, nous prendrons en compte leurs différentes contributions pour créer notre architecture afin de relever le défi associé à la base de données ICBHI.

3.4.1 Contributions apportées par l'architecture AlexNet

Comme mentionné précédemment, l'architecture AlexNet a apporté un véritable changement, en étant le premier réseau CNN entraîné avec succès pour une tâche de reconnaissance complexe et en apportant plusieurs contributions qui sont toujours valables aujourd'hui. Par exemple, l'utilisation de la fonction unité linéaire rectifiée (ReLU), comme fonction d'activation, permet d'éviter le problème du gradient évanescent et d'améliorer ainsi l'apprentissage. Une deuxième contribution concerne l'application du dropout aux CNNs qui permet de réduire le nombre de neurones d'activation dans la couche entièrement connectée. Le dropout présente aussi l'intérêt de réduire le risque surentrainement (overfitting). Et finalement, l'application des transformations aléatoires (translation d'image, rotation horizontale, etc.) pour augmenter synthétiquement l'ensemble de données.

3.4.2 Aperçu de l'architecture VGG

Cette architecture a été développée par le groupe VGG (Visual Geometry Group) de l'Université d'Oxford. Bien que le groupe n'ait obtenu que la deuxième place dans la tâche de classification du défi ILSVRC en 2014, leur méthode a influencé de nombreuses architectures récentes. Les deux architectures les plus performantes, encore couramment utilisées de nos jours, sont appelées VGG-16 et VGG-19. Les nombres (16 et 19) représentent la profondeur de ces architectures CNN. Dans leur article (Simonyan et Zisserman, 2014), Simonyan et Zisserman ont présenté six architectures CNN différentes, de 11 à 25 couches de profondeur. Chaque réseau est composé de cinq blocs de plusieurs convolutions consécutives, suivis d'une couche de Max Pooling et de trois couches denses finales (avec dropout pour l'entrainement). Dans l'ensemble, les réseaux AlexNet et VGG typiques sont représentés dans la Figure 3.3.

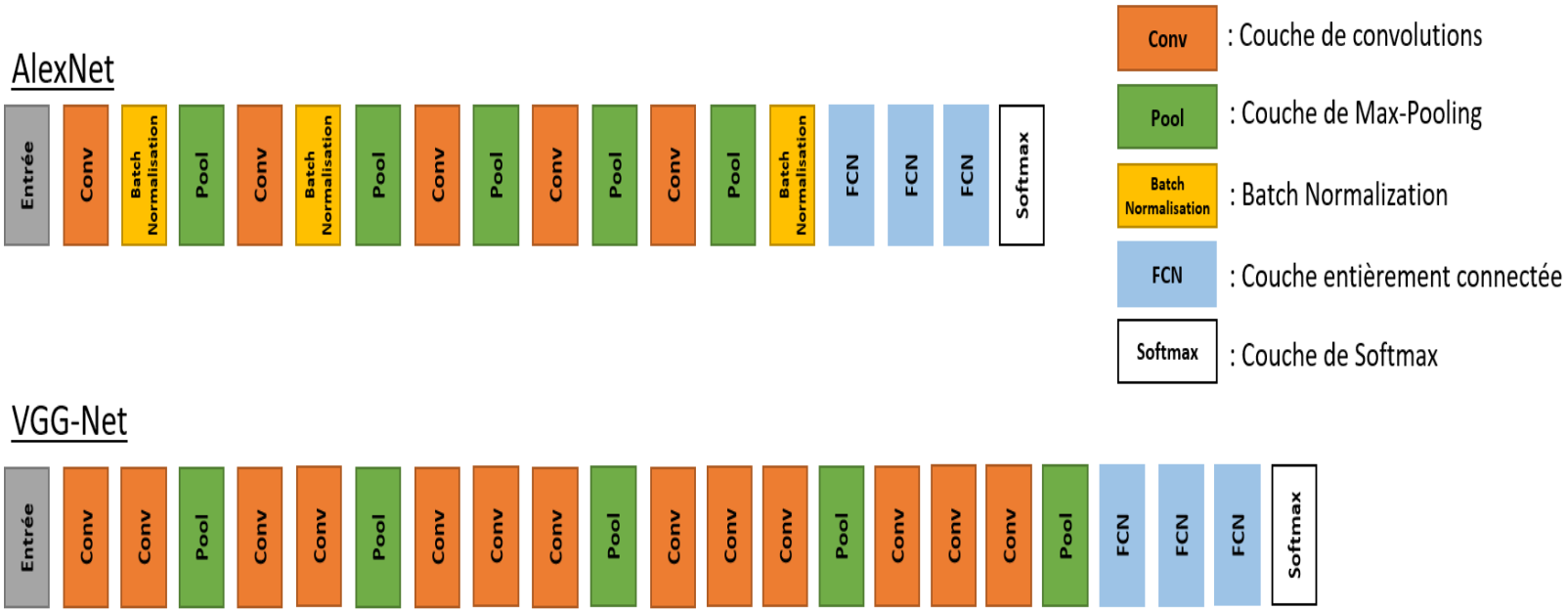


Figure 3.3. Architecture simplifiée des réseaux Alex-Net et VGG-16.

3.5 ARCHITECTURE PROPOSÉE

Le modèle CNN que nous avons proposé dans cette étude, comprend quatre couches de convolutions avec des nombres différents de filtres et de pas. La topologie du modèle proposé est illustrée à la figure 3.4. La première et la deuxième couche sont constituées de 32 noyaux de filtrage de même taille 3x3, tandis que la troisième et la quatrième sont constituées de 64 noyaux de filtrage de taille similaire 3x3. Le choix d'une taille de noyau fixe de 3x3 est reconnu comme le choix optimal adopté par les praticiens du domaine d'apprentissage machine jusqu'à présent, qui donne dans la plupart des cas de meilleurs résultats. Les couches de Max Pooling ont été appliquées avec différentes tailles de noyaux de Pooling pour adapter la taille de l'image à notre architecture. De plus, ces couches nous aideront à ne conserver que les caractéristiques les plus importantes et à réduire les paramètres appris. Nous avons implémenté d'autres blocs CNN qui effectuent la normalisation par lot (Batch normalisation), les fonctions d'activation (ReLU et LeakyReLU), le dropout moyen et les régulateurs L2 pour aider notre modèle à éviter le surentrainement tout en obtenant le meilleur résultat avec un faible coût de calcul (Chanane et Bahoura, 2021).

3.6 ENTRAINEMENT DU RESEAU

L'entrainement d'un réseau consiste à définir les paramètres des noyaux dans les couches de convolution et les poids dans les couches entièrement connectées qui minimisent la différence entre les prédictions de sortie (classe prédite) et les étiquettes de vérité (classe réelle). L'algorithme de rétropropagation basé sur la décente de gradient est l'une des méthodes les plus utilisées au cours de la dernière décennie, cependant les architectures de réseaux de neurones modernes ne sont pas les seules à avoir connu des améliorations au fil de cette décennie (Géron, 2019). En effet, la manière dont ces réseaux sont entrainés a également évolué, améliorant la fiabilité et la rapidité avec laquelle ils peuvent converger. Les algorithmes du taux d'apprentissage adaptatif sont plus performants que la descente de gradient et ses variantes en termes de vitesse, particulièrement dans le cas des réseaux de neurones profonds.

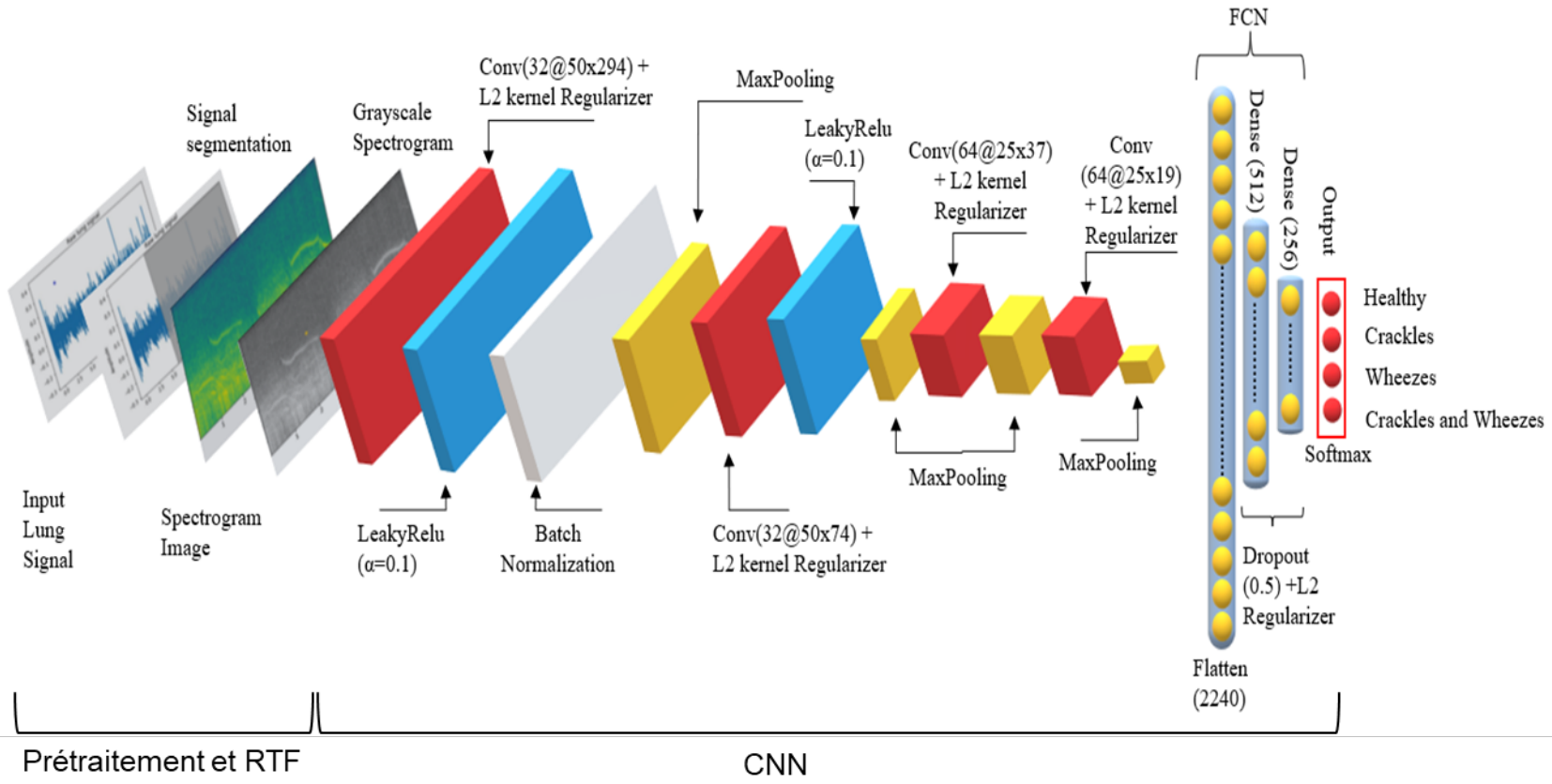


Figure 3.4. Topologie du réseau de neurones convolutif proposé (Chanane et Bahoura, 2021).

Dans cette étude, nous avons utilisé l'un des optimiseurs le plus populaire et les plus efficace, appelé l'optimiseur ADAM (Adaptive moment estimation), qui combine les idées de l'optimisation du momentum avec RMSProp (Kingma et Ba, 2015). Cet algorithme estime le premier moment (la moyenne) et le deuxième moment (la variance des gradients) et garde la trace d'une moyenne à décroissance exponentielle des gradients passés.

Pour estimer les moments, l'algorithme ADAM utilise la moyenne mobile exponentielle, obtenue à partir des deux gradients m_t et v_t qui sont respectivement donnés par les équations 3.2 et 3.3 :

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1).\, g_t \tag{3.2}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2).\, {g_t}^2 \tag{3.3}$$

où g_t représente le gradient du mini-lot utilisé, β_1 et $\beta_2 \in [0,1]$ sont les hyperparamètres qui gèrent les taux de décroissance exponentielle des moyennes mobiles m_t et v_t. L'hyperparamètre de décroissance du momentum β_1 est généralement initialisé à 0,9, tandis que l'hyperparamètre de décroissance de l'échelle β_2 est souvent initialisé à 0,999 (Kingma et Ba, 2015).

Puisque m_t et v_t sont des estimations des premier et deuxième moments, les propriétés dans les équations 3.4 et 3.5 doivent être vérifiées afin de s'assurer que nous disposons d'un estimateur sans biais.

$$E[m_t] = E[g_t] \tag{3.4}$$

$$E[v_t] = E[{g_t}^2] \tag{3.5}$$

Pour vérifier cette hypothèse, nous avons besoin de trouver le modèle que nous allons utiliser. Pour m = 0, 1, 2 et 3, l'équation 3.2 peut être reformulée comme suit :

$$\begin{gathered} m_0 = 0 \\ m_1 = \beta_1 m_0 + (1 - \beta_1) g_1 = (1 - \beta_1) g_1 \end{gathered} \tag{3.6}$$

$$m_2 = \beta_1 m_1 + (1-\beta_1) g_2 = \beta_1 (1-\beta_1) g_1 + (1-\beta_1) g_2$$

$$m_3 = \beta_1 m_2 + (1-\beta_1) g_3 = {\beta_1}^2 (1-\beta_1) g_1 + \beta_1 (1-\beta_1) g_2 + (1-\beta_1) g_3$$

On peut remarquer que plus la valeur de m augmente, moins les premières valeurs des gradients contribuent à la valeur globale, car elles sont multipliées par des valeurs bêta de plus en plus petites. Par conséquent, les équations 3.2 et 3.3 peuvent être reformulées respectivement selon les équations 3.7 et 3.8 :

$$m_t = (1-\beta_1) \sum_{i=1}^{t} {\beta_1}^{t-i} . g_i \tag{3.7}$$

$$v_t = (1-\beta_2) \sum_{i=1}^{t} {\beta_2}^{t-i} . {g_i}^2 \tag{3.8}$$

Maintenant nous souhaitons obtenir la valeur estimée de la moyenne $E[m_t]$ et de la variance mobile exponentielle $E[v_t]$ à un moment t, qui correspondent au premier et au second moment réels $E[g_t]$ et $E[{g_t}^2]$ respectivement, afin de pouvoir corriger l'écart entre les deux.

En prenant les espérances des côtés gauche et droit des équations précédentes, l'équation 3.4 pour le premier moment $E[m_t]$ peut-être écrit comme suit :

$$E[m_t] = E\left[(1-\beta_1) \sum_{i=0}^{t} {\beta_1}^{t-i} . g_i\right] \tag{3.9}$$

$$E[m_t] = E[g_t] . (1-\beta_1) \sum_{i=1}^{t} {\beta_1}^{t-i} + \zeta \tag{3.10}$$

$$E[m_t] = E[g_t] . (1-\beta_1) + \zeta \tag{3.11}$$

Idem pour le deuxième moment, l'équation 3.5 de l'espérance $E[v_t]$, peut-être écrit comme suit :

$$E[v_t] = E\left[(1-\beta_2)\sum_{i=1}^{t}{\beta_2}^{t-i}\,.\,{g_i}^2\right] \tag{3.12}$$

$$E[v_t] = E[{g_t}^2]\,.\,(1-\beta_2)\sum_{i=1}^{t}{\beta_2}^{t-i} + \zeta \tag{3.13}$$

$$E[v_t] = E[{g_t}^2]\,.\,(1-\beta_1) + \zeta \tag{3.14}$$

Nous pouvons maintenant approximer g_i avec g_t puisqu'elle ne dépend plus de i on peut la retirer de la somme. Comme cette approximation a lieu, l'erreur ζ apparait dans la formule. ζ= 0 dans le cas où $E[{g_t}^2]$ est stationnaire; sinon, ζ doit être considérée très petite (Kingma et Ba, 2015).

Nous devons maintenant corriger l'estimateur, afin que la valeur estimée soit celle que nous voulons. Cette étape est généralement appelée correction du biais. Les formules finales de notre estimateur seront données par les équations 3.15 et 3.16 suivantes :

$$\widehat{m}_t = \frac{m_t}{1-{\beta^t}_1} \tag{3.15}$$

$$\hat{v}_t = \frac{v_t}{1-{\beta^t}_2} \tag{3.16}$$

La seule chose qui reste à faire, est d'utiliser ces moyennes mobiles pour régler le taux d'apprentissage individuellement pour chaque paramètre. La façon dont cela est fait dans l'optimiseur ADAM est très simple, la mise à jour des poids se fait selon l'équation 3.17 :

$$\theta_t = \theta_{t-1} - \eta\frac{\widehat{m}_t}{\sqrt{\hat{v}_t} - \epsilon} \tag{3.17}$$

où $\epsilon = 10^{-8}$, et η représente le taux d'apprentissage ou parfois appelé taille du pas (step size). Cet algorithme nécessite moins de réglage des hyperparamètres. Cependant, il est très important de trouver un bon taux d'apprentissage. Si nous le fixons trop haut, l'entrainement peut diverger, alors que si nous le fixons trop bas, l'entrainement finira par

converger vers l'optimum, mais cela prendra beaucoup de temps. Nous avons essayé différentes valeurs de η variant de 0.01 à 0.000001 afin de de trouver la valeur optimale.

3.7 REGULARISATION

Les réseaux de neurones à apprentissage profonds ont généralement des centaines de milliers, voire des millions de paramètres. Cela leur apporte une énorme flexibilité et une grande capacité de modélisation, mais les rend également susceptibles de suradapter les données d'apprentissage. Dans ce projet, plusieurs techniques ont été utilisées pour éviter l'ajustement excessif, comme le fait de générer davantage de données d'apprentissage en augmentant les données, la normalisation par lots (Batch normalization), la réduction de la complexité de l'architecture et l'ajout d'un dropout dans la couche entièrement connectée. Cependant, l'amélioration de la généralisation reste un aspect relativement inconnu, et il y a très peu de techniques qui fonctionnent bien dans la pratique et qui ont des justifications théoriques bien établies.

Une fonction de perte, également appelée fonction de coût, mesure la correspondance entre les prédictions de sortie du réseau par propagation directe et les vraies étiquettes connues. Parmi les fonctions de perte couramment utilisées pour la classification multiclasses, on citera l'entropie croisée (Yamashita *et al*., 2018).

Pour la catégorisation multi-classes, nous définissons la fonction de coût, qui représente la perte moyenne sur l'ensemble d'apprentissage par l'entropie catégorique croisée, donnée par l'équation 3.18 comme suit:

$$\mathcal{L}_{ECC}(t\,,y) = -\frac{1}{N_C}\sum_{k=1}^{N_C} t_k log(y_k) \tag{3.18}$$

où N_C désigne le nombre de classes différentes, k l'indice du vecteur, t_k le vecteur des étiquettes des vraies classes et y_k représente le vecteur des scores obtenus par la fonction softmax pour chaque classe. Il est intéressant de savoir que plus l'entropie croisée est petite, plus les deux distributions de probabilité sont similaires, meilleure est la classification.

Pour une classe donnée k, la fonction pour calculer le score pour chaque entrée z de la couche de sortie, désignée par σ (softmax) est décrite par l'équation 3.19 suivants:

$$y_k = \sigma(z_1, \dots, z_C)_k = \frac{e^{z_k}}{\sum_{k=1}^{C} e^{z_k}} \tag{3.19}$$

Lorsque nous combinons la fonction d'activation softmax avec la fonction de perte d'entropie croisée, nous obtenons la régression softmax. Ce modèle peut être exprimé comme suit :

$$z = Wx + b \tag{3.20}$$

$$y = softmax(z) \tag{3.21}$$

$$\mathcal{L}_{ECC}(t\ , y) = -t^T log(y_k) \tag{3.22}$$

où W représente la matrice de poids, b le vecteur de biais, x le vecteur d'entrée du réseau de neurones convolutif et z le vecteur de sortie du réseau de neurones convolutif .

Dans ce projet, nous voulons ajouter un autre terme, appelé terme de régularisation, qui pousse le réseau à maintenir les valeurs de ses paramètres à un niveau bas et donc à une distribution plus homogène. Cela empêche le réseau de développer de paramètres dont les valeurs sont suffisamment importantes pour influencer sa prédiction (Planche et Andres, 2019).

Par conséquent, l'équation 3.18 peut-être exprimée par la formule 3.23 suivante :

$$\mathcal{L}_{ECC}(y, t) = -\frac{1}{N_C}\sum_{k=1}^{N_C} t_k log(y_k) + \mathcal{R}_{L2}(\omega) \tag{3.23}$$

où $\mathcal{R}(\omega)$ représente le terme de la régularisation L2. Il est défini par l'équation 2.24 ci-dessous :

$$\mathcal{R}_{L2}(\omega) = \frac{\lambda}{2}\sum_{i=1}^{D} \omega_i^2 \tag{3.24}$$

où D représente la taille du vecteur d'entrée et ω_i représente la valeur du poids et l'hyperparamètre λ est parfois appelé le coût du poids. C'est un facteur permettant de contrôler la valeur de la régularisation afin de réduire l'amplitude du terme de régularisation par rapport à la perte moyenne (Géron, 2019). La valeur utilisée de λ dans notre système de classification est de 0.010.

CHAPITRE 4
EXPÉRIMENTATION ET RÉSULTATS

Les méthodes de représentation temps-fréquence (RTF) et les modélisations, exposées précédemment, ont été appliquées aux signaux respiratoires en vue de classification multiclasses. Le présent chapitre décrit les protocoles d'expérimentation et les critères d'évaluation utilisés dans notre projet. Il présente également les résultats de nos différentes expérimentations.

Les travaux réalisés dans ce projet ont été implémentés avec Python sous Keras et Tensorflow sous Jupiter Nootbook ainsi que l'environnement Notebook de Google Colaboratory en utilisant un ordinateur avec Intel Core™ i5-10300H Processor@2.5 GHz à 4 cœurs, un GPU NVIDIA GeForce GTX1650 Ti GPU et 16 GO de RAM.

4.1 PRÉPARATION ET CRITÈRE D'ÉVALUATION DE LA BASE DE DONNÉES

Avant d'entamer la comparaison des techniques utilisées, nous commençons par définir certains critères d'évaluation. Nous répartissons les enregistrements audio en fonction de leurs annotations sur quatre classes de cycles respiratoires: normaux, crépitants, sibilants et à la fois crépitants et sibilants, en conservant la même répartition de la base des données établie dans (Rocha *et al*., 2019), avec 40 % pour le test et 60 % pour l'entrainement. Nous utilisons des critères d'évaluation spécifiques exigés dans le cadre de la compétition ICBHI pour évaluer les performances de classification, Les paramètres utilisés dans les critères d'évaluation sont présentés dans le tableau 4.1. Par exemple C_C est le nombre des signaux crépitants prédits crépitants, alors que C_W est le nombre de signaux crépitants prédits sibilants.

Tableau 4.1. Matrice de confusion pour la classification des sons respiratoires

		Classes prédites			
		Normal	Crépitant	Sibilant	Crépitant & Sibilant
Classes vraies	Normal (N)	$\mathbf{N_N}$	$\mathbf{N_C}$	$\mathbf{N_W}$	$\mathbf{N_B}$
	Crépitant (C)	$\mathbf{C_N}$	$\mathbf{C_C}$	$\mathbf{C_W}$	$\mathbf{C_B}$
	Sibilant (W)	$\mathbf{W_N}$	$\mathbf{W_C}$	$\mathbf{W_W}$	$\mathbf{W_B}$
	Crépitant & Sibilant (B)	$\mathbf{B_N}$	$\mathbf{B_C}$	$\mathbf{B_W}$	$\mathbf{B_B}$

Pour chaque expérience, nous avons utilisé la méthode d'évaluation des performances proposé par l'ICBHI, afin de comparer nos résultats avec ceux qui ont été publiés dans la littérature. Nous définissons le score moyen appelé $ICBHI_{score}$, la sensibilité, la spécificité et la justesse selon les équations suivantes :

$$Sensibilité = (C_C + W_W + B_B)/(C + W + B) \tag{4.1}$$

$$Spécificité = N_N/N \tag{4.2}$$

$$Justesse = (N_N + C_C + W_W + B_B)/(N + C + W + B) \tag{4.3}$$

$$ICBHI_{score} = (Sensibilité + Spécificité)/2 \tag{4.4}$$

où C, W, B et N désignent les nombres de cycles de crépitants, sibilants, crépitants et sibilants et normaux, respectivement, tandis que C_C, W_W, B_B et N_N représentent uniquement les prédictions positives de chaque classe. L'équation 4.3 représente l'exactitude (accuracy) de la classification. Ce critère sera utilisé pour la sélection initiale des modèles.

4.2 RÉÉCHANTILLONNAGE DES SONS RESPIRATOIRES

Les enregistrements audio présentés dans la base de données ICBHI ont été réalisés avec des fréquences d'échantillonnage différentes: 4 kHz, 10 kHz et 44.1 kHz (figure 4.1). Nous avons donc rééchantillonné tous les enregistrements audio à la même fréquence

d'échantillonnage de 22 kHz afin d'en assurer la cohérence entre les différentes images générées par les spectrogrammes. Ce choix a été fait après avoir expérimenté différentes fréquences d'échantillonnage. En même temps, nous avons pu préserver la qualité requise tout en garantissant un moindre temps de calcul que lorsque nous travaillons avec les données échantillonnées à 44,1 kHz.

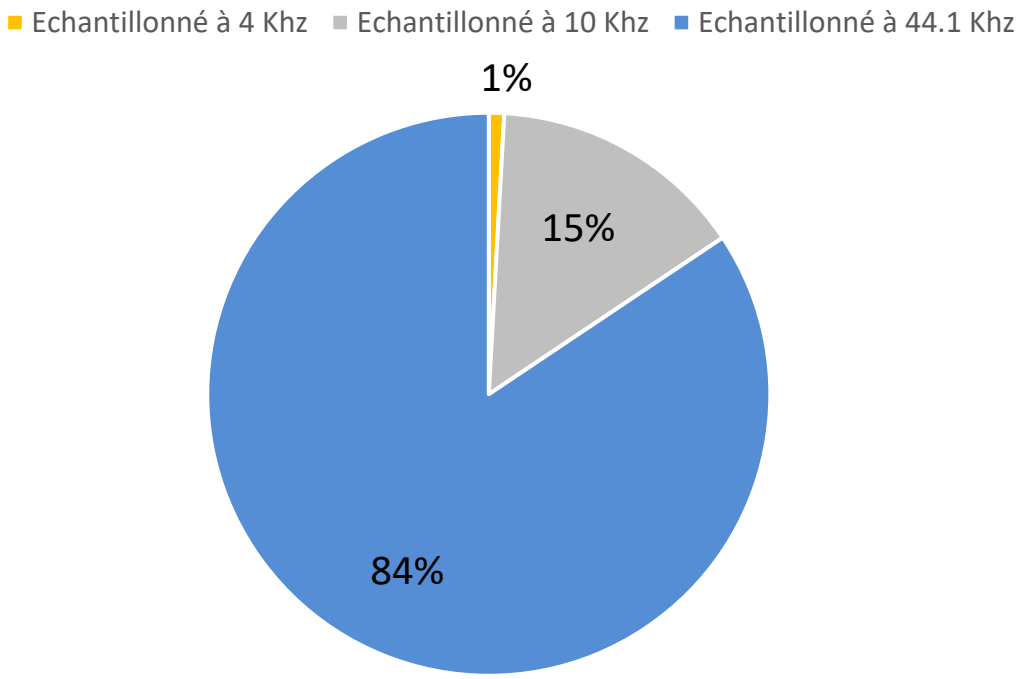


Figure 4.1. Diagramme à secteur des fréquences d'échantillonnage utilisées au niveau des enregistrements de sons respiratoires selon le nombre de fichiers dans la base de données.

4.3 SEGMENTATION ET DÉCOUPAGE DES CYCLES

L'analyse de la base de données ICBHI démontre que 65 % des cycles ont une durée inférieure à 3 secondes, tandis que 98 % des cycles dans la base de données ont une durée de cycle inférieure à 6 secondes. Un aspect délicat des réseaux de neurones convolutifs (CNN) est que, pendant le processus d'entrainement, les échantillons du mini lot (mini batch size) utilisés pour entrainer le modèle doivent avoir un tenseur statique. Comme la

durée d'un cycle respiratoire est variable, une certaine forme de rembourrage ou de masquage serait nécessaire. Par conséquent, pour garantir une dimension constante des tenseurs, nous découpons les cycles en segments de longueur fixe de 6 secondes ou en plusieurs segments de longueur fixe de 6 secondes chacun. Si la longueur du cycle est inférieure à celle du segment, le remplissage par zéros (bourrage de zéros) ou le remplissage par réflexion sont les deux méthodes proposées pour compléter les segments partiellement remplis. Ce processus est présenté à la figure 4.2.

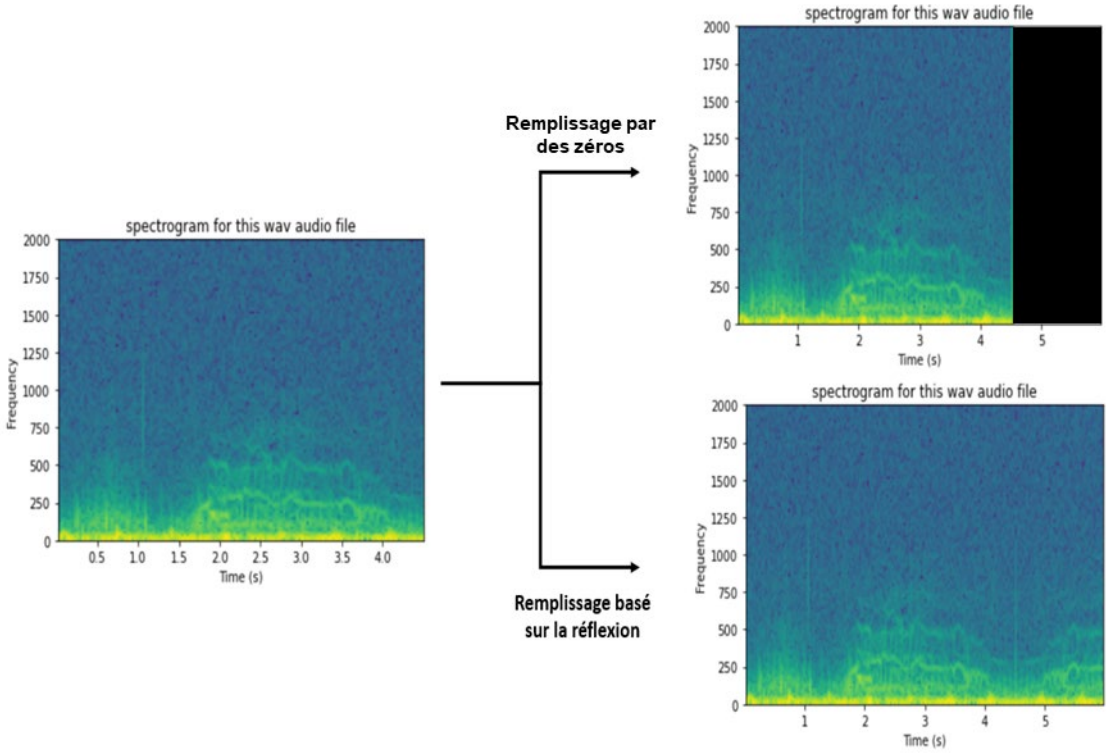


Figure 4.2. Les spectrogrammes obtenus après l'application des deux méthodes de remplissage proposées.

À titre d'exemple, un cycle de 6 secondes formera un seul segment. Un cycle dont la durée est comprise entre 6 et 12 secondes sera segmenté en deux, tandis qu'un cycle dont la durée est supérieure à 12 secondes, il sera décomposé en 3 segments. Cette segmentation est présentée en détail dans le tableau 4.2 et 4.3. La longueur du cycle respiratoire représente un défi énorme pour cette tâche de classification, c'est la raison pour laquelle, dans la section suivante, nous allons expérimenter différentes longueurs de segments afin

de définir le meilleur choix de longueur de segment. Le nombre de segments dans chaque enregistrement et sa durée en secondes sont présentés dans le tableau 4.4.

Tableau 4.2. Découpage et segmentation des cycles de la base de données ICBHI

Plages de durée des cycles respiratoires	**Nombre de cycles original**	**Opération de segmentation « Oui » si nécessaire**	**Total des segments résultantes**
Inférieurs à 6 s	6779	Non	6779
6s < cycle <12 s	118	Oui	236
Supérieurs à 12 s	1	Oui	3

Tableau 4.3. Distribution des segments.

Classes	**Nombre de cycles original**	**Nombre de segments résultantes**	**Nombre de segments ajoutés**
Normal	3642	3734	92
Crépitant	1864	1878	14
Sibilant	886	894	8
Crépitant & Sibilant	506	512	6
Total	6898	7018	120

Tableau 4.4. La base de données utilisée dans cette étude

Classes	**Normal**	**Crépitants**	**Sibilants**	**Crépis. Et Sibs.**	**Total**
Nombre de cycles originaux	3642	1864	886	506	6898
Nombre de segments résultants	3734	1878	894	512	7018
Segments créés par le processus de segmentation	92	14	8	6	120
Échantillons pour l'entrainement (60%)	2245	1127	535	307	4214
Échantillons d'essai (40%)	1489	751	359	205	2804

La segmentation des cycles respiratoires a été utilisée pour la classification des anomalies, c-à-d, la présence ou l'absence des sibilants et des crépitants (tableau 4.4). Dans la plupart des études, les auteurs ont proposé d'utiliser différentes longueurs de segments fixes (par exemple 3, 4, 5, 6, 7 secondes) à l'aide des techniques de bourrage différentes afin de maintenir la même durée pour tous les segments. (Perna et Tagarelli, 2019; Liu *et*

al., 2019; Pham *et al*., 2020; Gairola *et al*., 2020). Cependant, pour la prédiction des pathologies, ou chaque enregistrement identifie le statut du patient en termes de santé ou de présence de l'une des maladies ou pathologies respiratoires suivantes: Broncho pneumopathie chronique obstructive (BPCO), Bronchiectasie, Asthme, Infection des voies respiratoires supérieures et inférieures, Pneumonie et Bronchiolite. La plupart des travaux proposés utilisent la totalité des enregistrements audio des sons respiratoires.

4.4 MANIPULATION DES DONNÉES

4.4.1 Normalisation des données

La normalisation Min-Max est une technique courante qui consiste à remettre à l'échelle les données d'une plage originale dans une autre afin que toutes les valeurs soient comprises entre 0 et 1. L'obtention de l'homogénéité des dispositifs est cruciale lorsque l'on travaille avec des algorithmes d'apprentissage automatique, car des variables d'entrée non normalisées peuvent entrainer un processus d'apprentissage lent et instable. La normalisation min-max est définie par l'équation 3.3:

$$X_i' = \frac{x_i - \min(x)}{\max(x) - \min(x)} (1 - 0) \tag{3.3}$$

où $\min(x)$ et $\max(x)$ sont les valeurs minimale et maximale de la variable d'entrée x_i et X_i' représente la version normalisée de l'attribut d'entrée x_i.

4.4.2 Augmentation des données

Dans le domaine de l'apprentissage profond, la performance d'un modèle s'améliore souvent avec la quantité de données disponibles pour son entrainement. L'augmentation des données est une suite de techniques qui améliorent la taille et la qualité des ensembles de données d'entrainement afin de construire de meilleurs modèles d'apprentissage profond. Malheureusement, de nombreux domaines d'application tels que l'analyse d'images médicales n'ont pas accès aux données massives (big data). De plus, ils souffrent d'un déséquilibre de classe dans les bases de données disponibles, ce qui est le cas avec

cette base de données ICBHI. Cette approche était proposée précédemment dans des architectures populaires comme le cas d'AlexNet (Krizhevsky *et al*., 2012), VGG-16 (Simonyan et Zisserman, 2014) et GoogleNet (Szegedy *et al*., 2014), pour augmenter la généralisation et la capacité à s'adapter correctement à de nouvelles données, jamais vues auparavant. Dans ce projet, nous avons utilisé trois techniques d'augmentation de données appliquées directement aux représentations temps-fréquence résultantes.

4.4.2.1 Perturbation de la longueur du conduit vocal (VTLP)

La méthode de perturbation de la longueur du conduit vocal (VTLP) est représentative d'un groupe de mécanismes d'augmentation des données qui génèrent de nouveaux échantillons en perturbant ou en déformant les spectres des données d'entrainement existants. Par conséquent, à chaque fréquence f, une nouvelle fréquence f'est générée à l'aide de la formule suivante :

$$f' = \begin{cases} f_\alpha & f \leq f_{high}\ \dfrac{\min(\alpha, 1)}{\alpha} \\ \dfrac{f_s}{2} - \dfrac{\frac{f_s}{2} - f_{high}\min(\alpha, 1)}{\frac{f_s}{2} - f_{high}\ \frac{\min(\alpha, 1)}{\alpha}} \left(\dfrac{f_s}{2} - f\right) & sinon \end{cases} \tag{4.5}$$

où f_s représente la fréquence d'échantillonnage, pour chaque enregistrement de l'ensemble d'entrainement, un facteur de déformation α est choisi aléatoirement parmi [0.9, 1.1] pour déformer l'axe des fréquences et fixer la largeur de bande maximale du signal à f_{high}= [3200, 3800] (Jaitly et Hinton, 2013). La figure 4.3 présente les résultats de l'application de cette technique sur la RTF d'un cycle respiratoire sibilant.

4.4.2.2 Étirement temporel

Dans l'étirement temporel, la durée du signal audio est étendue en changeant la fréquence d'échantillonnage, tandis que son contenu fréquentiel reste inchangé. Dans notre projet, nous avons utilisé un taux d'échantillonnage aléatoire uniformément distribué avec ±10% du taux original (Nguyen et Pernkopf, 2020; Chanane et Bahoura, 2021).

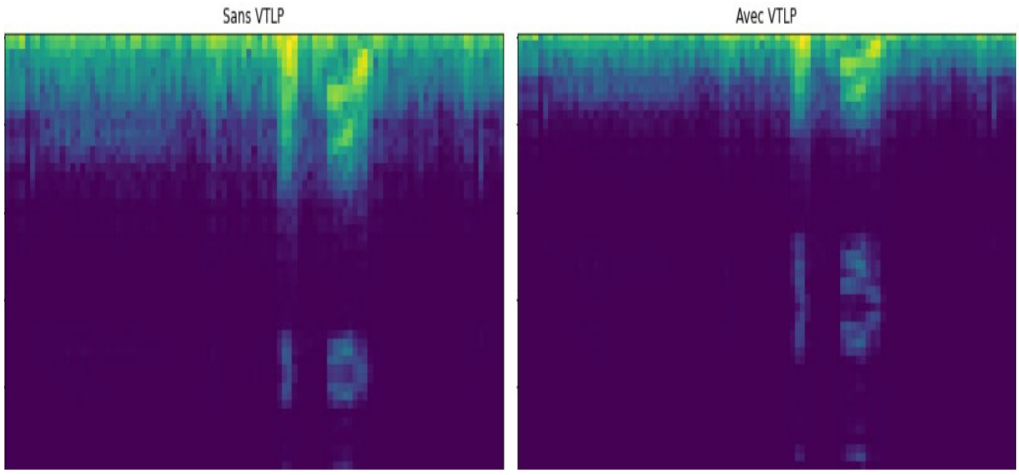


Figure 4.3. Représentation avant et après l'utilisation de la technique VTLP sur le spectrogramme.

4.4.2.3 Rotations aléatoires du spectrogramme

Rotations aléatoires du spectrogramme : Un renversement d'image se produit lorsque l'on inverse les rangées ou les colonnes de pixels dans le cas d'un renversement vertical ou horizontal, respectivement. Dans notre étude, nous avons utilisé le retournement horizontal pour étendre les échantillons d'entrainement (Chanane et Bahoura, 2021).

4.5 INFLUENCE DES DIFFÉRENTS PARAMÈTRES

Pour chacune des expériences suivantes, nous allons utiliser l'architecture que nous avons proposée pour évaluer l'effet des différents paramètres. Vu le nombre important de combinaisons et de paramètres sous-jacents, nous procèderons par élimination. En effet, tout au long de l'expérience, nous allons éliminer les techniques qui donnent de mauvais résultats et conserver uniquement celles qui semblent prometteuses. À la fin nous comparerons notre architecture avec les architectures de référence proposées dans la littérature, notamment AlexNet et VGG-net.

4.5.1 Influence de la longueur des segments

Il est intéressant de considérer comment la longueur du segment du signal respiratoire analysé influence la justesse de la classification. En tenant compte du fait que notre

architecture utilise la technique du remplissage par zéros pour rendre les cycles courts plus longs qu'un minimum donné, nous avons donc ajusté la longueur du segment du signal respiratoire analysé dans notre modèle de base de 2 à 10 secondes. Les graphiques à barres ci-dessous montrent les résultats de la classification pour différentes longueurs de segments générées par le spectrogramme de Mel. La figure 4.4 nous permet de constater que les segments de 6 secondes ont obtenu la meilleure performance de classification pendant la phase d'entrainement et de test respectivement de 96.34 % et 73.62 %. En outre, les segments de 5 et 7 secondes ont également obtenu de bons résultats en termes de justesse et de perte durant la phase des tests. Cependant, les petits segments tels que 2, 3 et 4 secondes ont obtenu les scores globaux les plus bas parmi tous les segments. Le modèle basé sur la segmentation à 9 secondes a obtenu une meilleure justesse particulièrement pendant l'entrainement 96.34 %, mais il est toujours moins efficace que le modèle qui utilise la longueur des segments de 6 secondes quand on considère la phase de validation.

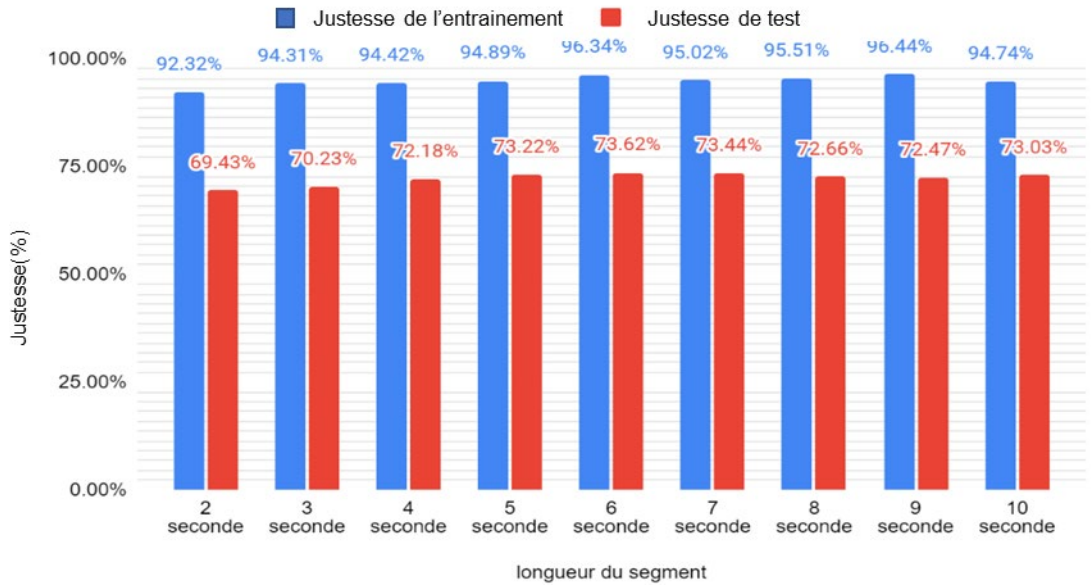


Figure 4.4. Comparaison des performances du modèle basé sur les spectrogrammes de Mel en utilisant différentes longueurs de segment.

Pour les écarts entre le modèle le plus performant et le moins performant, en fonction de la longueur du segment choisi, nous avons opté pour une courbe d'apprentissage qui est une représentation graphique de la relation entre la justesse du modèle et le nombre

d’itérations. Ces résultats sont obtenus en utilisant l’outil de TensorBoard pour le traçage simultané des courbes. Ensuite nous avons extraie ces données sous forme des fichiers CSV pour les afficher sous Excel. La figure 4.5 présente les courbes d'apprentissage du modèle basé sur des segments de 6 secondes et du modèle basé sur des segments de 2 secondes.

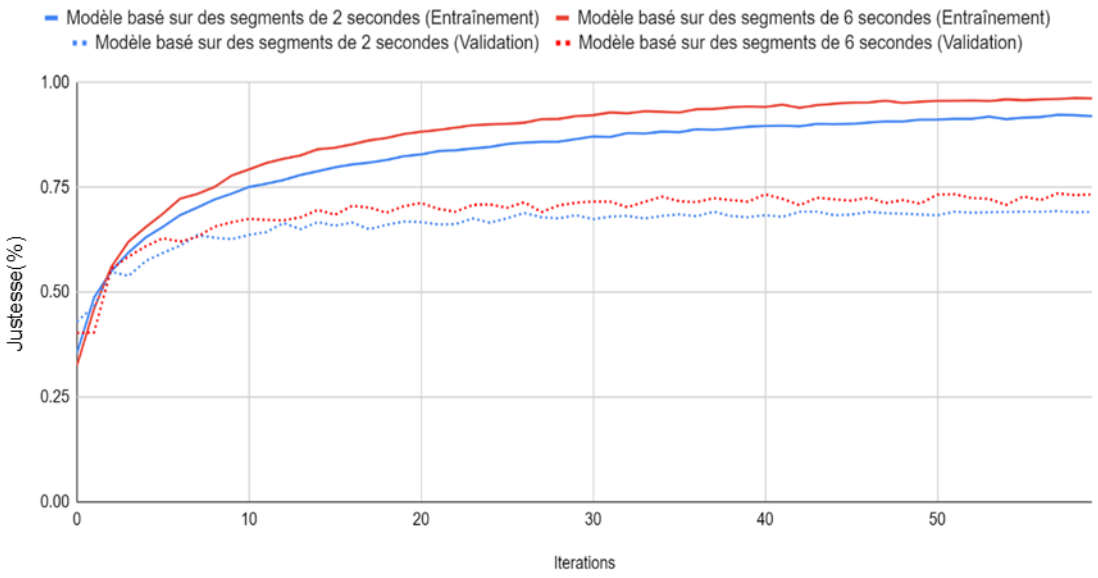


Figure 4.5. Courbes d'apprentissage de la justesse obtenues à partir des simulations des deux modèles.

4.5.2 Influence de la représentation temps-fréquence

Dans cette étude, nous examinons les performances de six représentations temps-fréquence différentes. Nous pouvons diviser cette étude en deux parties. En premier temps, nous utilisons des spectrogrammes à base de STFT, CQT et Mel, tandis que dans la deuxième étude, nous utilisons la décomposition en modes empiriques pour obtenir les six différentes fonctions intrinsèques (IMFs), ce qui nous permet de combiner chaque IMF individuellement avec les spectrogrammes de la première étude, ce qui aboutit à 18 combinaisons possibles. Les six meilleurs résultats de cette étude sont présentés en ordre décroissant dans la figure 4.6.

Les résultats obtenus par cette approche hybride n’ont pas réussi à dépasser ceux de l’approche de base. Nous observons une diminution remarquable de la justesse lorsque

nous utilisons le spectrogramme STFT au lieu du spectrogramme Mel. Cela peut s'expliquer par l'impact de la résolution en amplitude et en fréquence. En termes de résolution d'amplitude, les oscillations d'amplitude minimale ne peuvent être extraites, car les extrêmes de ces oscillations de faible amplitude ne peuvent être détectés.

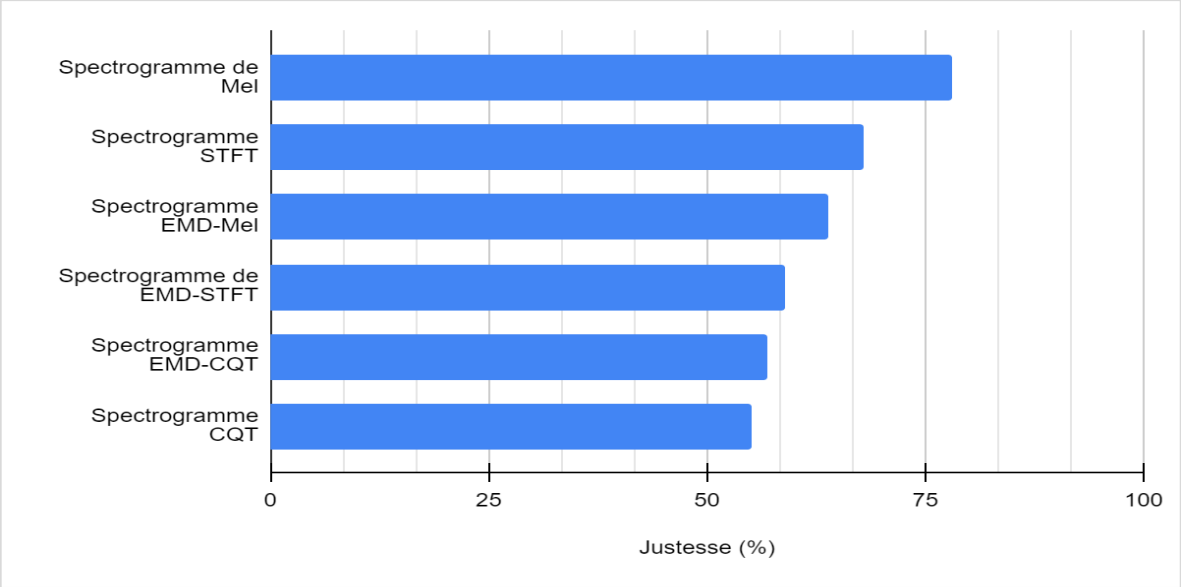


Figure 4.6. Résultats comparatifs des méthodes de représentation temps-fréquence. Les valeurs de justesse affichées sont obtenues pendant la phase de tests.

Selon l'étude que nous avons menée lors de notre analyse de la revue de littérature. Reichert *et al*. (2008) ont montré que les sons pulmonaires (captés sur le thorax ou sur le dos) ont une amplitude plus faible que les sons trachéaux. Cependant, dans cet ensemble de données en particulier, les sons trachéaux constituent 0.07 % (Gairola *et al*., 2020), ce qui complique la reconnaissance des oscillations de faible variation par l'algorithme.

En termes de résolution fréquentielle, la décomposition EMD se comporte comme un banc de filtres dyadiques (Huang *et al*., 1998). Ces filtres se chevauchent et le nombre d'extrêmes est réduit de moitié d'une composante IMF à l'autre. Le problème se pose lorsque l'on traite des signaux à composantes multiples, qui peuvent partager la même fréquence au même moment. En effet, à la fin du processus de filtrage, ces signaux seront représentés par la même composante IMF, et les fréquences ne pourront pas être résolues.

Cependant, cette méthode n'est pas encore aussi efficace lorsqu'il s'agit de traiter les sons respiratoires.

4.5.3 Influence des différentes composantes des IMF

Dans cette étude, nous illustrerons l'influence du choix de la composante IMF (fonctions de mode intrinsèque) sur les performances finales. L'expérience précédente a montré que la combinaison de la décomposition en mode empirique (EMD) et spectrogramme de Mel donne de meilleurs résultats que les autres combinaisons (figure 4.6). Cependant, dans cette expérience, nous allons montrer comment chaque composant spécifique de l'IMF se comporterait en utilisant la même démarche présentée dans la figure 2.11. Les résultats de cette étude sont présentés dans la figure 4.7.

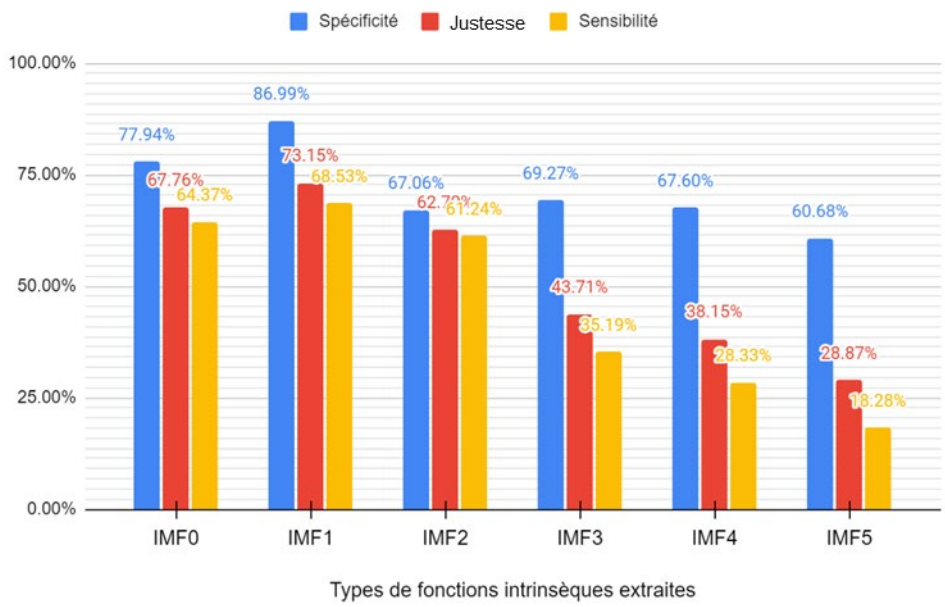


Figure 4.7. Résultats obtenus durant la phase de test de la sensibilité, spécificité et la justesse pour les différentes composantes des IMF.

Les IMF sont obtenues par ordre décroissant de fréquence, où l'IMF0 comprend les composantes de fréquence les plus élevées dans les sons respiratoires. Par conséquent, pour chaque segment, les IMF0 à IMF5 sont calculées afin de couvrir la gamme de fréquences utile pour l'analyse des sons respiratoires. Il a été montré que l'IMF1 obtient la meilleure

justesse. En effet, sur les 6 IMFs, toutes les composantes fréquentielles de la IMF1 couvrent la gamme de fréquences des sons adventices tels que les sibilants et les crépitants comme montre la figure 2.12

4.5.4 Influence des hyperparamètres du CNN

4.5.4.1 Influence de la taille du lot (Batch size)

Dans cette expérience, nous allons comparer différentes tailles de lots (batch size), car ce paramètre a un impact significatif sur les performances du modèle et le temps d'apprentissage. Le principal avantage de l'utilisation d'une grande taille de lot est que les accélérateurs matériels tels que les GPU peuvent les traiter efficacement, ce qui permet à l'algorithme d'apprentissage de voir plus d'exemples par seconde. Cependant, certains chercheurs prennent une autre direction et recommandent d'utiliser des tailles de lots plus petites, mais pas trop petites afin d'obtenir une meilleure généralisation tout en évitant les instabilités d'entrainement (Géron, 2019).

La figure 4.8 présente les résultats préliminaires que nous avons obtenus. Les performances des différentes combinaisons sont représentées par les barres verticales indiquant la justesse pour différente taille du lot dans la phase d'entrainement et de test. Les figure 4.9 et 4.10 présentent les courbes d'apprentissage pour différentes combinaisons selon la taille du lot. Une courbe d'apprentissage d'un modèle bien ajusté présente une perte de validation élevée au début, qui diminue progressivement lors de l'ajout d'exemples d'entrainement et s'aplatit graduellement, ce qui indique que l'ajout de plus d'exemples d'entrainement n'améliore pas les performances du modèle sur les données non observées. La figure 4.10 présente les pertes de validation qui diminue au début, mais elles commencent à diverger à partir de la 10^{ieme} itération. Afin de choisir le meilleur modèle, il faut savoir qu'un bon modèle est celui qui présente une faible valeur de perte. Par conséquent, le modèle avec une taille de lot de 256 présente une faible perte de validation.

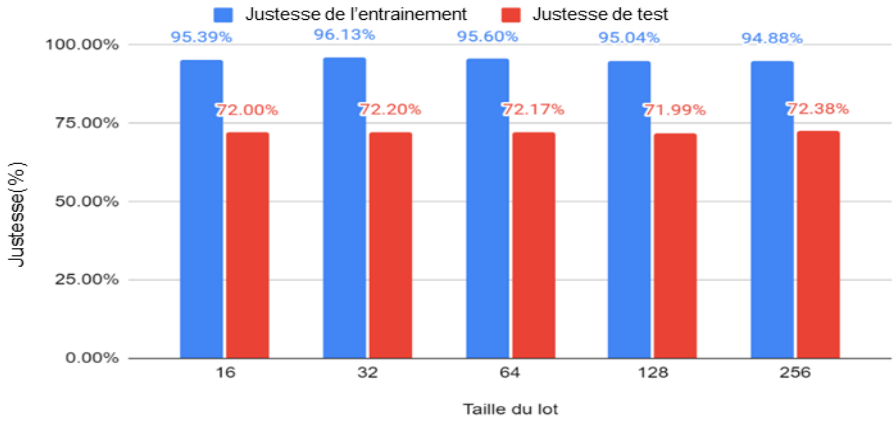


Figure 4.8. Effet de la taille du lot sur la performance (justesse) du système de classification.

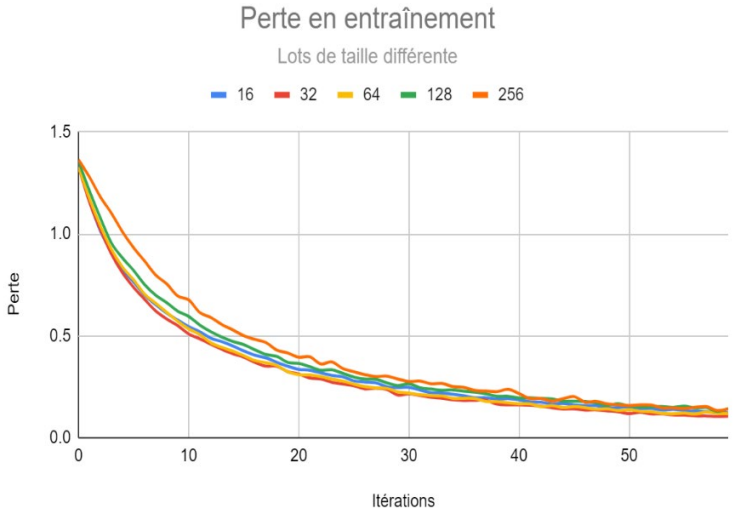


Figure 4.9. Perte d'entrainement testée pour le modèle CNN en fonction de la taille des lots.

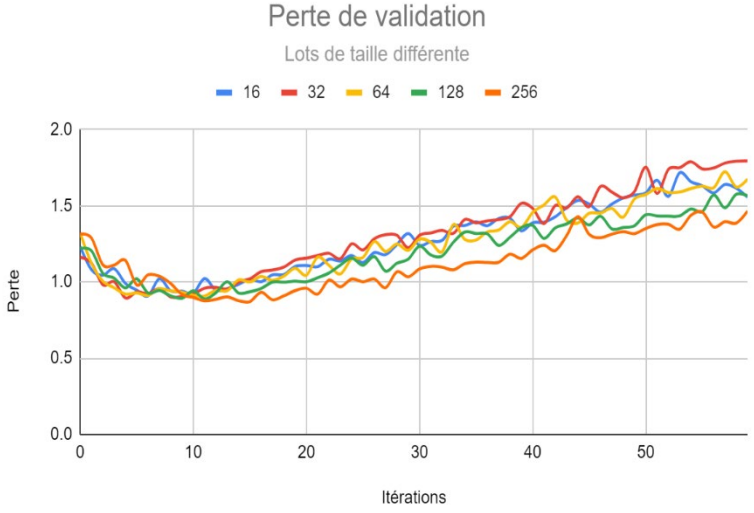


Figure 4.10. Perte de validation testée pour le modèle CNN en fonction de la taille des lots.

4.5.4.2 Influence de la régularisation sur les erreurs de généralisation (Overfitting)

L'augmentation de l'erreur de généralisation peut être mesurée par la performance du modèle sur l'ensemble des données de test. Précédemment, nous avons remarqué que la perte pendant la phase de test commence à diverger après quelques itérations. Afin de remédier à ce problème, nous avons intégré trois techniques, soit la régularisation L1/L2, la normalisation par lots et le dropout qui vont servir à diminuer cette erreur de généralisation. Pour illustrer l'efficacité de ces techniques sur les performances de systèmes de classification, la figure 4.11 présente les résultats de cette étude en termes de perte. À partir de ces courbes, nous pouvons remarquer que l'ajout de régulateurs a réussi à réduire l'overfitting. Le modèle de base commence à diverger après la 20^{ieme} itération, ce qui signifie qu'il commence à suradapter l'ensemble d'apprentissages alors que le modèle optimisé continue de converger malgré l'utilisation d'un nombre élevé d'itérations.

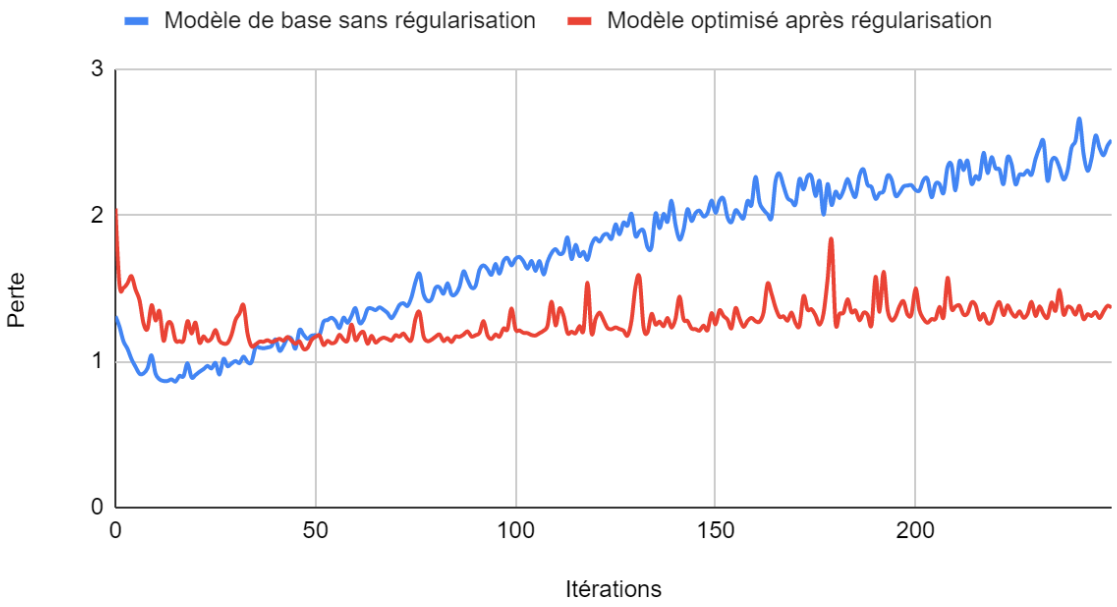


Figure 4.11. Comparaison de la fonction de perte durant la phase de test avant et après l'ajout de régulateurs.

4.5.4.3 Influence de l'augmentation des données sur les performances

Notre système n'est capable d'apprendre que lorsque les données d'apprentissage contiennent suffisamment de caractéristiques pertinentes. Les performances des réseaux neuronaux convolutifs sont fortement influencées par la disponibilité de données massives pour minimiser le surajustement des modèles et augmenter la justesse de la classification. Ce phénomène se produit lorsqu'un réseau apprend une fonction à très haute variance pour modéliser les données d'apprentissage (Shorten et Khoshgoftaar, 2019). Malheureusement, de nombreux domaines d'application n'ont pas accès au big data, comme l'analyse d'images médicales. Nous avons implémenté trois techniques d'augmentation de données pour remédier le déséquilibre présenté dans les classes de la base de données utilisée.

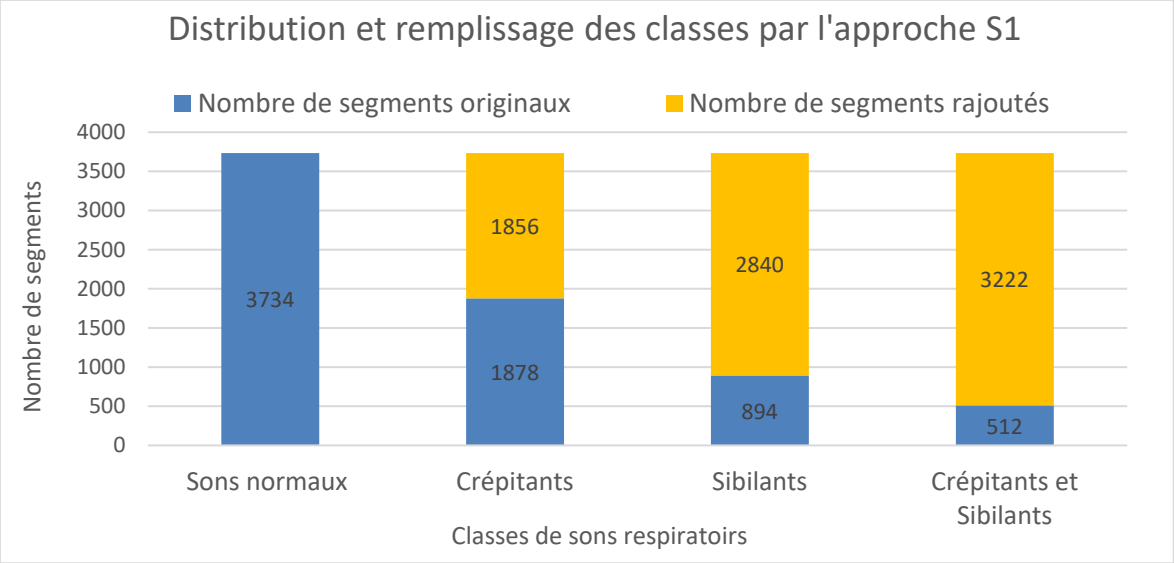


Figure 4.12. Première approche pour l'augmentation des données.

Ces méthodes sont proposées pour ajouter ou supprimer des exemples de l'ensemble de données d'entrainement pour modifier la distribution des classes. Notre première approche (S1) consiste à créer de nouveaux exemplaires grâce à l'augmentation des données, puis les ajouter aux classes sous-représentées pour qu'elles correspondent à la taille de la classe la plus significative, à savoir les sons normaux (voir figure 4.12). Cette opération est appelée suréchantillonnage. La deuxième approche (S2), utilise le chemin inverse. Nous créons des classes équilibrées en supprimant les instances de la classe

surreprésentée (sous-échantillonnage). Cependant, la troisième approche (S3), utilise une combinaison de suréchantillonnage et de souséchantillonnage (Chawla *et al*., 2002). Tout d'abord, nous supprimons les instances de la classe surreprésentée (suréchantillonnage), puis nous appliquons l'augmentation des données (suréchantillonnage) au reste des classes pour éviter un sous-apprentissage. Les deux types d'augmentation peuvent être efficaces lorsqu'ils sont utilisés isolément, mais avec les travaux de Chawla et *al*. (2002), il a été prouvé qu'il serait plus efficace d'utiliser les deux types de méthodes ensemble. Le tableau 4.5 ci-dessous, présente les résultats des segments obtenus après les augmentations de données effectuées par les différentes approches. Nous rappelons que nous avons conservé la même répartition de la base des données établie dans les travaux de Rocha *et al*. (2019), avec 40% pour le test et 60% pour l'entrainement.

Tableau 4.5. Paramètres de configuration de notre modèle.

	Sons normaux	**Crépitants**	**Sibilants**	**Crépitants et Sibilants**	**Total des segments utilisés**
Segments originaux avant augmentation	3734	1878	894	512	7018
Suréchantillonnage (S1)	3734	3734	3734	3734	14936
Souséchantillonnage aléatoire (S2)	512	512	512	512	2048
Souséchantillonnage suivi de suréchantillonnage (S3)	2562	2562	2562	2562	10248

La figure 4.13 présente les performances de justesse et de perte durant la phase de test pour les trois approches. Tout d'abord, la courbe en bleu représente la première approche (S1) où nous avons essayé d'équilibrer les échantillons de chaque classe en utilisant l'augmentation des données. La courbe rouge (S2) représente la deuxième approche c-à-d, sans augmentation des données, puisque nous avons sous-échantillonner les classes surreprésentées. La courbe jaune (S3) représente la troisième approche avec l'augmentation des données de l'ensemble d'entrainement en appliquant le retournement d'image, le VTLP et l'étirement temporel.

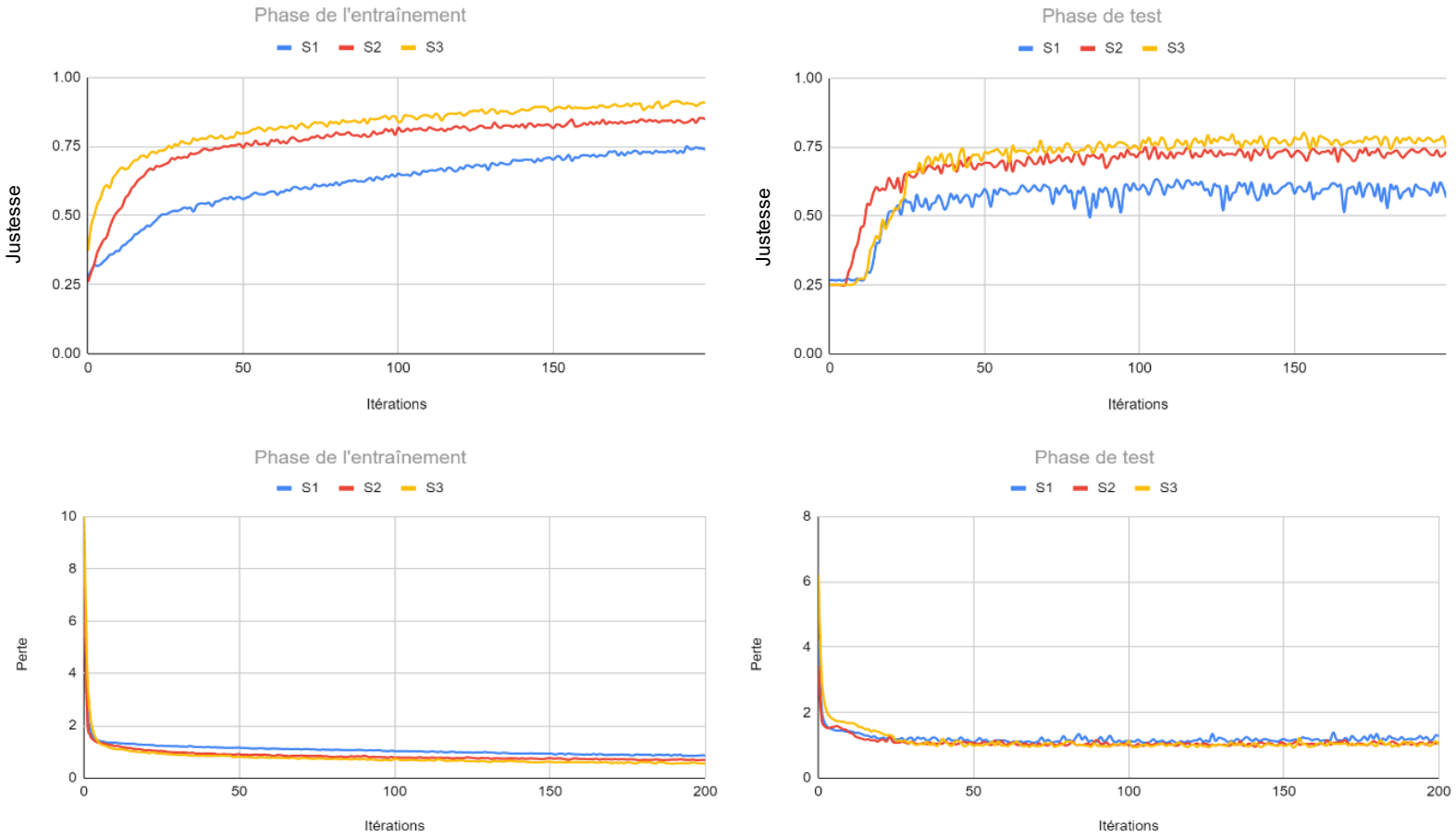


Figure 4.13. Courbes d'apprentissage établies à partir de différents schémas d'augmentation des données. La première approche est présentée dans (S1), la deuxième approche est présentée dans (S2) et la troisième approche qui a obtenu les meilleurs résultats dans (S3).

Nous pouvons rajouter que notre proposition avec les techniques de sous-échantillonnage et de suréchantillonnage démontre que l'ajout de nouveaux spectrogrammes synthétiques de Mel permet une très bonne classification pour toutes les classes. Afin de vérifier les résultats présentés par la troisième approche, nous effectuons par la suite une série vérification avec la méthode de validation croisée sur toute la base de données permettent de prouver la robustesse de cette approche.

4.6 INFLUENCE DU TAUX D'APPRENTISSAGE

Le taux d'apprentissage est sans doute l'hyperparamètre le plus important. La plupart des spécialistes considèrent la valeur η de 0.001, comme une valeur standard pour le taux d'apprentissage. Dans cette étude, nous avons expérimenté différents taux d'apprentissage allant de 0.01 jusqu'à 0.000001. La figure 4.14 montre comment l'utilisation de différents taux d'apprentissage affecte la fonction de perte pendant la phase d'entrainement de notre modèle.

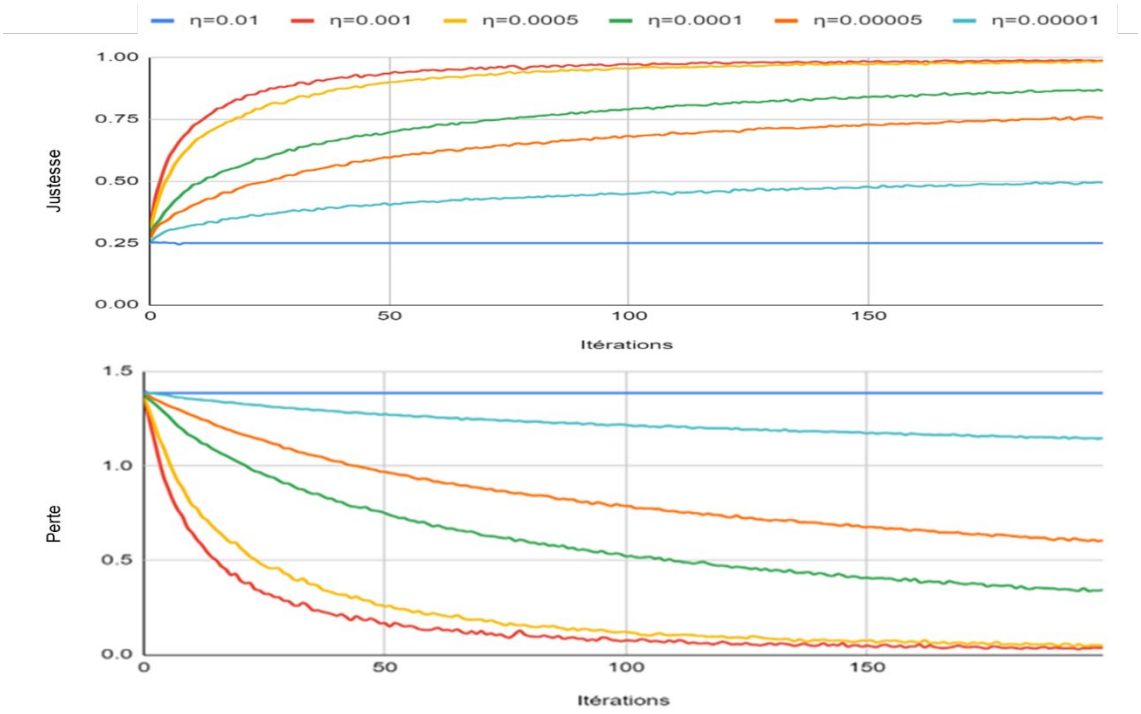


Figure 4.14. Courbes d'apprentissage d'entrainement (Justesse/Perte) pour différentes valeurs de η.

Effectivement, lorsque nous utilisons une faible valeur de η, l'entrainement finira par converger vers l'optimum, mais cela prendra beaucoup de temps. À titre d'exemple, dans le cas de η=0.00001 l'entrainement a pris relativement plus qu'une heure pour finir, tandis qu'un taux d'apprentissage trop élevé η= 0.01 peut causer un dépassement des minima locaux. Donc il pourrait ne jamais converger vers la valeur optimale.

Afin de choisir la valeur optimale du taux d'apprentissage, nous allons nous baser sur les graphes obtenus lors de la phase de test pour les modelés retenus (η=0.001, η=0.0005, η=0.0001 et η=0.00005). Dans ce scénario, nous sommes contraints de faire un compromis entre la justesse de la classification et la perte minimale que nous pouvons atteindre. La figure 4.15 présente les résultats lors de la phase de tests. Pour ce système de classification nous choisissons la valeur de η= 0.0001 comme valeur optimale du taux d'apprentissage.

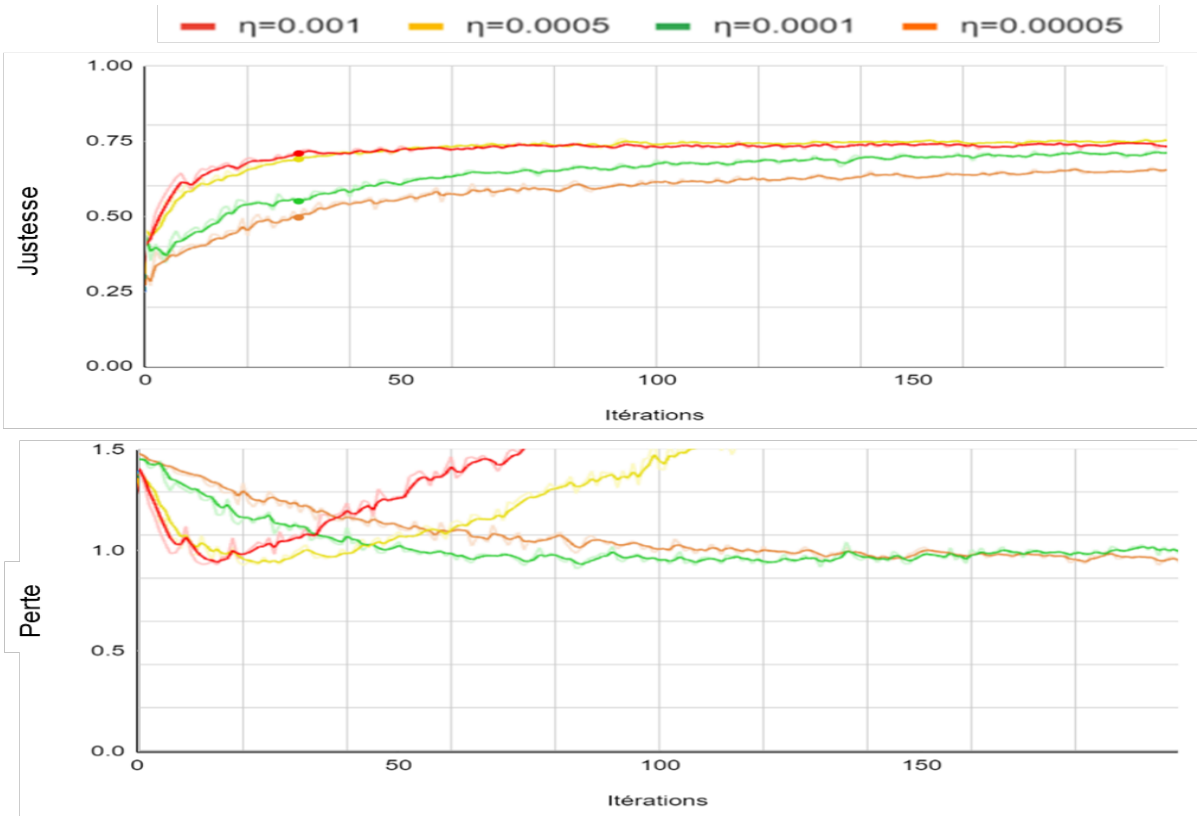


Figure 4.15. Courbes d'apprentissage de validation (Justesse/Perte) pour différentes valeurs de η.

4.7 INFLUENCE DU CHOIX DE LA FRÉQUENCE D'ÉCHANTILLONNAGE

Dans cette étude, nous avons testé différentes fréquences d'échantillonnage, soit 4, 6, 8, 10, 16 et 22 kHz. Pour les différents tests, nous avons utilisé le spectrogramme de Mel étant donné qu'il est la meilleure représentation temps-fréquence. Les résultats présentés à la figure 4.16 montrent que la fréquence d'échantillonnage de 22 kHz donne la meilleure justesse. Ceci peut être justifié par le nombre total des fichiers échantillonnés à 44 kHz qui correspond à 84% de tous les fichiers de la base de données. Lorsque nous effectuons un souséchantillonnage à des fréquences beaucoup plus basses, par exemple 4 kHz, cette opération entraine une perte de qualité des sons et donc une mauvaise performance du système de classification de sons respiratoires. Nous avons également utilisé la fréquence d'échantillonnage de 44 kHz. Cependant, dans cette expérience, nous n'avons pas pu obtenir de résultats en raison du temps de calcul élevé.

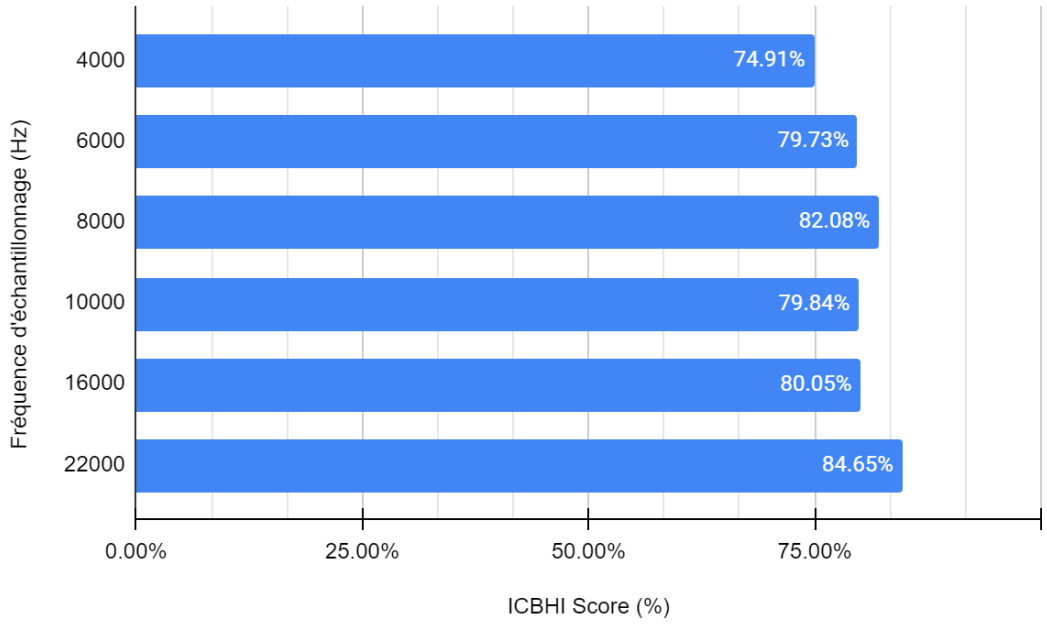


Figure 4.16. Effets de la fréquence d'échantillonnage sur la performance (Score ICBHI) du système de classification.

4.8 INFLUENCE DU CHOIX DE LA FENÊTRE DE PONDÉRATION

Afin de voir l'effet du choix de la fenêtre de pondération sur les performances du système de classification de sons respiratoires, nous avons proposé trois expérimentations différentes en utilisant le spectrogramme de Mel. Cette étude permet de choisir la fenêtre de pondération adéquate qui donne les meilleurs résultats. La figure 4.17 résume les résultats obtenus selon les trois types des fenêtres (Hanning, Hamming et Blackman) qui ont été fréquemment proposées dans la littérature. Nous constatons que le système de classification basé sur la fenêtre de pondération de type Hanning donne le meilleur résultat, soit une justesse de 82.12%. Il faut mentionner que chacune des représentations temps-fréquence proposées se base sur des fenêtres de pondération pour générer des images de spectrogrammes.

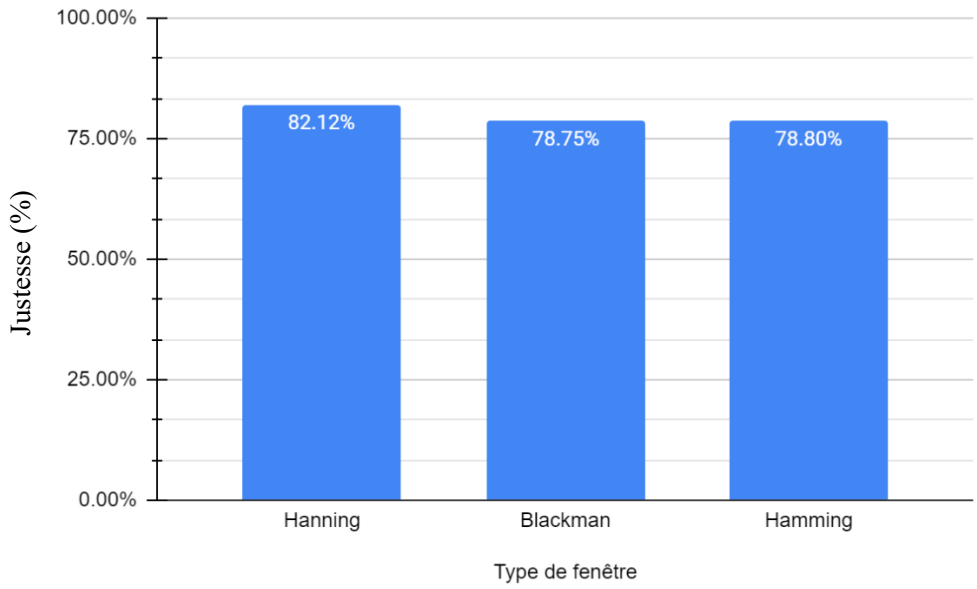


Figure 4.17. Effets du type de fenêtres de pondération sur la performance (Justesse) du système de classification.

4.9 PERFORMANCE DE CLASSIFICATION DU SYSTÈME PROPOSÉ

Les résultats obtenus lors des expériences précédentes nous permettent de déterminer les meilleurs paramètres de configuration pour notre système de classification. Le modèle le plus performant sera évalué selon les critères proposés par la base de données ICBHI

dans le but de tester les performances de la catégorisation multiclasses. Le tableau 4.6 fournit la liste des paramètres sélectionnés. À partir de la meilleure combinaison des paramètres testés, nous avons obtenu une justesse de 97.28% sur l'ensemble d'entrainement et 82.12 % sur l'ensemble de test. La matrice de confusion de notre modèle qui a produit les meilleures performances est présentée dans la figure 4.18.

Tableau 4.6. Paramètres de configuration de notre modèle.

Paramètres	**Valeurs**
Fréquence d'échantillonnage	22 kHz
Longueur de la fenêtre	512
Fenêtre de pondération	Hanning
Longueur du segment	6 seconds
Normalisation	Min-Max
RTF	Mel-Spectrogramme
Taille du lot	256
Taux d'apprentissage	0.0001
Optimiseur	Adam
Régularisation	Normalisation par lots + Dropout + Régularisation L2
Nombre d'itérations	200

Le réseau de neurones convolutif (CNN) conçu dans cette étude vise à classer les sons respiratoires en quatre classes. Nous avons analysé les facteurs, au niveau des caractéristiques, qui affectent la justesse de la classification comme la longueur du cycle respiratoire, la représentation temps-fréquence, l'architecture du réseau jusqu'aux hyperparamètres de notre réseau. Pour mieux exploiter les informations de l'image, nous avons testé plusieurs représentations temps-fréquence. Il a été démontré que l'utilisation du spectrogramme Mel avec le réseau CNN produit de meilleurs résultats que les autres méthodes (voir figure 4.6). Après avoir expérimenté différentes architectures basées sur des réseaux de neurones convolutifs couramment utilisés, tels qu'AlexNet et VGGNet-16, nous avons conçu notre propre architecture basée sur un réseau CNN, qui est illustrée dans la figure 3.4.

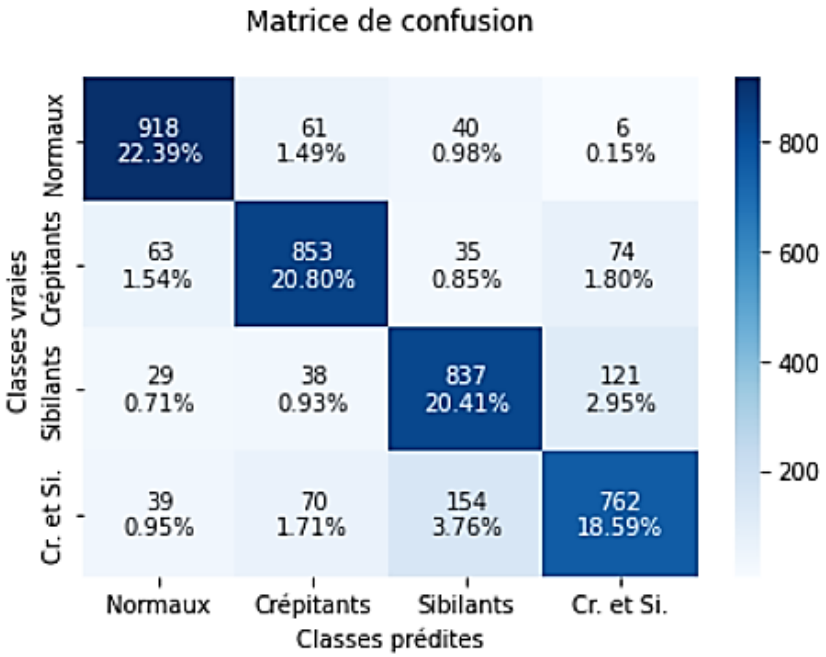


Figure 4.18. Matrice de confusion obtenue pour la classification des anomalies.

4.10 COMPARAISON ET DISCUSSION

Dans le modèle CNN proposé, nous avons intégré l'augmentation des données pour résoudre le problème de généralisation. Par la suite nous avons constaté que l'utilisation d'une combinaison de sous-échantillonnage et de suréchantillonnage présenté par l'approche S3 réduit efficacement le grand déséquilibre des classes tout en fournissant de meilleurs résultats. Ces stratégies ont permis d'améliorer les performances du système conçu avec une capacité suffisante pour apprendre les régularités réelles dans les données d'entrainement, mais pas assez pour mémoriser l'ensemble d'entrainement ou exploiter les régularités accidentelles ce qui permet d'éviter le phénomène de généralisation. Les résultats du tableau 4.7 montrent que la classification par le modèle CNN proposé donne un meilleur score ICBHI, soit 84.65%. Les architectures de références VGG-16 et AlexNet ont obtenu des scores ICBHI moins performant respectivement de 70.66% et 72.94%.

Tableau 4.7. Résumé de la comparaison des performances par modèle.

Architecture	Spécificité (%)	Sensibilité (%)	Justesse (%)	ICBHI $_{score}$ (%)
AlexNet	75.41	70.47	71.70	72.94
VGG-16	72.78	68.55	69.60	70.66
Notre modèle	89.56	79.74	82.20	84.65

L'architecture proposée dans cette étude permet de réduire le temps de calcul grâce à la réduction des paramètres. Cette réduction des paramètres entrainables du réseau est obtenue en utilisant moins de couches de convolutions. Potentiellement, un tel système permet d'éviter les erreurs de généralisation et de suradaptation tout en offrant de meilleures performances globales, car il n'est pas nécessairement avantageux d'avoir une fonction discriminante trop complexe. Les autres réseaux de références comme AlexNet et VGG-16, proposent des architectures plus profondes et un nombre plus élevé de couches de convolutions. En effet, il faut savoir que la profondeur est étroitement liée à la complexité des données et à la tâche de classification à réaliser. Les matrices de confusion ont été utilisées pour calculer les résultats qui sont illustrés dans la figure 4.19.

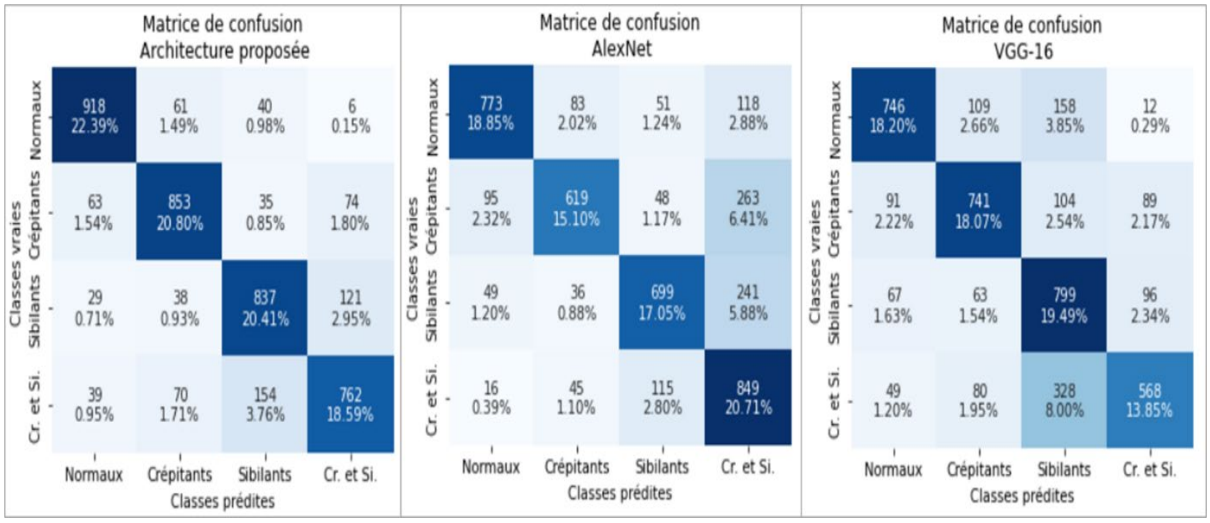


Figure 4.19. Matrices de confusion et pourcentage de justesse par classe pour la classification des anomalies pour les différents modèles.

Il est intéressant de mentionner que la nature de nos données n'est pas complexe, mais très sensible. Le nombre limité des segments disponibles pour l'entrainement rend cette classification très difficile. Pour mieux illustrer l'optimisation réalisée sur ce modèle, nous avons présenté les résultats avant et après respectivement dans les figures 4.20 et 4.21.

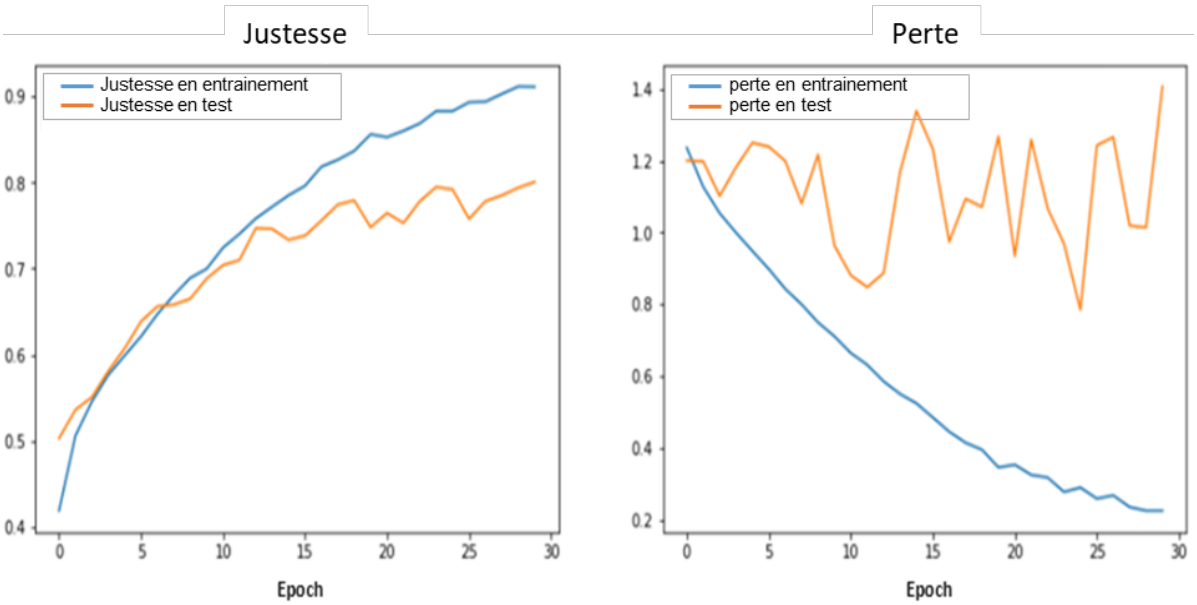


Figure 4.20. Justesse et perte du système de classification des sons respiratoires par CNN (avant optimisation).

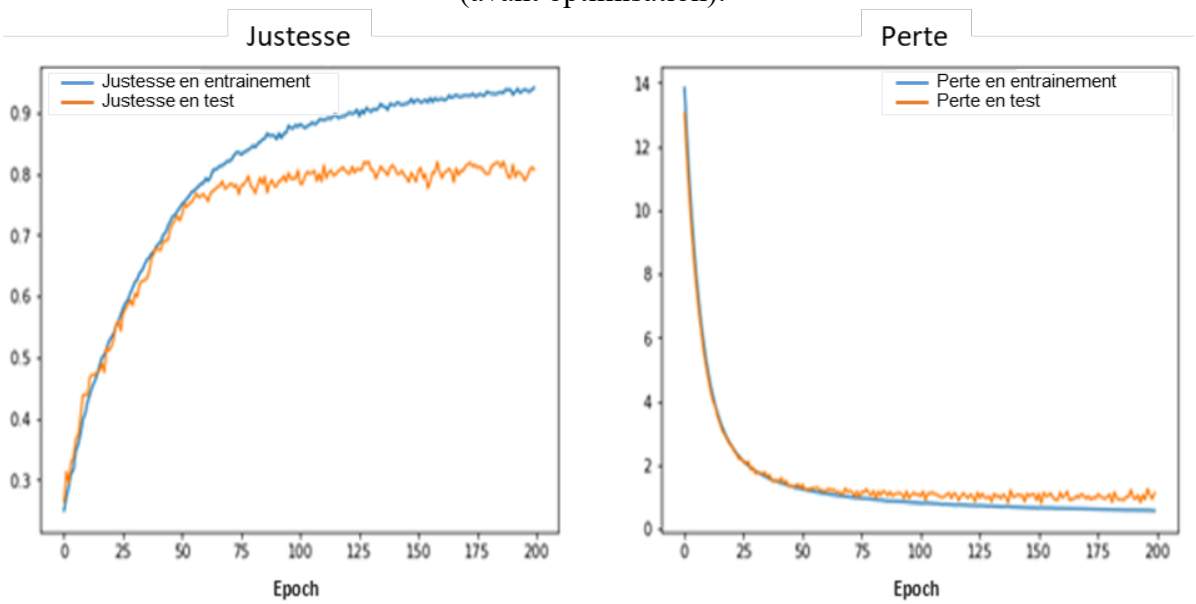


Figure 4.21. Justesse et perte du système de classification des sons respiratoires par CNN (après optimisation).

Les figures ci-dessus présentent les performances globales des ensembles d'apprentissages et de tests avant et après optimisation. Dans la figure 4.20, il est intéressant de constater que juste après un nombre très faible d'itérations (30 itérations), les performances du système ont connu une dégradation brusque alors que le modèle est devenu moins apte à la généralisation. En effet, après l'application des techniques proposées dans les sections précédentes, soit le dropout, la régularisation L2 et la normalisation du lot, donne un effet exclusivement positif (voir la figure 4.21). En revanche, la performance apparait marginalement détériorée à partir de la 50ème itération. En fait, nous pouvons voir ce caractère de dégradation dans les courbes présentant la performance de justesse. La courbe de l'ensemble de test qui est présenté en orange, ne suit plus l'évolution de la courbe d'entrainement. Ce phénomène indique que l'ensemble de données de validation ne fournit pas une quantité suffisante de données. Cela pourrait être justifié par le fait que la base de données est affectée par un déséquilibre sévère entre les classes, ou encore par la complexité des caractéristiques distinctives associées à chaque classe, étant donné que la base de données présentait une forte perturbation en termes de bruit ambiant et parfois de bruit de fond qui masquait toute possibilité de détecter les sons respiratoires. Dans l'ensemble, malgré la complexité de la tâche à accomplir et la taille relativement petite de l'ensemble de données, nous avons obtenu des résultats satisfaisants même avec un nombre élevé d'itérations tant dans la phase d'entrainement que dans la phase de test.

4.11 VALIDATION DES PERFORMANCES DU SYSTÈME DE CLASSIFICATION

Afin de valider les résultats présentés par la troisième approche d'augmentation, nous allons effectuer une série vérification avec la méthode de validation croisée permettent de prouver la robustesse de cette approche. Les résultats sont présentés dans le tableau 4.8.

La figure 4.22 montre un diagramme en boite des résultats de la validation croisée à 3 blocs. La justesse obtenue par les différents modèles variait de 0.587 à 0.801, la sensibilité variait de 0.585 à 0.764 et la spécificité variait de 0.238 à 0.919. Le meilleur modèle présentait une justesse de 0.797, une sensibilité de 0.761 et une spécificité de 0.907. Les

résultats indiquent que la classification par l'architecture proposée garantit la meilleure classification avec un score ICBHI de 83.43 %.

Tableau 4.8. Bilan des performances comparatives des modèles de classification par validation croisée

Architecture	**Architecture proposée**		**AlexNet**		**VGG-16**	
Nombre de blocs (KFolds)	3	5	3	5	3	5
Fréquence d'échantillonnage (Hz)	22000	22000	22000	22000	22000	22000
Fenêtre de pondération	Hanning	Hanning	Hanning	Hanning	Hanning	Hanning
Nombre de filtres Mel	50	50	50	50	50	50
Longueur du segment(s)	6	6	6	6	6	6
Longueur de la fenêtre	512	512	512	512	512	512
RTF	MEL	MEL	MEL	MEL	MEL	MEL
Taille du lot	256	256	256	256	256	256
Taux d'apprentissage	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001
Spécificité	90.72 %	90.84 %	55.49 %	59.53 %	87.77 %	83.47 %
Justesse	79.78 %	79.76 %	62.93 %	59.89 %	76.61 %	77.54 %
Sensibilité	76.13 %	76.06 %	65.41 %	60.01 %	72.88 %	75.56 %
Score ICBHI	**83.43 %**	**83.45 %**	60.45 %	59.77 %	80.33 %	79.52 %

La figure 4.23 montre un diagramme en boite des résultats de la validation croisée à 5 blocs. La justesse obtenue par les différents modèles variait de 0.539 à 0.805, la sensibilité variait de 0.538 à 0.771 et la spécificité variait de 0.316 à 0.951. Les résultats semblent similaires en termes de résultats obtenus par rapport à la validation croisée à 3 blocs. Le modèle AlexNet présente la plus grande variation de spécificité, ce qui explique ses faibles performances en matière de généralisation. Le meilleur score ICBHI est utilisé comme critère pour le choix du meilleur système de classification de sons respiratoires. Le meilleur modèle présentait une justesse de 0.797, une sensibilité de 0.76, une spécificité de 0.908 et un score ICBHI de 83.45 %.

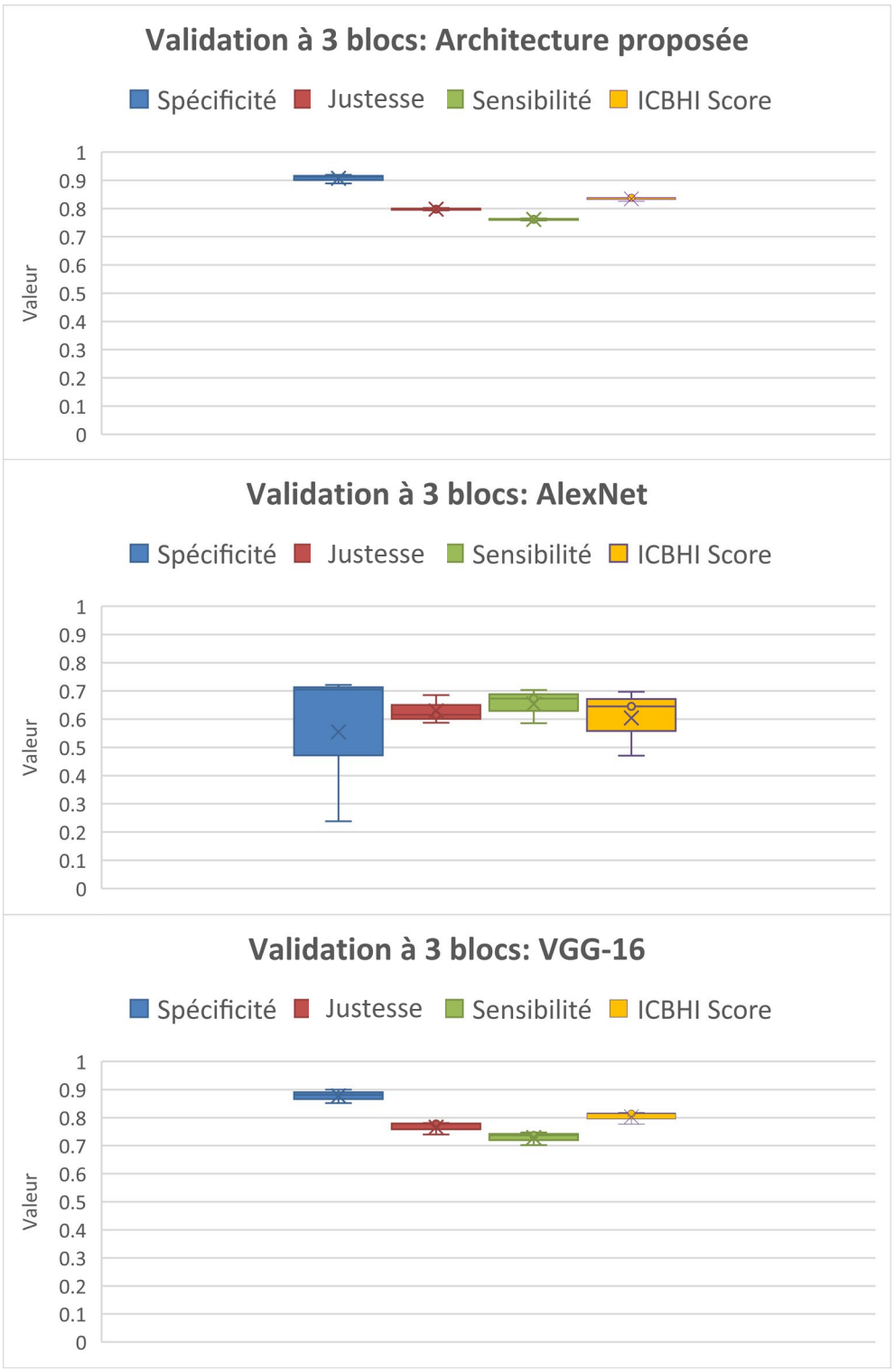


Figure 4.22. Comparaison entre les différents modèles par diagramme en boite des résultats de la validation croisée à 3-blocs.

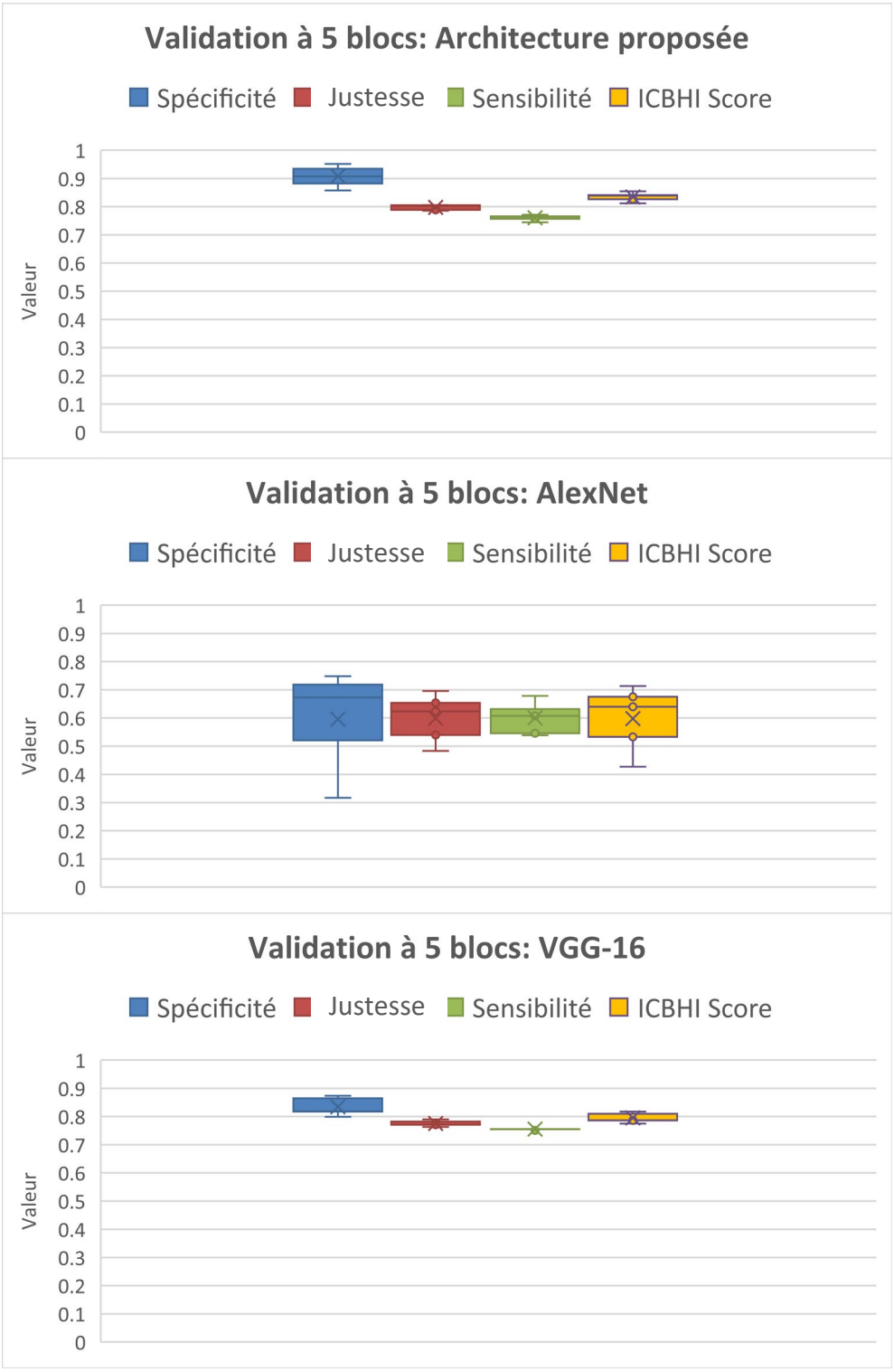


Figure 4.23. Comparaison entre les différents modèles par diagramme en boite des résultats de la validation croisée à 5-blocs.

Dans ce projet, nous avons exploré les réseaux de neurones convolutifs (CNN) pour classifier les sons respiratoires à l'aide d'images de spectrogrammes de Mel. La base de données de référence ICBHI 2017 a été utilisée pour comparer le système de classification développé aux méthodes existantes dans la littérature (voir tableau 4.9). La méthode proposée a démontré des performances remarquables avec un score ICBHI 84.65 % sur la tâche de classification à quatre classes en utilisant 40% de la base de données pour le test. Cependant, ce résultat était vérifié avec la méthode de la validation croisée qui teste toute la base de données. Les résultats présentés dans le tableau tableau 4.9 montrent une légère dégradation du score par rapport au premier score avant d'utiliser la validation croisée avec une moyenne de 83.45 %. Les figures 4.24-4.29 présentent les matrices de confusion résultant de cette expérimentation. On remarque que l'architecture AlexNet présente un important problème de généralisation, par rapport aux deux autres architectures qui ont été remarquablement stables lors des tests de validation croisée à 3 et 5 blocs.

Tableau 4.9. Comparaison avec les systèmes proposés dans l'état de l'art en utilisant la base de données ICBHI.

Méthodes utilisées pour la classification des anomalies à 4 classes	**Score ICBHI** (%)
MFCC-HMM (Rocha *et al*., 2019)	40
STFT-SVM (Rocha *et al*., 2019)	53
MFCC-LSTM (Perna et Tagarelli, 2019)	74
CNN-MoE (Pham *et al*., 2020)	79
CNN-LDA+RSE (Demir *et al*., 2020b)	71
Snapshot ensemble-8 cycles CNN (Nguyen et Pernkopf, 2020)	78.4
CRNN avec CNN-MoE (Pham *et al*., 2021)	80
NMRNN (Kochetov *et al*., 2018)	61
CNN-CBA+BRC+FT (Gairola *et al*., 2020)	68.5
CNN+RNN (Acharya et Basu, 2020)	71.81
CNN (Chanane et Bahoura, 2021)	**80.4**
Modèle proposé	**83.45**

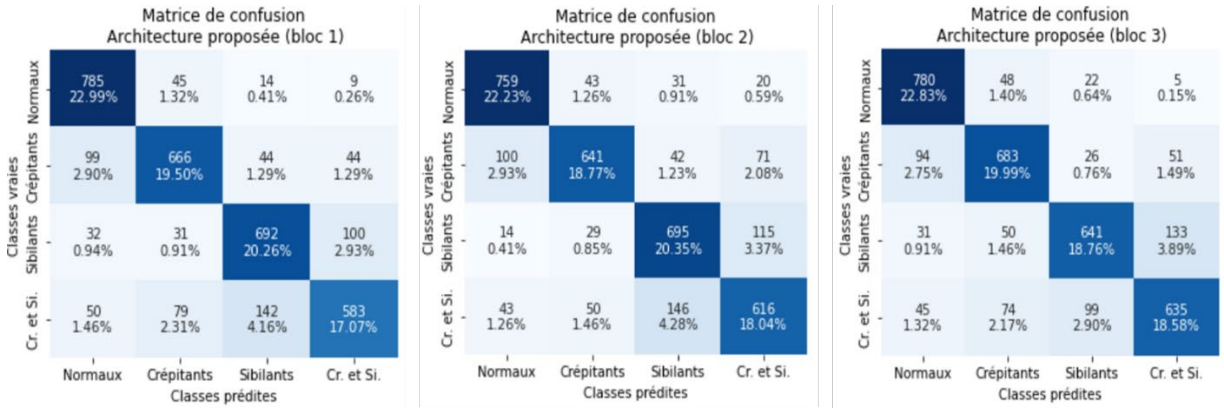


Figure 4.24. Matrices de confusion de la validation croisée à trois blocs obtenus par l'architecture proposée.

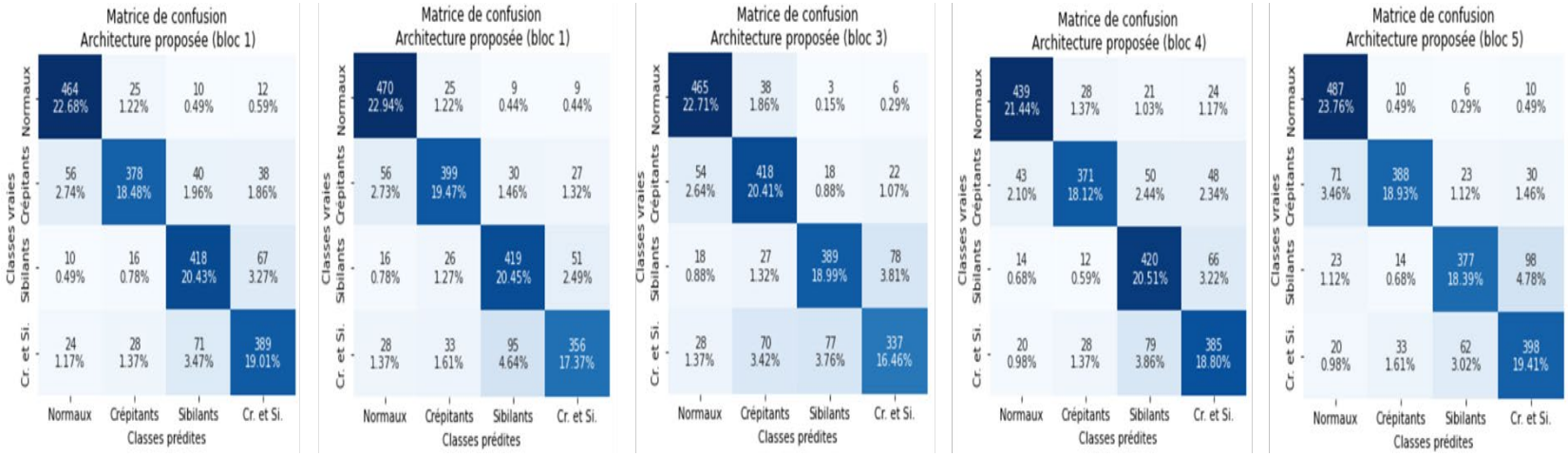


Figure 4.25. Matrices de confusion de la validation croisée à cinq blocs obtenus par le modèle proposé.

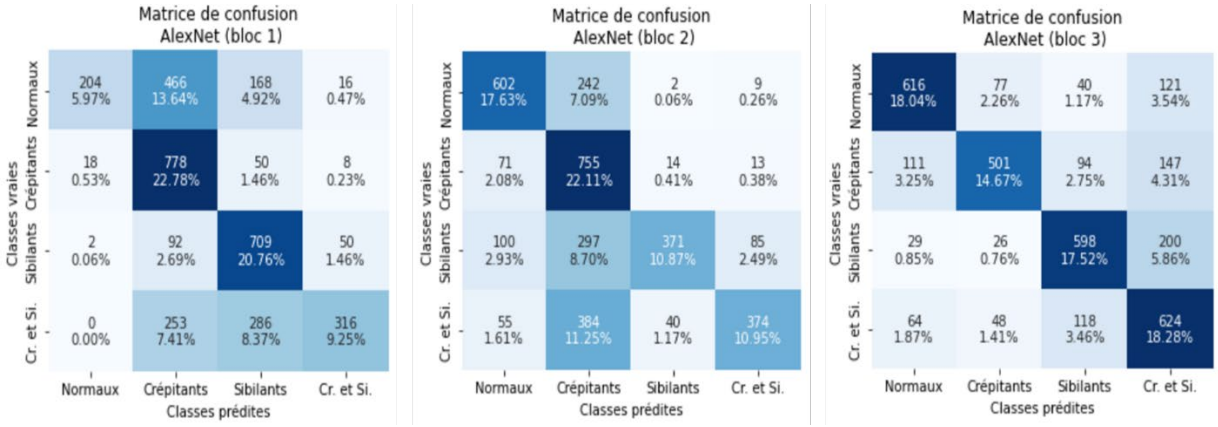


Figure 4.26. Matrices de confusion de la validation croisée à trois blocs obtenus par AlexNet.

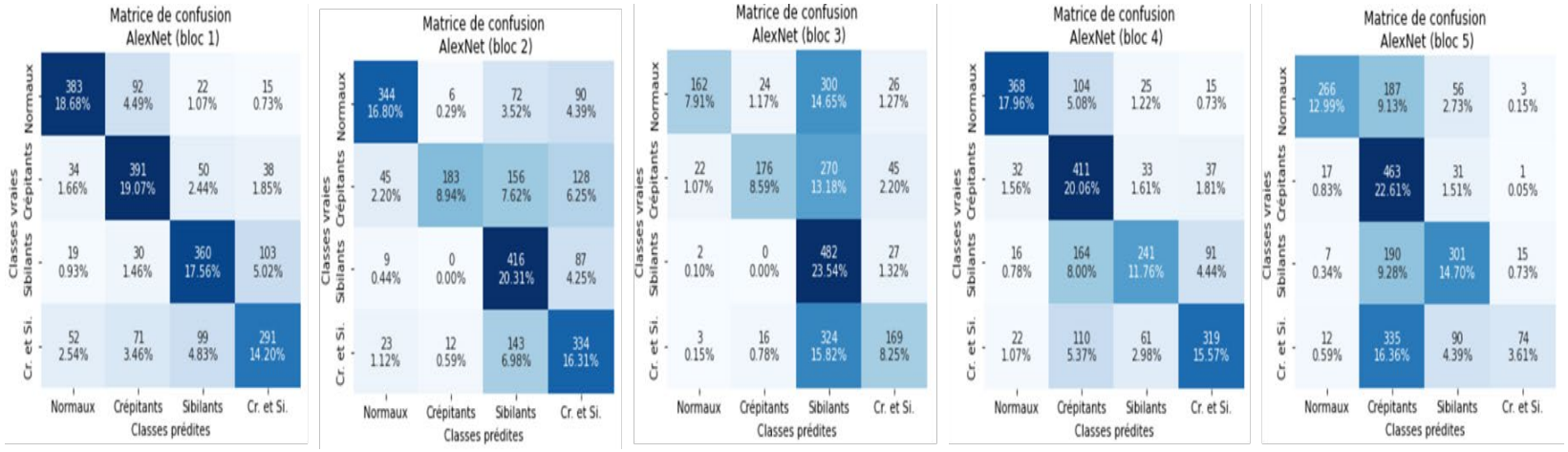


Figure 4.27. Matrices de confusion de la validation croisée à cinq blocs obtenus par AlexNet.

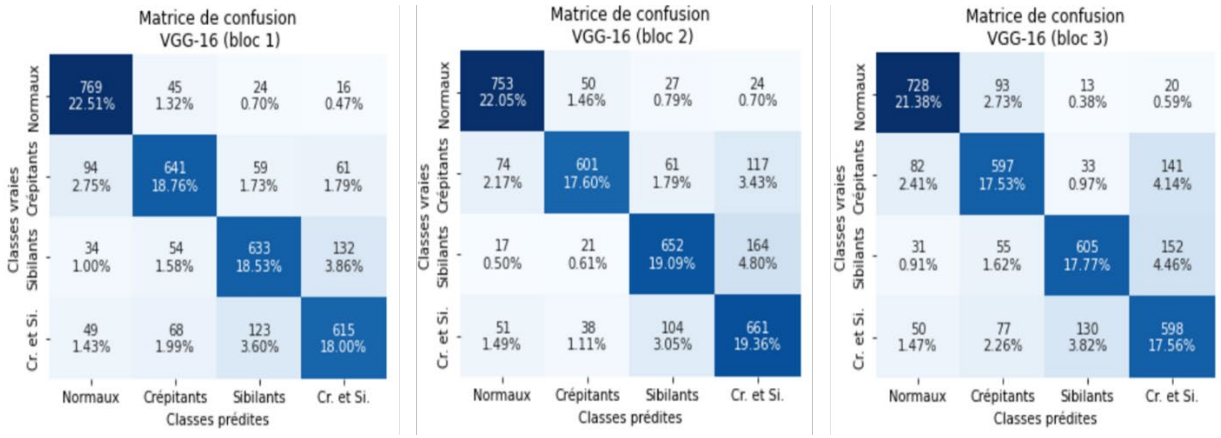


Figure 4.28. Matrices de confusion de la validation croisée à trois blocs obtenus par VGG-16.

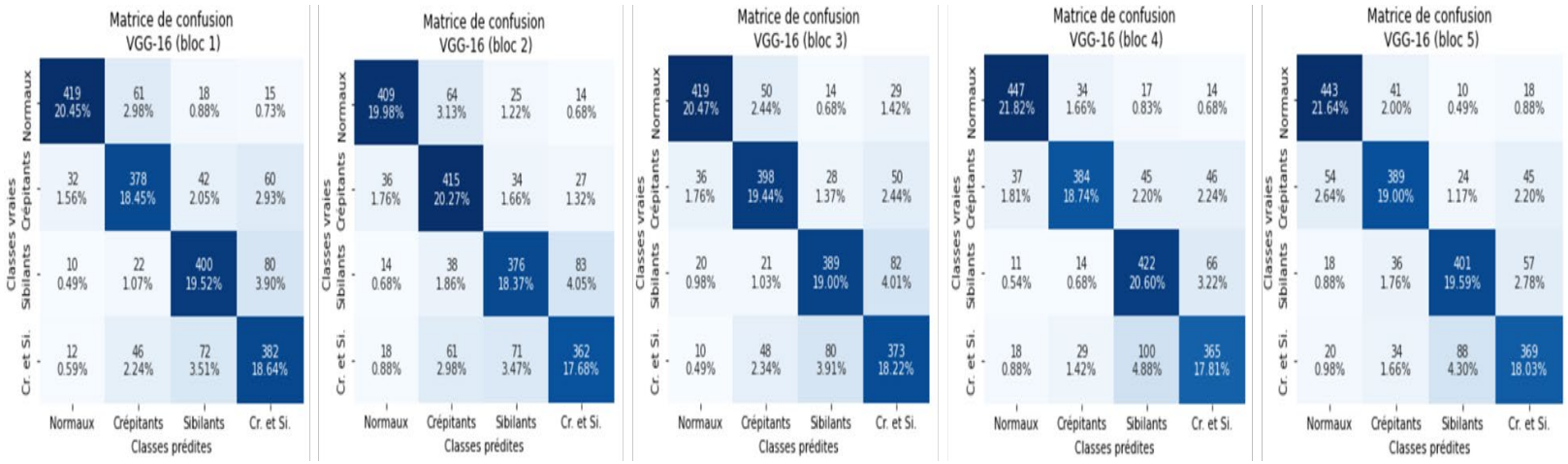


Figure 4.29. Matrices de confusion de la validation croisée à cinq blocs obtenus par VGG-1

CONCLUSION GÉNÉRALE

Ce travail de recherche consiste à construire un système de classification automatique des sons respiratoires adventices. Nous nous sommes particulièrement intéressés aux crépitants, et aux sibilants qui sont considérés comme des signes pour plusieurs complications et maladies respiratoires. Ce projet entre dans le cadre d'une problématique de recherche plus vaste, qui vise à développer des techniques de reconnaissance des sons adventices dans le but de réaliser un système d'aide au diagnostic des maladies pulmonaires.

La première phase a permis de mener une étude expérimentale comparative ayant conduit à l'évaluation et l'analyse des effets des différents types de représentations temps-fréquence couramment utilisées. Les expérimentations ont été réalisées en utilisant le langage Python avec ses différentes bibliothèques. La fréquence d'échantillonnage, la longueur de la fenêtre, la longueur du cycle, le type de fenêtre, le chevauchement des fenêtres et le choix des composantes des IMF sont les paramètres qui ont été considérés comme variables dans cette investigation. La démarche a été divisée en deux parties. Dans la première, nous avons utilisé l'approche temps-fréquence par défaut, tandis que dans la seconde, nous avons utilisé l'EMD en combinaison avec le spectrogramme STFT, le spectrogramme CQT et le spectrogramme de Mel. Les résultats ont montré que le spectrogramme de Mel permet d'avoir un meilleur résultat parmi toutes les techniques utilisées.

La seconde phase a permis de développer un modèle CNN pour classifier les sons respiratoires en utilisant les images de spectrogrammes de Mel. Il a été observé que le fait d'aller plus profond avec les couches de convolutions du CNN affecterait la performance du modèle négativement en termes de perte de validation. À travers cette étude, nous avons pu trouver le nombre optimal de couches de convolutions qui pourrait produire une

meilleure performance avec moins d'erreurs de généralisation. Le modèle est devenu robuste et capable de s'adapter correctement à de nouvelles données jamais vues auparavant. Un autre résultat concerne la taille du filtre du champ réceptif. En effet, nous avons obtenu de meilleurs résultats en utilisant une taille de filtre fixe de 3x3 plutôt que des tailles de filtre plus grandes. Les résultats ont également montré que la normalisation par lots accélère les entrainements et fournit une certaine régularisation tout en réduisant l'erreur de généralisation. Un autre paramètre qui favorise le succès de cette architecture est l'utilisation de la fonction d'activation leakyReLU au lieu de la fonction ReLU standard. Cette fonction d'activation a été développée pour surmonter l'une des principales lacunes de la fonction d'activation ReLU. En ce qui concerne les optimiseurs, nous avons choisi d'utiliser l'algorithme ADAM au lieu de beaucoup d'autres, la raison étant que cet algorithme peut réduire considérablement le temps du processus d'entrainement et nécessite moins de réglage des hyperparamètres.

La troisième phase a permis de développer la stratégie d'optimisation du procédé tout en considérant l'influence de l'ajustement des hyperparamètres du réseau CNN, de l'augmentation des données et des techniques de régularisation des données sur la diminution des erreurs de généralisation afin d'éviter le surapprentissage. Le développement et la validation des modèles sont basés sur les critères proposés dans le cadre du défi ICBHI 2017. Les résultats obtenus révèlent d'excellentes performances et démontrent la pertinence de la stratégie d'optimisation proposée. L'intégration de la régularisation avec différents schémas d'augmentation des données proposés permet de réduire les erreurs de généralisation et augmenter les performances de ce modèle par 4 %. Les effets d'autres techniques tels que le débruitage profond et l'utilisation du RNN avec le réseau CNN peuvent faire l'objet de recherches supplémentaires.

Bien que ces résultats soient très satisfaisants, des investigations numériques et expérimentales supplémentaires sont souhaitables pour enrichir la justesse des résultats, qui ont été considérablement influencés par plusieurs facteurs. Ces facteurs comprennent les appareils utilisés pour collecter les données sonores, l'emplacement du capteur lors de la

collecte des données, le nombre déséquilibré des enregistrements audio dans chaque classe et la durée des différents enregistrements.

Les considérations possibles pour augmenter les performances du système sont énumérées ci-dessous :

- Utilisation d'un ensemble de données plus grand, avec une distribution équilibrée des classes.
- Application de nouvelles techniques avancées de traitement du signal pour extraire des caractéristiques plus pertinentes et distinctives.
- Implémentation d'une architecture combinant le débruitage et la classification. Pour cela nous avons proposé d'utiliser un réseau de neurones convolutif connu sous le nom U-Net. Ce modèle est basé sur l'apprentissage non supervisé et qui est capable de faire un débruitage profond. Il a été appliqué avec succès au rehaussement de la parole, mais pas encore au débruitage des sons respiratoires en raison du manque de bases de données.

RÉFÉRENCES BIBLIOGRAPHIQUES

Abbas, A., & Fahim, A. (2010). An Automated Computerized Auscultation and Diagnostic System for Pulmonary Diseases. Journal of Medical Systems, 34(6), 1149-1155. https://doi.org/10.1007/s10916-009-9334-1

Acharya, J., & Basu, A. (2020). Deep Neural Network for Respiratory Sound Classification in Wearable Devices Enabled by Patient Specific Model Tuning. IEEE Transactions on Biomedical Circuits and Systems, 14(3), 535-544. https://doi.org/10.1109/TBCAS.2020.2981172.

Aykanat, M., Kılıç, Ö., Kurt, B., & Saryal, S. (2017). Classification of lung sounds using convolutional neural networks. EURASIP Journal on Image and Video Processing, 1(2017), 65-74. https://doi.org/10.1186/s13640-017-0213-2.

Bahoura, M., & Pelletier, C. (2004). Respiratory sounds classification using Gaussian mixture models. IEEE Canadian Conference on Electrical and Computer Engineering, 3(2004), 1309-1312. https://doi: 10.1109/CCECE.2004.1349639.

Bahoura, M. (2009). Pattern recognition methods applied to respiratory sounds classification into normal and wheeze classes. Computers in Biology and Medicine, 39(9), 824–843. https://doi.org/10.1016/J.COMPBIOMED.2009.06.011

Bahoura, M. (2018). FPGA implementation of an automatic wheezing detection system. Biomedical Signal Processing and Control, 46(2018), 76-85. https://doi.org/https://doi.org/10.1016/j.bspc.2018.05.017

Bahoura, M. (2019). Efficient FPGA-Based Architecture of the Overlap-Add Method for Short-Time Fourier Analysis/Synthesis. Electronics, 8(12), 1533-1544. https://doi.org/10.3390/electronics8121533

Bardou, D., Zhang, K., & Ahmad, S. M. (2018). Lung sounds classification using convolutional neural networks. Artificial intelligence in medicine, 88(2018), 58–69. https://doi.org/10.1016/j.artmed.2018.04.008.

Boashash, B. (2015). Time-frequency signal analysis and processing: a comprehensive reference. Academic press.

Brown, J. C., & Puckette, M. S. (1992). An efficient algorithm for the calculation of a constant Q transform . The Journal of the Acoustical Society of America, 92(5), 2698–2701. https://doi.org/10.1121/1.404385

Chanane, H., & Bahoura, M. (2021). Convolutional Neural Network-based Model for Lung Sounds Classification. Midwest Symposium on Circuits and Systems (MWSCAS), 555–558. https://doi.org/10.1109/MWSCAS47672.2021.9531887

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16(2002), 321–357. https://doi.org/10.1613/JAIR.953

Chen, H., Yuan, X., Pei, Z., Li, M., & Li, J. (2019). Triple-Classification of Respiratory Sounds Using Optimized S-Transform and Deep Residual Networks. IEEE Access, 7, 32845-32852. https://doi.org/10.1109/ACCESS.2019.2903859

Demir, F., Sengur, A., & Bajaj, V. (2020). Convolutional neural networks based efficient approach for classification of lung diseases. Health information science and systems, 8(1), 1-8. https://doi.org/10.1007/s13755-019-0091-3

Demir, F, Ismael, A. M., & Sengur, A. (2020). Classification of Lung Sounds With CNN Model Using Parallel Pooling Structure. IEEE Access, 8(2020), 105376–105383. https://doi.org/10.1109/ACCESS.2020.3000111

Forkheim, K. E., Scuse, D., & Pasterkamp, H. (1995). A comparison of neural network models for wheeze detection. IEEE WESCANEX 95. Communications, Power, and Computing. Conference Proceedings, 1(1995), 214-219. https://doi.org/10.1109/WESCAN.1995.493973

Fraiwan, L., Hassanin, O., Fraiwan, M., Khassawneh, B., Ibnian, A. M., & Alkhodari, M. (2021). Automatic identification of respiratory diseases from stethoscopic lung sound signals using ensemble classifiers. Biocybernetics and Biomedical Engineering, 41(1), 1–14. https://doi.org/10.1016/J.BBE.2020.11.003

Fulop, S. A. (2011). The Fourier Power Spectrum and Spectrogram. In S. A. Fulop (Ed.), Speech Spectrum Analysis (pp. 69-106). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-17478-0_4

Gairola, S., Tom, F., Kwatra, N., & Jain, M. (2021). RespireNet: A Deep Neural Network for Accurately Detecting Abnormal Lung Sounds in Limited Data Setting. 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), 527-530.

García-Ordás, M. T., Benítez-Andrades, J. A., García-Rodríguez, I., Benavides, C., & Alaiz-Moretón, H. (2020). Detecting Respiratory Pathologies Using Convolutional Neural Networks and Variational Autoencoders for Unbalancing Data. Sensors, 20(4), 1214. https://www.mdpi.com/1424-8220/20/4/1214

Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media.

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A. R., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., & Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Processing Magazine, 29(6), 82–97. https://doi.org/10.1109/MSP.2012.2205597

Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Snin, H. H., Zheng, Q., Yen, N. C., Tung, C. C., & Liu, H. H. (1998). The empirical mode decomposition and the Hubert spectrum for nonlinear and non-stationary time series analysis. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 454(1971), 903–995. https://doi.org/10.1098/RSPA.1998.0193

Huzaifah, M. (2017). Comparison of time-frequency representations for environmental sound classification using convolutional neural networks. ArXiv Preprint ArXiv:1706.07156. https://doi.org/10.48550/arXiv.1706.07156

Jácome, C., Ravn, J., Holsbø, E., Aviles-Solis, J. C., Melbye, H., & Ailo Bongo, L. (2019). Convolutional Neural Network for Breathing Phase Detection in Lung Sounds. Sensors, 19(8), 1798-1808. https://www.mdpi.com/1424-8220/19/8/1798

Jaitly, N., & Hinton, G. E. (2013). Vocal tract length perturbation (VTLP) improves speech recognition. In Proc. ICML Workshop on Deep Learning for Audio, Speech and Language , 117(2013), 21-26.

Kehtarnavaz, N. (2008). CHAPTER 7 - Frequency Domain Processing. In N. Kehtarnavaz (Ed.), Digital Signal Processing System Design (Second Edition, pp. 175-196). Academic Press. https://doi.org/https://doi.org/10.1016/B978-0-12-374490-6.00007-6

Kim, Y., Hyon, Y., Jung, S. S., Lee, S., Yoo, G., Chung, C., & Ha, T. (2021). Respiratory sound classification for crackles, wheezes, and rhonchi in the clinical field using deep learning. Scientific Reports, 11(1), 17186. https://doi.org/10.1038/s41598-021-96724-7

Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization http://arxiv.org/abs/1412.6980

Kochetov, K., Putin, E., Balashov, M., Filchenkov, A., & Shalyto, A. (2018). Noise masking recurrent neural network for respiratory sound classification. In International Conference on Artificial Neural Networks, 208-217. https://doi.org/10.1007/978-3-030-01424-7_21

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Commun. ACM, 60(6), 84–90. https://doi.org/10.1145/3065386.

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

Lehrer, S. (2002). Understanding Lung Sounds. W.B. Saunders. https://books.google.pt/books?id=LURrAAAACAAJ&source=gbs_book_other_versions

Liu, R., Cai, S., Zhang, K., & Hu, N. (2019, 21-24 Nov. 2019). Detection of Adventitious Respiratory Sounds based on Convolutional Neural Network. 2019 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS), 298–303. 10.1109/ICIIBMS46890.2019.8991459

Liu, Y., Lin, Y., Zhang, X., Gao, S., Wang, Z., Zhang, H., & Chen, G. (2019). Lung sound diagnosis with deep convolutional neural network and two-stage pipeline model. In Lecture Notes in Electrical Engineering, 536, 97-114. Springer Singapore. https://doi.org/10.1007/978-981-13-6837-0_8

Loizou, P.C. (2007). Speech Enhancement: Theory and Practice (1st ed.). CRC Press. https://doi.org/10.1201/9781420015836

Loudon, R.G. (1993). Clinical application of wheeze analysis, Monaldi Archives of Chest Disease, 48(5), 583–585.

Lozano, M., Fiz, J. A., & Jané, R. (2013). Estimation of instantaneous frequency from empirical mode decomposition on respiratory sounds analysis. 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 981–984.https://doi.org/10.1109/EMBC.2013.6609667

Mikami, R., Murao, M., Cugell, D. W., Chretien, J., Cole, P., Meier-Sydow, J., Murphy, R. L., & Loudon, R. G. (1987). International Symposium on Lung Sounds. Synopsis of proceedings. Chest, 92(2), 342–345. https://doi.org/10.1378/chest.92.2.342

Mohamed, A. R., Dahl, G., & Hinton, G. (2009, December). Deep belief networks for phone recognition. In NIPS workshop on deep learning for speech recognition and related applications, 1(9), 39-48.

Mushtaq, Z., & Su, S. F. (2020). Efficient classification of environmental sounds through multiple features aggregation and data enhancement techniques for spectrogram images. Symmetry, 12(11), 1822-1856. https://doi.org/10.3390/sym12111822

Nakano, H., Furukawa, T., & Tanigawa, T. (2019). Tracheal sound analysis using a deep neural network to detect sleep apnea. Journal of Clinical Sleep Medicine, 15(8), 1125–1133. https://doi.org/10.5664/jcsm.7804.

Netter, F. H. (2018). Atlas of Human Anatomy: Latin Terminology E-Book: English and Latin Edition. Elsevier Health Sciences.

Nguyen, T, & Pernkopf, F. (2020). Lung Sound Classification Using Snapshot Ensemble of Convolutional Neural Networks. 2020 42nd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC), 760–763. https://doi.org/10.1109/EMBC44109.2020.9176076

Palaniappan, R., Sundaraj, K., & Sundaraj, S. (2014). A comparative study of the svm and k-nn machine learning algorithms for the diagnosis of respiratory pathologies using pulmonary acoustic signals. BMC Bioinformatics, 15(1), 223. https://doi.org/10.1186/1471-2105-15-223

Pelletier, C., 2006. Classification des sons respiratoires en vue d'une détection automatique ´ des sibilants. Université du Québec à Rimouski, Québec, Canada.

Perna, D. (2018). Convolutional Neural Networks Learning from Respiratory data. 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2109–2113. https://doi.org/10.1109/BIBM.2018.8621273

Perna, D., & Tagarelli, A. (2019). Deep auscultation: Predicting respiratory anomalies and diseases via recurrent neural networks. 2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS), 50–55.

Pesu, L., Helistö, P., Ademovič, E., Pesquet, J. C., Saarinen, A., & Sovijärvi, A. R. A. (1998). Classification of respiratory sounds based on wavelet packet decomposition and learning vector quantization. Technology and Health Care, 6, 65-74. https://doi.org/10.3233/THC-1998-6108

Pham, L. D., Phan, H., Palaniappan, R., Mertins, A., & McLoughlin, I. (2021). CNN-MoE based framework for classification of respiratory anomalies and lung disease detection. IEEE Journal of Biomedical and Health Informatics. 25(8), 2938-2947. https://doi.org/10.1109/JBHI.2021.3064237.

Pham, L., McLoughlin, I., Phan, H., Tran, M., Nguyen, T., & Palaniappan, R. (2020). Robust deep learning framework for predicting respiratory anomalies and diseases. 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), 164–167. https://doi.org/10.1109/EMBC44109.2020.9175704.

Planche, B., & Andres, E. (2019). Hands-On Computer Vision with TensorFlow 2: Leverage deep learning to create powerful image processing apps with TensorFlow 2.0 and Keras. Packt Publishing Ltd.

Pramono, R. X. A., Bowyer, S., & Rodriguez-Villegas, E. (2017). Automatic adventitious respiratory sound analysis: A systematic review. PLOS ONE, 12(5), 1–43. https://doi.org/10.1371/journal.pone.0177926

Qawaqneh, Z., Mallouh, A. A., & Barkana, B. D. (2017). Deep neural network framework and transformed MFCCs for speaker's age and gender classification. Knowledge-Based Systems, 115, 5–14. https://doi.org/10.1016/j.knosys.2016.10.008

Rees, & Calverley, P. M. A. (2002). Handbook of chronic obstructive pulmonary disease. Martin Dunitz.

Reichert, S., Gass, R., Brandt, C., & Andrès, E. (2008). Analysis of Respiratory Sounds: State of the Art. Clinical Medicine. Circulatory, Respiratory and Pulmonary Medicine, 2, CCRPM.S530. https://doi.org/10.4137/ccrpm.s530

Rocha, B. M., Filos, D., Mendes, L., Serbes, G., Ulukaya, S., Kahya, Y. P., Jakovljevic, N., Turukalo, T. L., Vogiatzis, I. M., Perantoni, E., Kaimakamis, E., Natsiavas, P., Oliveira, A., Jácome, C., Marques, A., Maglaveras, N., Pedro Paiva, R., Chouvarda, I., & de Carvalho, P. (2019). An open access database for the evaluation of respiratory sound classification algorithms. Physiological Measurement, 40(3), 035001. https://doi.org/10.1088/1361-6579/ab03ea

Rocha, B. M., Pessoa, D., Marques, A., Carvalho, P., & Paiva, R. P. (2020). Automatic Classification of Adventitious Respiratory Sounds: A (Un)Solved Problem?. Sensors 2021, 21(1), 57-76. https://doi.org/10.3390/S21010057

Schörkhuber, C., & Klapuri, A. (2010, July). Constant-Q transform toolbox for music processing. 7th sound and music computing conference (SMC2010), 3-64. https://doi.org/10.5281/zenodo.849741

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. Journal of Big Data, 6(1), 60. https://doi.org/10.1186/s40537-019-0197-0

Shuvo, S. B., Ali, S. N., Swapnil, S. I., Hasan, T., & Bhuiyan, M. I. H. (2020). A Lightweight CNN Model for Detecting Respiratory Diseases from Lung Auscultation Sounds using EMD-CWT-based Hybrid Scalogram. IEEE Journal of Biomedical and Health Informatics, 25, 2595-2603. https://doi.org/10.48550/arXiv.2009.04402

Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. http://arxiv.org/abs/1409.1556

Sovijärvi, A. R. A., Vanderschoot, J., & Earis, J. E. (2000). Standardization of computerized respiratory sound analysis. Eur Respir Rev, 10, 585. http://www.ilsa.cc/referenc.htm

Statistique Canada. “L’asthme et la maladie pulmonaire obstructive chronique (MPOC) au Canada, 2018” Canada, https://www.canada.ca/fr/sante-publique/services/publications/maladies-et-affections/asthme-maladie-pulmonaire-obstructive-chronique-canada-2018.html. Consulté le 20 juillet 2020.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2014). Going Deeper with Convolutions. http://arxiv.org/abs/1409.4842

WHO (World Health Organization). “The top 10 causes of death” WHO, https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death. Consulté le 20 juillet 2020.

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. Insights into Imaging 2018 9:4, 9(4), 611–629. https://doi.org/10.1007/S13244-018-0639-9

Zeiler, A., Faltermeier, R., Keck, I. R., Tomé, A. M., Puntonet, C. G., & Lang, E. W. (2010). Empirical mode decomposition - An introduction. Proceedings of the International Joint Conference on Neural Networks (IJCNN), 1-8. https://doi.org/10.1109/IJCNN.2010.5596829

Zeiler, M. D., & Fergus, R. (2013). Visualizing and Understanding Convolutional Networks. Computer Vision. European Conference on Computer Vision, 818-833. https://doi.org/10.1007/978-3-319-10590-1_5