REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

**UNIVERSITE YAHIA FARES DE MEDEA**

**F**aculté de **T**echnologie

**D**épartement du **G**énie **E**lectrique

# Mémoire de fin d'études de Master

Filière : Électronique
Spécialité : Électronique des systèmes embarqués

# Conception et implémentation d'une solution Edge AI pour la surveillance des roulements et le diagnostic des pannes dans les pompes centrifuges

Présentés par :

**M. TCHOKETCH KEBIR Fouad**

**M. KERROUCHE Abderrahmane**

Proposés et Dirigés par :

**M. CHADLI Samir**

**Dr. BENGHERBIA Billel**

**Promotion 2022-2023**

**Remerciements**

Nous voudrions exprimer notre sincère gratitude à **Allah Tout-Puissant** pour nous avoir donné force, santé et détermination pour mener à bien ce projet. Nous le devons à nos parents pour leur soutien et leurs encouragements indéfectibles, qui ont été une source constante de force tout au long de ce parcours.

Nous adressons nos sincères remerciements à notre estimé professeur et encadrant, **M. CHADLI Samir**, et **Dr. BENGHERBIA Billel** pour leur soutien indéfectible et leurs conseils. Leur confiance en nos capacités et leur motivation continue nous ont poussés à donner le meilleur de nous-mêmes

Nous tenons également à exprimer notre gratitude aux membres du jury, **Dr. TOUBAL Abdelmoughni** et **Dr. REBIAI Mohamed**, pour avoir assisté et évalué notre travail. Leur expérience et leurs commentaires ont été inestimables pour façonner nos projets futurs.

De plus, nous adressons nos remerciements à **M. BERKANI Mohamed Rafik Aymene** et **M. TOBBAL Abdelhafid** pour leurs contributions à la réussite de ce projet.

À nos amis, collègues et à toute la promotion 2023, nous vous remercions pour votre soutien indéfectible. Vos encouragements et votre amitié ont été déterminants tout au long de ce parcours.

En conclusion, nous exprimons notre gratitude à tous ceux qui ont contribué à notre succès. Nous emporterons les leçons et les souvenirs avec nous alors que nous entamons le prochain chapitre de nos vies. Merci à tous de faire partie intégrante de notre parcours universitaire.

**Dédicaces**

*« La vie n'est qu'un éclair, Et un jour de réussite est un jour très cher. »*

*Du profond de mon cœur, Je dédie ce travail à tous ceux qui me sont chers :*

*A mes très chers parents : WAHIBA & ACHOUR*

*Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consentie pour mon instruction et mon bien-être.*

*Je vous remercie pour tout le soutien et l'amour que vous me portez depuis mon enfance et j'espère que votre bénédiction m'accompagne toujours.*

*Que ce modeste travail soit l'exaucement de vos vœux tant formulés, le fruit de vos innombrables sacrifices. Puisse ALLAH, le très Haut, vous accorde santé, bonheur et longue vie.*

*Je dédie également ce succès à mon grand-père Tayeb, décédé en 2004, que ALLAH lui fasse miséricorde*

*Et mon grand-père Mohamed et à ma grand-mère Atika décédée en 2018, que ALLAH ait pitié d'eux*

*A mon cher frère Tayeb, qui n'a jamais cessé de me conseiller, m'encourager et me soutenir tout au long de ma vie ; Que ALLAH vous protège et t'offre la chance et le bonheur.*

*A mon binôme Abderrahmane, avec qui je travaille en étroite collaboration depuis notre deuxième année, et avec qui j'ai relevé tous les défis ensemble.*

*A toute la famille TCHOKETCH KEBIR et à toute la famille OTSMAN.*

*A tous mes ami(e)s sans exception.*

*A tous ceux que j'aime, et à tous ceux qui m'aiment... de près ou de loin.*

**« *FOUAD* »**

**Dédicaces**

*Il y a des moments où les mots ne suffisent pas pour exprimer la gratitude, l'amour, le respect et l'appréciation que l'on ressent. Ce sont des sentiments si profonds et si intenses qu'ils transcendent les frontières du langage. Dans des moments comme celui-ci, les mots se perdent dans le silence et nous cherchons d'autres moyens de montrer notre appréciation.*

*Je dédie cette thèse...*

*A mes chers parents pour leur amour inconditionnel, leurs encouragements et leur soutien.*

*À ma mère, DJAZIRI, je vous remercie pour vos prières et vos encouragements, et grâce à ce que j'ai, j'ai pu surmonter tous les obstacles, car vous êtes unique en votre genre.*

*A mon père KERROUCHE, qui m'a beaucoup aidé, m'a tout donné pour que je puisse y parvenir*

*Travail, merci d'être toujours à mes côtés. Considérez un autre paragraphe*

*Merci à mes sœurs et mon frère qui m'ont soutenu*

*Et un merci particulier à mon ami FOUAD, je vous souhaite encore plus de réussite et d'excellence.*

*A toute ma famille*

*Que Dieu Tout Puissant vous protège et prolonge la vie*

*Je ne pourrai jamais exprimer le respect que j'ai pour vous. Votre prière est votre encouragement*

*Votre soutien m'a toujours été d'une grande aide.*

*Amis, chacun par son nom.*

*A tous ceux qui ont aidé, encouragé et contribué directement ou indirectement à la réalisation de cet objectif de ce travail.*

*Merci d'être toujours à mes côtés*

*Dans ce moment-là, il est essentiel de se rappeler que même si les mots peuvent parfois échouer, c'est le but derrière eux qui compte vraiment. C'est l'intention qui guide nos actions qui nous pousse à exprimer notre gratitude et notre amour de toutes les manières possibles.*

***« Abderrahmane »***

**ملخـــص**

نقدم في هذا العمل طريقة لاستخراج خصائص الاشارات الصوتية الصادرة عن محامل الكريات في مضخات الطرد المركزي. كما هو معروف فمشكل محامل الكريات يؤثر بشكل كبير في جميع الآلات و اي عطل في هذه المحامل يودي الى خسارة كبيرة . ولحل هذا المشكل تم اقتراح نظام ذكي يعتمد على التعرف على الاصوات الناتجة عن اعطال محامل الكريات اعتمادا على طريقة MFCC للتنبؤ بموقع الخلل على المحمل باستخدام شبكة عصبية تلافيفية (CNN). تم تنفيذ الحل المقترح على لوحة التطوير Raspberry Pi 3b+ التي أعطت نتائج مقبولة.

**الكلمات المفتاحية**: CNN، librosa، MFCC ، مضخات الطرد المركزي، محامل كروية ، Raspberry Pi ،اشارات صوتية.

**Résumé**

Dans ce travail, nous présentons une méthode d'extraction des caractéristiques des signaux acoustiques générés par les roulements à billes dans les pompes centrifuges. Comme nous le savons, le problème des roulements à billes affecte grandement toutes les machines, et un dysfonctionnement des roulements entraîne une grande perte. Pour résoudre ce problème, un système intelligent a été proposé basé sur la reconnaissance des sons résultant de la défaillance des roulements basé sur la méthode MFCC afin de prédire la localisation du défaut sur le roulement à l'aide d'un Réseau de Neurones Convolutionel (CNN). La solution proposée a été implémentée sur une carte Raspberry Pi 3b+ qui a donnée des résultats acceptables.

**Mots clés** : CNN, librosa, MFCC, pompes centrifuges, Raspberry Pi, roulements à billes, signaux acoustiques.

**Abstract**

In this work, we present a method for extracting the characteristics of the acoustic signals generated by ball bearings in centrifugal pumps. As we know, the problem of ball bearings affects all machines greatly, and bearing malfunction leads to great loss. To solve this problem, an intelligent system has been proposed based on the knowledge of the sounds resulting from bearing failure. The proposed method is to process the MFCC then to predict the location of the fault on the bearing using a Convolutional Neural Network (CNN), the proposed manipulation has been implemented on the Raspberry pi 3b+ board. It is proven to be effective and has reliable results.

**Keywords**: CNN, librosa, MFCC, centrifugal pumps, Raspberry Pi, ball bearings, acoustic signals.

# Sommaire

Liste des Figures ..... I

Liste des Tableaux ..... IV

Liste des Abréviations ..... V

Introduction Générale ..... 1

Chapitre I : Généralité sur les pompes centrifuges

I.1 Introduction ..... 3

I.2 Pompes centrifuges ..... 3

I.2.1 Corps de la pompe (ou boîtier) ..... 4

I.2.2 Roue (ou rotor) ..... 4

I.2.3 Diffuser ..... 5

I.2.4 Arbre ..... 5

I.2.5 Palier (ou roulement) ..... 5

I.2.6 Garniture mécanique ..... 6

I.2.7 Moteur ..... 6

I.3 Utilisation ..... 7

I.3.1 Approvisionnement en eau ..... 7

I.3.2 Traitement des eaux usées ..... 7

I.3.3 Industrie chimique et pétrolière ..... 7

I.3.4 Industrie alimentaire et pharmaceutique ..... 7

I.3.5 Industrie pétrochimique ..... 8

I.3.6 Industrie minière ..... 8

I.4 Fonctionnement ..... 8

I.4.1 Aspiration ..... 8

I.4.2 Accélération ..... 8

I.4.3 Conversion de l'énergie cinétique en énergie de pression ..... 8

I.4.4 Évacuation du fluide ..... 8

I.5 Les Avantages ..... 9

I.5.1 Débit régulier ..... 9

I.5.2 Pression de fonctionnement élevée ..... 9

I.5.3 Capacité à traiter des fluides visqueux ..... 9

I.5.4 Maintenance réduite ..... 9
I.5.5 Taille et poids réduits ..... 9
I.5.6 Coût initial inférieur ..... 9
I.6 Analyse acoustique des machines tournantes ..... 10
I.6.1 Les aspects importants de l'analyse acoustique des machines tournantes ..... 10
I.6.1.1 Acquisition des signaux ..... 10
I.6.1.2 Analyse spectrale ..... 10
I.6.1.3 Détection des signatures de fréquence ..... 10
I.6.1.4 Comparaison avec les valeurs de référence ..... 10
I.6.1.5 Diagnostic des défauts ..... 10
I.6.2 Définition d'une acoustique : ..... 11
I.6.2.1 Fréquence : ..... 11
I.6.2.2 Amplitude ..... 11
I.6.2.3 Durée ..... 11
I.6.2.4 Direction et localisation ..... 11
I.6.2.5 Phase ..... 11
I.7 Les roulements ..... 12
I.7.1 Roulements à billes et à rouleaux ..... 12
I.7.2 Structure mécanique d'un roulement ..... 13
I.7.2.1 Bague intérieure : ..... 13
I.7.2.2 Bague extérieure : ..... 13
I.7.3 Éléments roulants : ..... 13
I.7.3.1 Cage ..... 13
I.7.3.2 Joints ou boucliers ..... 13
I.7.4 Causes des défauts des roulements à billes ..... 14
I.7.4.1 L'usure ..... 14
I.7.4.2 Dentelure ..... 14
I.7.4.3 Bavures ..... 15
I.7.4.4 Ecaillage ..... 15
I.7.4.5 Corrosion ..... 16
I.7.4.6 Coup, fissures, Cassures ..... 16
I.7.4.7 Passage de courant ..... 17
I.7.5 Les types de défauts les roulements ..... 17
I.7.5.1 Défaut au niveau de la bague externe ..... 17

I.7.5.2 Défaut de type écaillage affectant la bague interne ..... 18
I.7.5.3 Défaut de type écaillage sur un élément roulant ..... 18
I.7.5.4 Défauts de type déversement de bague ..... 18
I.7.6 Fréquences caractéristiques ..... 19
I.8 Conclusion ..... 19
Chapitre II : Intelligence Artificielle et MFCC
II.1 Introduction ..... 20
II.2 Deep Learning ..... 20
II.3 Deep Learning vs. Machine learning ..... 21
2.1 Type d'apprentissage ..... 22
II.3.1 Apprentissage supervisé ..... 22
II.3.2 Apprentissage non-supervisé ..... 22
II.4 Détection des anomalies ..... 23
II.5 Les réseaux de neurones ..... 23
II.5.1 Neurone biologique ..... 23
II.5.2 Des réseaux de neurones artificiels ..... 24
II.6 Perceptron ..... 24
II.6.1 Un perceptron ..... 24
II.6.2 Fonctionne Un Perceptron ..... 25
II.6.3 Les fonctions d'activation ..... 25
II.6.3.1 Types de fonctions d'activation : ..... 25
II.6.3.1.1 Fonction d'étape binaire ..... 26
II.6.3.1.2 Fonction d'activation linéaire ..... 26
II.6.3.1.3 Fonctions d'activation non linéaires ..... 26
II.7 Algorithmes Deep Learning ..... 29
II.7.1 Multi layer Perceptrons ..... 29
II.7.1.1 Fonctionne l'algorithme d'apprentissage en profondeur MLP ..... 29
II.7.1.2 Avantages du MLP ..... 30
II.7.1.3 Inconvénients du MLP ..... 30
II.7.2 Récurrent Neural Networks ..... 30
II.7.2.1 Fonctionnent les algorithmes d'apprentissage en profondeur RNN ..... 31
II.7.2.2 La structure de base de RNN ..... 31

II.7.2.3 Avantages de RNN ..... 31
II.7.2.4 Inconvénients de RNN ..... 31
II.7.3 Long Short-Term Memory Networks (LSTM): ..... 31
II.7.3.1 Fonctionnent les algorithmes d'apprentissage en profondeur LSTM ..... 32
II.7.3.2 Avantages de LSTM ..... 32
II.7.3.3 Inconvénients de LSTM ..... 32
II.7.4 Convolution Neural Networks ..... 33
II.7.4.1 Opération de convolution ..... 33
II.7.4.2 Les couches de CNN ..... 34
II.7.4.2.1 La couche d'entrée ..... 34
II.7.4.2.2 La couche de convolution ..... 34
II.7.4.2.3 Mise en commun maximale (Max pooling) ..... 35
II.7.4.2.4 Couche d'aplatissement (Flattenning layer) ..... 36
II.7.4.2.5 Entièrement connecté (Fully connected) ..... 37
II.7.4.3 La normalisation par lots (BN) ..... 37
II.7.4.4 Abandon (Dropout) ..... 37
II.7.4.5 L'optimisation des performances des réseaux de neurones convolutifs (CNN) 38
II.7.4.6 Facteurs peuvent affecter les performances des réseaux de neurones convolutifs (CNN) 39
II.7.4.7 Un exemple dans CNN ..... 39
II.7.4.7.1 Base de données ..... 39
II.7.4.7.2 La formation ..... 40
II.7.5 CIFAR CNN Exemple ..... 41
II.8 Extraction des caractéristiques des signaux acoustiques ..... 44
II.8.1 Kurtogramme ..... 45
II.8.2 Spectrogramme ..... 45
II.8.3 MFCC ..... 46
II.8.3.1 Étapes de calcul de la matrice MFCC ..... 47
II.8.3.1.1 Cadrage en mfcc (segmentation) ..... 47
II.8.3.1.2 Fenêtrage ..... 48
II.8.3.1.3 Transformée de Fourier rapide (FFT) ..... 49
II.8.3.1.4 Banc de filtres Mel ..... 50
II.8.3.1.5 Le logarithme ..... 51
II.8.3.1.6 Transformée discrète en cosinus (DCT) ..... 51

II.8.3.1.7 Puissance spectrale ..... 52
II.9 Application de la méthode MFCC sur les signaux acoustiques ..... 52
II.9.1 Base de données choisie ..... 52
II.9.1.1 Banc d'essai ..... 52
II.9.2 Le Modèle CNN ..... 55
II.9.2.1 Python ..... 55
II.9.2.2 TensorFlow et Keras ..... 56
II.9.2.3 Librosa ..... 56
II.10 Conclusion ..... 56
Chapitre III : implémentation, test et résultats
III.1 Introduction ..... 57
III.2 La base de donnes ..... 57
III.3 Segmentation les signaux ..... 58
III.4 Traitement MFCC sur les signaux acoustiques ..... 59
III.5 Décomposition ..... 60
III.6 Modèle CNN ..... 60
III.7 Le modèle CNN proposé ..... 60
III.8 Résultats de classification ..... 61
III.9 Implémentation ..... 67
III.9.1 Raspberry Pi 3B+ ..... 68
III.9.2 Tensorflow ..... 68
III.9.3 Préparation du modèle ..... 69
III.9.4 Résultat d'implémentation ..... 69
III.10 Conclusion ..... 73
Conclusion générale ..... 74
Références bibliographiques ..... 75
Annexes
Annexe A ..... 81
Annexe B ..... 82
Annexe C ..... 83

Annexe D ..... 84

Annexe E ..... 87

# Liste des Figures

Chapitre Ⅰ :

Figure 1. 1: Pompe centrifuge [2].....3
Figure 1. 2 : Pompe centrifuge avec moteur [4].....4
Figure 1. 3 : Les différents types de roues [5].....4
Figure 1. 4 : Diffuser de pompe centrifuge [6].....5
Figure 1. 5 : Arbre avec double roue radiale [6].....5
Figure 1. 6 : Roulement à billes à gorge profonde.....6
Figure 1. 7 : La garniture mécanique [8].....6
Figure 1. 8 : Moteur de pompe [9].....7
Figure 1. 9 : frottement par glissement et roulements [17].....12
Figure 1. 10 : les roulements [17].....13
Figure 1. 11 : Composantes d'un palier [18].....14
Figure 1. 12 : L'usure abrasive dans une bague intérieure [19].....14
Figure 1. 13 : Défaut de dentelure sur la bague interne d'un roulement à bille [19].....15
Figure 1. 14 : Défaut de bavure sur un roulement à billes [19].....15
Figure 1. 15 : Défaut d'écaillage sur la bague interne d'un roulement à billes [19].....16
Figure 1. 16 : Corrosion sur d'un roulement à billes [19].....16
Figure 1. 17 : Cassures d'un roulement à billes [19].....17
Figure 1. 18 : Passage de courant d'un roulement à bill [19].....17
Figure 1. 19 : Défaut au niveau de la bague externe.....18
Figure 1. 20 : Défaut de type écaillage affectant la bague interne.....18
Figure 1. 21 : Défaut de type écaillage sur un élément roulant.....18
Figure 1. 22 : Défaut de type déversement de bague.....18

Chapitre Ⅱ :

Figure 2. 1 : les domaines de l'intelligence artificielle [29].....21
Figure 2. 2: example Deep learning vs. Machine learning [28].....22
Figure 2. 3 : Neurone biologique [26].....24
Figure 2. 4 : réseaux de neurones artificiels [27].....24
Figure 2. 5 : un perceptron [30].....25
Figure 2. 6 : Les fonctions d'activation [31].....28
Figure 2. 7 : Multi-layer Perceptrons [33].....29

Figure 2. 8 : fonctionnent les algorithmes d'apprentissage en profondeur RNN [33] ..... 31
Figure 2. 9 : fonctionnent les algorithmes d'apprentissage en profondeur LSTM [33] ..... 32
Figure 2. 10 : A simple CNN architecture [36] ..... 34
Figure 2. 11 : les étapes de couche convolution ..... 34
Figure 2. 12 : Max pooling ..... 36
Figure 2. 13 : couche d'aplatissement (flattening layer) ..... 36
Figure 2. 14 : Réseau entièrement connecté ..... 37
Figure 2. 15 : La technique d'abandon ..... 38
Figure 2. 16 : Architecture CNN conçue pour la classification des fruits ..... 41
Figure 2. 17 : Courbes de (a) perte moyenne et (b) précision moyenne lors de la formation du modèle pour la classification des fruites ..... 42
Figure 2. 18 : Exemple de cartes d'activation de la 3e couche de convolution pour deux images du jeu de données Fruit-360 ..... 43
Figure 2. 19 : Matrice de confusion pour les résultats de la classification des fruits sur l'ensemble de données Fruit 360 ..... 43
Figure 2. 20 : kurtogramme ..... 45
Figure 2. 21 : spectrogramme ..... 46
Figure 2. 22 : MFCC ..... 46
Figure 2. 23 : Cadrage en MFCC (segmentation) ..... 48
Figure 2. 24 :fenêtre de hamming ..... 49
Figure 2. 25 : Filter Bank on a Mel-Scale ..... 50
Figure 2. 26 : Spectrogramme du signal ..... 51
Figure 2. 27 : Etapes de calcul de la matrice MFCC ..... 52
Figure 2. 29 : Une photographie typique du banc d'essai (A) avec capteur acoustique (B) ..... 54
Figure 2. 28 : Schéma du banc d'essai avec accéléromètre ..... 54
Figure 2. 30 : Photographies du composant défectueux à l'étude ..... 55

Chapitre Ⅲ :

Figure 3. 1 : signal de cas roulement Sans défaut ..... 58
Figure 3. 2 : 1er segment de signal cas roulement Sans défaut ..... 59
Figure 3. 3 : signal temporel ..... 59
Figure 3. 4 : matrice MFCC ..... 59
Figure 3. 5 : modèle CNN proposé ..... 61
Figure 3. 6 : Résultat du modèle CNN proposé ..... 62

Figure 3. 7 : premier segment des données de test finales .....63
Figure 3. 8 : Calculez les caractéristiques MFCC pour le premier segment des données de test finales. .....63
Figure 3. 9 : les résultats de test final .....64
Figure 3. 10 : temps de traitement par modele tflite.....65
Figure 3. 11 : résultat de prédiction avec percentage pour chaque image (segment) multi class .....66
Figure 3. 12 : Temps de traitement par modèle tflite .....67
Figure 3. 13 : Organigramme des étapes d'implémentation.....67
Figure 3. 14 : Raspberry Pi 3 Model B+ .....68
Figure 3. 15 : Préparation des données.....69
Figure 3. 16 : affichage sur Écran HDMI 5 pouces (Annexe D).....71
Figure 3. 17 : Résultat de prédiction.....72
Figure 3. 18 : résultat de prédiction avec percentage pour chaque image (segment) multi class .....72

# Liste des Tableaux

Chapitre Ⅱ :

Tableau 2. 1 : paramétres du modèle CNN pour la classification des fruits ..... 42
Tableau 2. 2 : Comparaison entre l'exemple proposé et les modèles pré-entrainés pour la classification des fruits sur l'ensemble de données Fruit-360 ..... 44
Tableau 2. 3 : Spécification de la pompe centrifuge ..... 53
Tableau 2. 4: Spécification du roulement à billes profond à une rangée à l'étude (NBC : 6203ZZ) ..... 53
Tableau 2. 5 : Différentes condition de pompe à l'étude ..... 54

Chapitre Ⅲ :

Tableau 3. 1 : Partage des images du la base de données des images MFCC ..... 60
Tableau 3. 2 : résultat de prédiction de test ..... 61
Tableau 3. 3 : temps de traitement (test final) ..... 73
Tableau 3. 4 : temps de traitement (test multi classe) ..... 73

# Liste des Abréviations

**BPFO:** Ball Pass Frequency Outer Race

**BPFI:** Ball Pass Frequency Inner Race

**BPF:** Ball Pass Frequency

**BCF:** Ball Cage Frequency

**ANN:** artificial neural networks

**Relu:** Rectified Linear Unit

**MLP:** Multi-layer Perceptrons

**RNN:** Recurrent Neural Networks

**LSTM:** Long Short-Term Memory Network

**CNN:** Convoluted neural networks

**FC:** fully connected layer

**BN:** Batch Normalization

**DNN:** deep neural networks

**MFCC:** Mel frequency cepstral coefficients

**STFT:** short-term Fourier transform

**FFT:** fast fourier transform

**DCT:** Discrete transformed into cosine

**API:** Application Programming Interface

**SGD :** Stochastic Gradient Descent

# Introduction Générale

# *Introduction générale*

Dans l'industrie moderne, les pompes sont largement utilisées dans diverses applications, et très souvent, la continuité de la production repose sur le bon fonctionnement de ces machines. Cependant, des défauts peuvent survenir et avoir un impact sur les performances des pompes, et il est primordial de détecter ces défauts tôt pour éviter des pannes catastrophiques. La surveillance de l'état de ces machines peut être mise en œuvre à l'aide de méthodes traditionnelles telles que l'analyse globale des signaux de vibrations et des signaux acoustiques. De nouvelles approches basées sur des méthodes automatiques où l'intelligence artificielle (IA) entre en jeu, sont de plus en plus développées pour classer avec précision les différents défauts liés à ce type de machines.

L'intelligence artificielle (IA) et l'apprentissage en profondeur (Deep Learning) sont des domaines de recherche et de développement qui ont eu un impact significatif sur les systèmes embarqués. L'IA fait référence à la capacité des machines à simuler l'intelligence humaine, en utilisant des algorithmes et des modèles mathématiques pour accomplir des tâches spécifiques. L'apprentissage en profondeur est une branche de l'IA qui se concentre sur l'utilisation de réseaux de neurones artificiels multicouches pour apprendre à partir de grandes quantités de données. Dans le domaine des systèmes embarqués, l'IA et le Deep Learning ont ouvert de nouvelles possibilités et amélioré les performances de ces systèmes. Par exemple, les voitures autonomes utilisent des techniques d'apprentissage en profondeur pour détecter et reconnaître des objets sur la route, permettant une conduite autonome.

En outre, les systèmes embarqués utilisent également l'IA et l'apprentissage en profondeur pour la détection d'anomalies, la classification des données, la reconnaissance vocale, la vision par ordinateur et de nombreuses autres applications. Ces techniques permettent aux systèmes embarqués d'être plus autonomes, intelligents et capables de prendre des décisions en temps réel.

Dans ce travail nous aborderons la mise en œuvre pratique d'une méthode de détection et de classification des défauts de roulement des paliers utilisés sur les pompes centrifuges.

Dans un premier temps, nous commencerons par la préparation de la base de données nécessaire pour entraîner et tester nos modèles. Cette base de données contiendra des enregistrements audio des roulements intacts et défectueux.

Ensuite, nous appliquerons notre méthode d'extraction de caractéristiques utilisant les coefficients cepstraux de fréquence Mel (MFCC) aux enregistrements audios de la base de données. Une fois les caractéristiques MFCC extraites, nous procéderons à la construction de réseaux de neurones convolutionnels (CNN) pour identifier et classer les défauts de roulement. Nous formerons plusieurs CNN à l'aide de nos données de formation et évaluerons leurs performances à l'aide de nos données de test. Sur la base des résultats obtenus, nous sélectionnerons le réseau le plus performant pour une utilisation ultérieure dans notre étude expérimentale finale.

Une fois la phase de test terminée sur l'ordinateur, nous passerons à l'exécution sur la carte Raspberry Pi 3b+. Nous allons répliquer les mêmes étapes pour préparer les données, extraire les fonctionnalités MFCC et les déployer sur le modèle choisi sur le Raspberry Pi 3b+. Cela nous permettra de comparer les performances du modèle entre un environnement informatique et un environnement Raspberry Pi 3b+, ce qui est crucial pour évaluer la faisabilité d'une exécution en temps réel sur un appareil embarqué.

# Chapitre I : Généralité sur les pompes centrifuges

## I.1 Introduction

Les pompes dans les industries sont largement utilisées pour des applications différentes et nombreuses et la continuité de la production repose sur le bon état des machines. Cependant, des défauts peuvent toujours survenir et affecter les performances des machines, et ces défauts doivent être détectés à leurs débuts pour éviter tout catas.

La surveillance de l'état de telles machines peut être mise en œuvre en utilisant les méthodes traditionnelles comme l'analyse globale des vibrations et acoustique jusqu'aux méthodes automatiques, où l'intelligence artificielle intervient dans le but de classer précisément les différents défauts de la machine. [1]

Dans ce chapitre, nous démontrons l'utilisation de l'analyse acoustique pour identifier les défauts des roulements à billes dans les pompes centrifuges.

## I.2 Pompes centrifuges

Une pompe est une pièce de machinerie utilisée pour élever l'eau d'un niveau à un autre ou pour transporter un fluide d'un endroit à un autre. L'énergie électrique, qui est ensuite transformée en énergie de pression, est absolument nécessaire au fonctionnement de toute pompe. Il est crucial de choisir la bonne pompe en fonction des exigences du processus et de l'application si vous souhaitez obtenir le rendement le plus élevé. Une pompe centrifuge fondamentale est composée d'une variété de petites pièces d'équipement, y compris un moteur électrique, un arbre, une boîte à garniture, une garniture de presse-étoupe, une palette, une roue et un boîtier. [1]

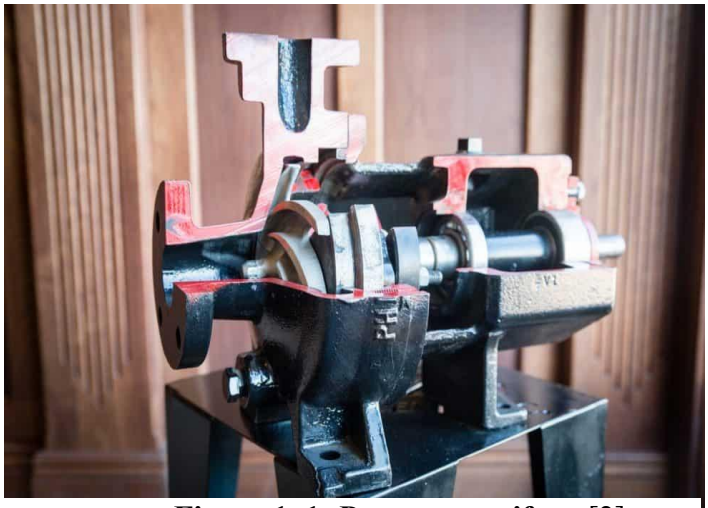


**Figure 1. 1: Pompe centrifuge** [2]

Les pompes centrifuges sont composées de plusieurs éléments essentiels qui travaillent ensemble pour assurer le bon fonctionnement du système.

Les principaux éléments d'une pompe centrifuge : [3]

### I.2.1 Corps de la pompe (ou boîtier)

Il s'agit de la structure externe de la pompe qui contient tous les autres composants. Le corps de la pompe est généralement en métal, tel que la fonte ou l'acier inoxydable, et il est conçu pour résister à la pression et aux forces générées par le fluide en mouvement. [4]



**Figure 1. 2 : Pompe centrifuge avec moteur [4]**

### I.2.2 Roue (ou rotor)

La roue est l'élément rotatif de la pompe centrifuge. Elle est généralement composée de pales ou d'aubes incurvées qui sont fixées à un moyeu central. La rotation de la roue génère la force centrifuge nécessaire pour accélérer le fluide. [5]




**Figure 1. 3 : Les différents types de roues [5]**

**A** : Roues à flux radial, **B** : Roues à flux semi-axiale, **C** : Roues vortex

### I.2.3 Diffuser

Le diffuser est une structure en forme de volute ou de canaux courbés qui entoure la roue de la pompe. Son rôle est de ralentir le fluide accéléré par la roue et de convertir son énergie cinétique en énergie de pression. [6]

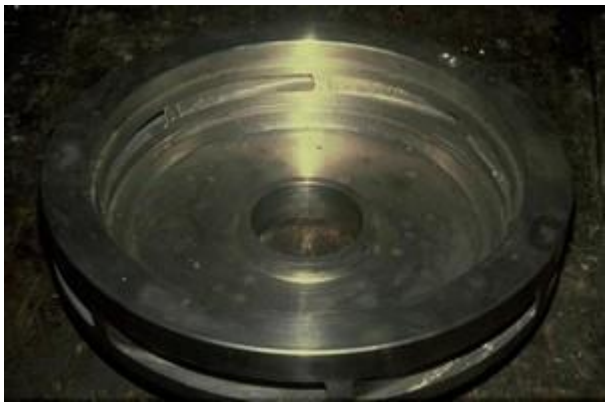


**Figure 1. 4 : Diffuser de pompe centrifuge [6]**

### I.2.4 Arbre

L'arbre est une pièce cylindrique qui transmet la puissance du moteur à la roue de la pompe. Il est généralement fabriqué en acier robuste et est monté de manière à tourner de manière concentrique avec la roue. [6]

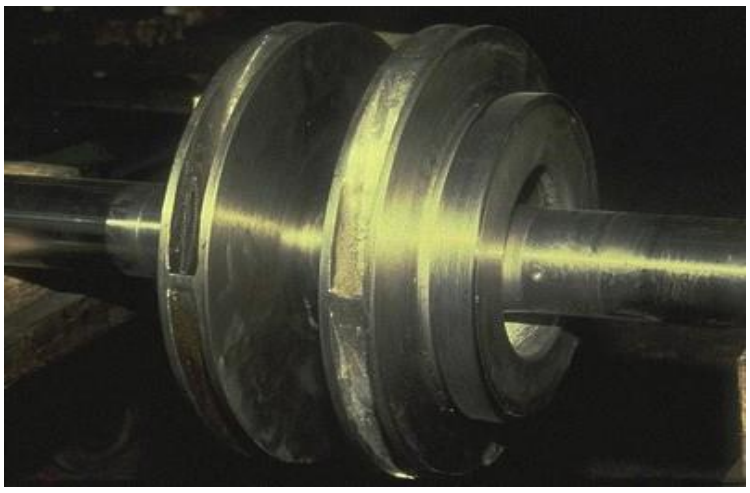


**Figure 1. 5 : Arbre avec double roue radiale [6]**

### I.2.5 Palier (ou roulement)

Les paliers ou roulements sont des éléments qui soutiennent et guident l'arbre de la pompe. Ils réduisent les frottements et les vibrations, assurant ainsi un fonctionnement en douceur de l'arbre. Les paliers peuvent être à billes, à rouleaux ou à palier lisse, en fonction des exigences de la pompe.[7]



**Figure 1. 6 : Roulement à billes à gorge profonde**

### I.2.6 Garniture mécanique

Le Garniture mécanique agit comme un clapet anti-retour et un palier coulissant. La fonction évidente est celle d'un clapet anti-retour pour empêcher le liquide sous pression de s'échapper de la pompe ou d'aspirer de l'air dans la pompe dans des conditions de vide.[8]

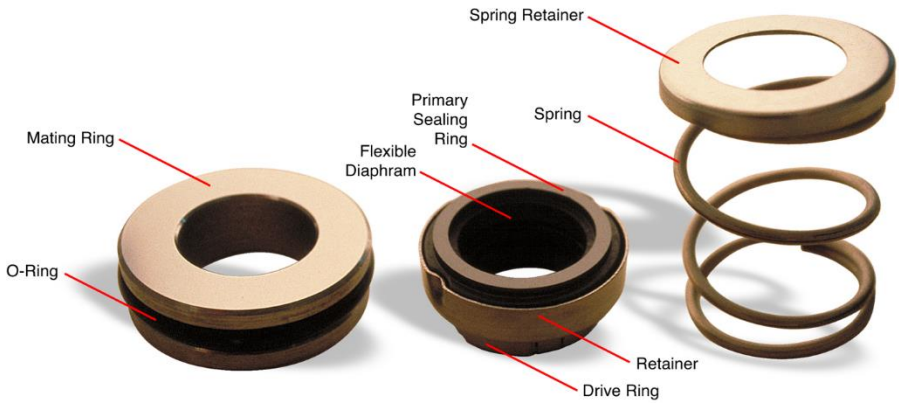



**Figure 1. 7 : La garniture mécanique [8]**

### I.2.7 Moteur

Le moteur fournit la puissance nécessaire pour entraîner la pompe centrifuge. Il peut être électrique, diesel ou hydraulique, en fonction de l'application et des besoins spécifiques. Le moteur est généralement monté sur le boîtier de la pompe et est connecté à l'arbre de la pompe par l'intermédiaire d'un accouplement.[9]



**Figure 1. 8 : Moteur de pompe [9]**

## I.3 Utilisation

Les pompes centrifuges sont utilisées dans de nombreuses applications industrielles et commerciales en raison de leur capacité à transférer efficacement des fluides : [10, 11]

### I.3.1 Approvisionnement en eau

Utilisées pour l'approvisionnement en eau dans les systèmes de distribution d'eau potable, les stations de pompage d'eau, les tours de refroidissement et les systèmes d'irrigation agricole. Elles sont capables de pomper de grandes quantités d'eau sur de longues distances.

### I.3.2 Traitement des eaux usées

Transfert des eaux usées brutes, des boues et des effluents traités. Elles peuvent également être utilisées pour le pompage des eaux pluviales dans les systèmes de drainage.

### I.3.3 Industrie chimique et pétrolière

Transfert de produits chimiques, de liquides inflammables, de carburants et d'autres substances corrosives. Elles sont conçues pour résister aux produits chimiques agressifs et aux conditions de travail extrêmes.

### I.3.4 Industrie alimentaire et pharmaceutique

Transfert de liquides alimentaires, de produits laitiers, de boissons, de produits pharmaceutiques et d'ingrédients. Elles doivent respecter des normes strictes en matière d'hygiène et de sécurité alimentaire.

### I.3.5 Industrie pétrochimique

Transfert de produits pétrochimiques, de gaz, de lubrifiants et de produits chimiques. Elles sont adaptées aux conditions de travail à haute pression et à haute température.

### I.3.6 Industrie minière

Pompage des boues, des effluents miniers, de l'eau de mine et des fluides de traitement. Elles doivent être robustes pour résister aux particules solides abrasives présentes dans les applications minières.

## I.4 Fonctionnement

Basé sur le principe de la force centrifuge générée par la rotation d'une roue ou d'un rotor à l'intérieur de la pompe. Les étapes clés du fonctionnement d'une pompe centrifuge :[12]

### I.4.1 Aspiration

Le processus commence lorsque la pompe est mise en marche. Le fluide à pomper est aspiré à travers l'orifice d'admission de la pompe, généralement situé au centre du boîtier de la pompe.

### I.4.2 Accélération

Une fois le fluide aspiré, il entre en contact avec la roue ou le rotor de la pompe. La roue est munie de pales ou d'aubes qui sont inclinées de manière à accélérer le fluide lorsque la roue tourne. Lorsque le fluide entre en contact avec les pales, il est poussé vers l'extérieur et accéléré en raison de la force centrifuge générée par la rotation de la roue.

### I.4.3 Conversion de l'énergie cinétique en énergie de pression

En accélérant le fluide, la pompe convertit son énergie cinétique en énergie de pression. Lorsque le fluide est poussé vers l'extérieur par les pales de la roue, il entre dans une zone appelée le diffuser. Le diffuser est une structure en forme de volute ou de canaux courbés qui entoure la roue. En passant à travers le diffuser, le fluide ralentit et son énergie cinétique est transformée en énergie de pression, augmentant ainsi la pression du fluide.

### I.4.4 Évacuation du fluide

Une fois que le fluide a traversé le diffuser, il est expulsé par la sortie de la pompe à haute pression. Le fluide peut ensuite être dirigé vers le système ou l'application souhaitée, tel que le transport de l'eau vers un réseau de distribution ou l'acheminement de produits chimiques dans un processus industriel.

## I.5 Les Avantages

Les pompes centrifuges offrent plusieurs avantages par rapport aux pompes alternatives (ou pompes volumétriques) : [4, 13, 71]

### I.5.1 Débit régulier

Les pompes centrifuges fournissent un débit régulier et constant, ce qui est important dans de nombreuses applications industrielles. Les pompes alternatives, en revanche, génèrent un débit pulsatoire en raison de leur mécanisme de fonctionnement, ce qui peut entraîner des variations de pression et des vibrations dans le système.

### I.5.2 Pression de fonctionnement élevée

Les pompes centrifuges sont généralement capables de générer des pressions plus élevées que les pompes alternatives. Elles sont donc plus adaptées aux applications nécessitant un transfert de fluide à haute pression.

### I.5.3 Capacité à traiter des fluides visqueux

Les pompes centrifuges sont plus efficaces pour pomper des fluides visqueux, tels que les liquides épais ou contenant des particules en suspension. Les pompes alternatives peuvent rencontrer des difficultés à pomper ces types de fluides, en raison de leur mécanisme de fonctionnement et de leur sensibilité accrue aux obstructions.

### I.5.4 Maintenance réduite

Les pompes centrifuges sont généralement plus faciles à entretenir que les pompes alternatives. Elles ont moins de pièces mobiles et ne nécessitent pas de réglages fréquents. Les pompes alternatives peuvent nécessiter un entretien plus fréquent en raison de l'usure des pièces de mouvement, telles que les pistons ou les soupapes.

### I.5.5 Taille et poids réduits

Les pompes centrifuges ont tendance à être plus compactes et plus légères que les pompes alternatives de même capacité. Cela facilite leur installation et réduit les coûts associés à l'espace et au transport.

### I.5.6 Coût initial inférieur

Dans de nombreux cas, les pompes centrifuges sont moins chères à l'achat que les pompes alternatives. Elles offrent donc une solution économique pour les applications nécessitant un débit élevé.

## I.6 Analyse acoustique des machines tournantes

Cette méthode utilisée pour évaluer et diagnostiquer l'état de fonctionnement des machines en se basant sur les signaux sonores qu'elles produisent. Elle peut fournir des informations précieuses sur les défauts, les vibrations et les problèmes de fonctionnement des machines tournantes, telles que les pompes centrifuges, les moteurs, les turbines, les compresseurs, etc.[15]

### I.6.1 Les aspects importants de l'analyse acoustique des machines tournantes

#### I.6.1.1 Acquisition des signaux

Les signaux acoustiques sont captés à l'aide de microphones ou de capteurs de vibration placés stratégiquement sur la machine tournante. Les signaux sont enregistrés et analysés pour détecter les variations de fréquence, d'amplitude et de formes d'onde.

#### I.6.1.2 Analyse spectrale

L'analyse spectrale est une technique couramment utilisée en analyse acoustique des machines tournantes. Elle consiste à convertir le signal acoustique du domaine temporel vers le domaine fréquentiel à l'aide de la transformée de Fourier. Cela permet d'identifier les composantes fréquentielles présentes dans le signal et d'identifier les fréquences caractéristiques associées aux défauts ou aux problèmes de la machine.

#### I.6.1.3 Détection des signatures de fréquence

Chaque défaut ou problème de la machine tournante génère des signatures de fréquence spécifiques dans le signal acoustique. Par exemple, un désalignement des composants peut produire des fréquences latérales, tandis qu'un déséquilibre peut générer des fréquences harmoniques. En analysant les pics de fréquence dans le spectre, il est possible de détecter et d'identifier les défauts.

#### I.6.1.4 Comparaison avec les valeurs de référence

Les analyses acoustiques des machines tournantes impliquent souvent la comparaison des signaux mesurés avec des valeurs de référence. Ces valeurs peuvent provenir de machines similaires en bon état de fonctionnement ou de modèles mathématiques. Les différences entre les signaux mesurés et les valeurs de référence peuvent indiquer des anomalies ou des problèmes.

#### I.6.1.5 Diagnostic des défauts

En analysant les caractéristiques des signaux acoustique.

### I.6.2 Définition d'une acoustique :

L'acoustique est une branche de la physique qui se concentre sur l'étude des propriétés et du comportement des sons. Elle englobe l'analyse, la mesure et la compréhension des phénomènes liés aux ondes sonores, tels que leur production, leur propagation, leur interaction avec l'environnement et leur perception par les êtres vivants. Les domaines d'application de l'acoustique sont nombreux et comprennent l'ingénierie du son, l'architecture, la musique, la médecine, la communication, etc. Son objectif est de comprendre et de contrôler les phénomènes sonores afin de les utiliser de manière optimale dans divers contextes, ce qui implique souvent la conception et l'optimisation d'équipements acoustiques. [16]

#### I.6.2.1 Fréquence :

La fréquence d'un son correspond au nombre de cycles vibratoires complets par seconde et est mesurée en hertz (Hz). Elle est perçue comme la hauteur du son, avec des fréquences plus élevées correspondant à des sons plus aigus et des fréquences plus basses correspondant à des sons plus graves.

#### I.6.2.2 Amplitude

L'amplitude d'un son représente l'intensité ou la force du son et est mesurée en décibels (dB). L'amplitude est liée à la puissance ou à l'énergie du son et détermine son volume perçu, avec des amplitudes plus élevées correspondant à des sons plus forts.

#### I.6.2.3 Durée

La durée d'un son fait référence à la période pendant laquelle il persiste. Certains sons peuvent être brefs, comme un claquement de doigts, tandis que d'autres peuvent être prolongés, comme une note de musique tenue.

#### I.6.2.4 Direction et localisation

L'acoustique permet également de déterminer la direction et la localisation d'une source sonore. La localisation peut être déterminée par des différences de temps d'arrivée du son aux deux oreilles, ainsi que par des différences d'intensité et de spectre sonore.

#### I.6.2.5 Phase

La phase d'un son se réfère à la position relative des oscillations sonores par rapport à un point de référence. La phase peut avoir un impact sur l'interférence des ondes sonores et la perception du son.

## I.7 Les roulements

Les roulements sont des dispositifs mécaniques qui ont pour fonction de supporter et de centrer les arbres de machines, de diminuer la friction absorbée, et de permettre le transfert de mouvement en facilitant la rotation d'un élément par rapport à un autre. Les roulements remplissent deux fonctions majeures : ils transmettent le mouvement en ce sens qu'ils supportent et guident les composants dont la rotation s'effectue l'un par rapport à l'autre, et ils transmettent des forces.

Les roulements sont utilisés dans de nombreuses applications industrielles, telles que les machines-outils, les moteurs, les véhicules, les équipements électriques, etc. Les roulements peuvent être à billes ou à rouleaux.[17, 18]

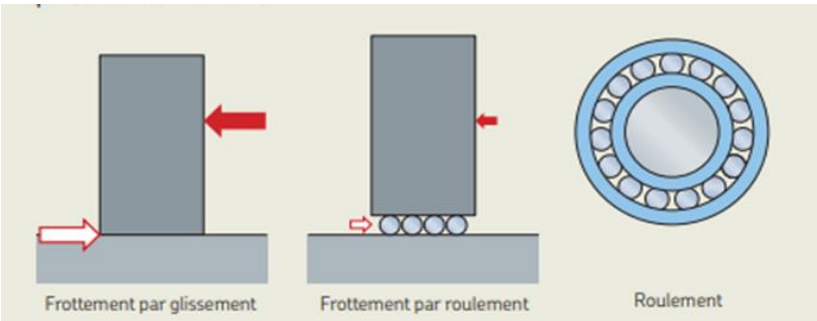



**Figure 1. 9 : frottement par glissement et roulements [17]**

### I.7.1 Roulements à billes et à rouleaux

Deux types d'éléments roulants de base permettent de distinguer deux familles de roulements de base :

- La bille → roulement à billes
- Le rouleau → roulement à rouleaux

Les billes et les rouleaux sont en contact avec les pistes de manière différente.[18]

Les billes conduisent à un contact ponctuel avec les pistes des bagues **A** Avec l'augmentation des charges agissant sur le roulement, le point de contact devient une zone elliptique. La petite zone de contact produit un faible frottement, ce qui permet aux roulements à billes d'accepter des vitesses élevées mais, en revanche, limite leur capacité de charge.

Les rouleaux conduisent à un contact linéaire avec les pistes des bagues **B** Lorsque la charge agissant sur le roulement augmente, la ligne de contact prend une forme quelque peu rectangulaire. En raison de la zone de contact plus large et, par conséquent, du frottement plus

élevé, un roulements roulement à rouleaux peut accepter des charges plus importantes, mais des vitesses plus faibles, qu'un roulement à billes de la même taille.



A B

**Figure 1. 10 : les roulements [17]**

**A : Roulements à billes, B : Roulements à rouleaux**

### I.7.2 Structure mécanique d'un roulement

La structure d'un roulement mécanique se compose de plusieurs composants, dont :

#### I.7.2.1 Bague intérieure :

C'est la partie du roulement qui tourne avec l'arbre.

#### I.7.2.2 Bague extérieure :

C'est la partie fixe du roulement qui est montée dans le logement.

### I.7.3 Éléments roulants :

Ce sont les billes ou les rouleaux qui roulent entre les anneaux intérieur et extérieur, permettant une rotation en douceur.

#### I.7.3.1 Cage

Il s'agit d'un composant qui sépare et guide les éléments roulants pour éviter qu'ils n'entrent en collision les uns avec les autres.

#### I.7.3.2 Joints ou boucliers

Ce sont des composants optionnels qui protègent le roulement des contaminants et retiennent la lubrification.

La conception d'un roulement peut varier en fonction de son utilisation prévue, et il existe différents types de roulements, tels que les roulements à billes, les roulements à rouleaux et les roulements à aiguilles. Les roulements peuvent supporter des charges radiales et axiales, et ils sont utilisés dans diverses applications, y compris les machines, les véhicules et les équipements.[16]

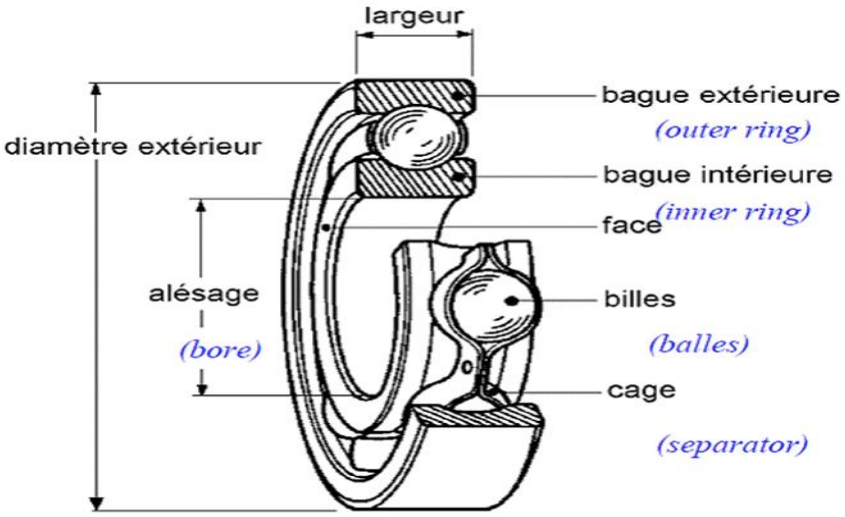



**Figure 1. 11 : Composantes d'un palier [18]**

### I.7.4 Causes des défauts des roulements à billes

#### I.7.4.1 L'usure

L'usure des roulements peut être causée par plusieurs facteurs tels que la charge, la vitesse, la température, la contamination, l'alignement incorrect, le manque de lubrification, etc. Pour prolonger la durée de vie des roulements, il est important de les entretenir régulièrement en vérifiant leur état et en remplaçant ceux qui sont endommagés ou usés. Il est également important de choisir des roulements de qualité adaptés aux conditions de fonctionnement pour assurer une longue durée de vie et une performance optimale. [19, 20]

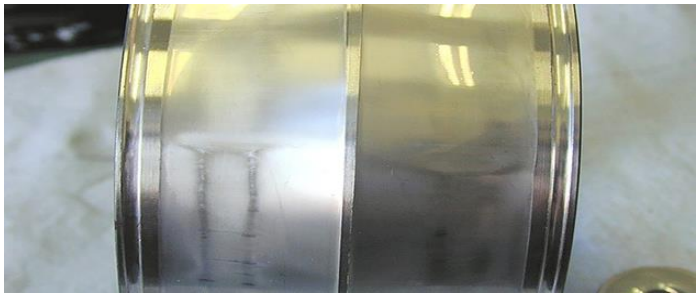


**Figure 1. 12 : L'usure abrasive dans une bague intérieure [19]**

#### I.7.4.2 Dentelure

La qualité de la denture des roulements est importante pour assurer une transmission de couple fluide et sans à-coups, ainsi qu'une longue durée de vie des roulements. Les défauts de denture, tels que les frottements, les usures et les cassures, peuvent entraîner des pannes prématurées du roulement et des problèmes de fonctionnement pour le mécanisme où il est

utilisé. Pour assurer une bonne qualité de denture, il est important de choisir des roulements de qualité et de les entretenir régulièrement pour éviter tout dommage. [19, 20]

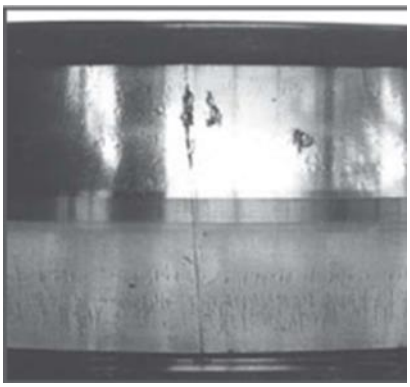


**Figure 1. 13 : Défaut de dentelure sur la bague interne d'un roulement à bille [19]**

### I.7.4.3 Bavures

Les bavures sont des petites projections de matière qui se forment sur les bords des pièces lorsqu'elles sont usinées. Les bavures peuvent causer des problèmes de fonctionnement pour les roulements en interférant avec leur mouvement et en réduisant leur durée de vie. [19, 20]

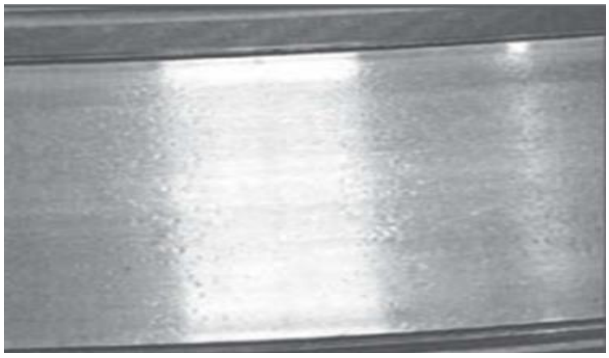


**Figure 1. 14 : Défaut de bavure sur un roulement à billes [19]**

### I.7.4.4 Ecaillage

L'écaillement peut se produire sur la surface des roulements lorsqu'ils sont soumis à des charges excessives ou à des chocs répétés. Cela peut entraîner une perte de matière sur la surface du roulement, ce qui peut réduire la durée de vie du roulement et causer des problèmes de fonctionnement pour le mécanisme où il est utilisé.

L'écaillement peut être causé par plusieurs facteurs, notamment une lubrification inadéquate, une installation incorrecte, des chocs ou des vibrations excessives, une surcharge ou une utilisation dans des conditions environnementales extrêmes. [19, 20]



**Figure 1. 15 : Défaut d'écaillage sur la bague interne d'un roulement à billes [19]**

### I.7.4.5 Corrosion

La corrosion des roulements peut se produire lorsqu'ils sont exposés à des environnements corrosifs, tels que des fluides corrosifs, des atmosphères salines ou humides, ou des températures élevées. La corrosion peut causer des dommages importants aux roulements, y compris la formation de piqûres, la rugosité de la surface, la fissuration et la perte de matière. [19, 20]

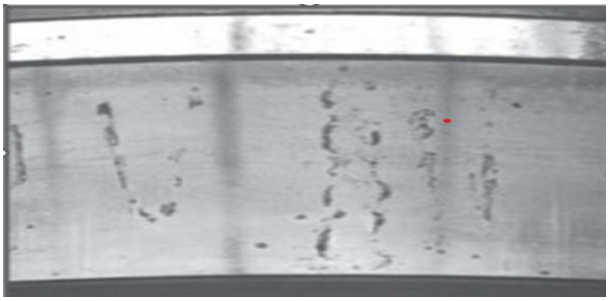


**Figure 1. 16 : Corrosion sur d'un roulement à billes [19]**

### I.7.4.6 Coup, fissures, Cassures

Les coups, les fissures et les cassures sont des signes évidents de dommages importants sur les roulements. Ils peuvent être causés par des surcharges, des chocs, des vibrations excessives, une installation incorrecte, une usure importante ou une utilisation dans des conditions environnementales extrêmes. [19, 20]



**Figure 1. 17 : Cassures d'un roulement à billes [19]**

### I.7.4.7 Passage de courant

Le passage de courant dans les roulements, également connu sous le nom de courant vagabond, est un phénomène qui peut se produire dans les systèmes de roulements qui fonctionnent avec des moteurs électriques. Ce phénomène peut causer des dommages aux roulements et réduire leur durée de vie.

Le passage de courant se produit lorsque des courants électriques traversent les roulements, créant des arcs électriques qui peuvent endommager les surfaces de roulement et provoquer la formation de microfissures. [19, 20]

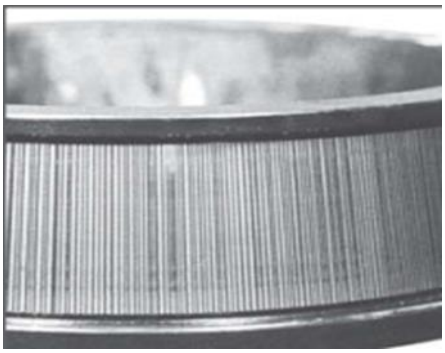


**Figure 1. 18 : Passage de courant d'un roulement à bill [19]**

### I.7.5 Les types de défauts les roulements

Les défauts des roulements peuvent être classés en plusieurs types, notamment :[19]

### I.7.5.1 Défaut au niveau de la bague externe

L'image d'un défaut de type écaillage affectant la bague extérieure d'un roulement ressemble à un peigne de lignes dont le pas correspond à la fréquence du défaut.



**Figure 1. 19 : Défaut au niveau de la bague externe**

### I.7.5.2 Défaut de type écaillage affectant la bague interne

Une image pour un défaut de type écaillage affectant la bague interne de roulement est un peigne de raies.



**Figure 1. 20 : Défaut de type écaillage affectant la bague interne**

### I.7.5.3 Défaut de type écaillage sur un élément roulant

Un défaut de type écaillage sur un élément roulant (bille, rouleau ou aiguille).



**Figure 1. 21 : Défaut de type écaillage sur un élément roulant**

### I.7.5.4 Défauts de type déversement de bague

Le défaut de type déversement de bague extérieure



**Figure 1. 22 : Défaut de type déversement de bague**

### I.7.6 Fréquences caractéristiques

Des pistes de roulement se dégradent par fatigue en conduisant aux écaillages puis à la ruine du matériau. Lors de la mise en rotation, un train d'impulsion est généré par le défaut à une fréquence bien définie que l'on appelle « fréquence caractéristique » du défaut de roulement. [19]

On peut considérer quatre fréquences caractéristiques :

La fréquence de passage d'un élément roulant sur un défaut de bague extérieur :

$$BPFO = \frac{n\,F}{2}\left(1 - \frac{d}{D_m}\cos\alpha\right) \quad \text{Éq 1}$$

La fréquence de passage d'un élément roulant sur un défaut de bague intérieure :

$$BPFI = \frac{n\,F}{2}\left(1 + \frac{d}{D_m}cos\alpha\right) \quad \text{Éq 2}$$

La fréquence de passage d'un défaut d'élément roulant sur la bague intérieure ou extérieure :

$$BPF = \frac{D_m\,F}{2\,d}\left(1 - \frac{d^2}{D_m}cos^2\alpha\right) \quad \text{Éq 3}$$

La fréquence de passage d'un défaut de cage :

$$CF = \frac{F}{2}\left(1 - \frac{d}{D_m}cos\alpha\right) \quad \text{Éq 4}$$

Avec :

N : la fréquence de rotation en [Hz] ;

n : le nombre des billes ;

$D_m$: Le diamètre moyen en [mm] ;

$\alpha$: L'angle de contact ;

d : le diamètre de la bille

## I.8 Conclusion

Dans ce chapitre, nous avons présenté la pompe centrifuge, y compris ses éléments, ses utilisations, son fonctionnement et ses avantages. De plus, nous avons discuté de l'importance de l'analyse acoustique dans l'identification des problèmes de roulement dans les pompes centrifuges à machines tournantes.

# Chapitre II : Intelligence Artificielle et MFCC

## II.1 Introduction

L'intelligence artificielle (IA) et l'apprentissage en profondeur (Deep Learning) sont des domaines de recherche et de développement qui ont un impact significatif sur les systèmes embarqués.

L'intelligence artificielle se réfère à la capacité des machines à simuler l'intelligence humaine, en utilisant des algorithmes et des modèles mathématiques pour accomplir des tâches spécifiques. L'apprentissage en profondeur est une branche de l'IA qui se concentre sur l'utilisation de réseaux de neurones artificiels à plusieurs couches pour apprendre à partir de grandes quantités de données.

Dans le domaine des systèmes embarqués, l'intelligence artificielle et le deep learning ont ouvert de nouvelles possibilités et ont amélioré les performances de ces systèmes. Par exemple, les voitures autonomes utilisent des techniques d'apprentissage en profondeur pour détecter et reconnaître les objets sur la route, permettant ainsi une conduite autonome.

De plus, les systèmes embarqués utilisent également l'IA et le deep learning pour la détection d'anomalies, la classification des données, la reconnaissance vocale, la vision par ordinateur et bien d'autres applications. Ces techniques permettent aux systèmes embarqués d'être plus autonomes, intelligents et capables de prendre des décisions en temps réel.

## II.2 Deep Learning

Deep learning est un sous-ensemble de machine learning qui utilise des réseaux de neurones à trois couches ou plus pour simuler la structure et la fonction du cerveau humain. C'est une technique qui apprend aux ordinateurs à apprendre par l'exemple et à extraire des fonctionnalités de niveau supérieur à partir d'une entrée brute. Deep learning en profondeur est utilisé dans diverses applications, notamment le traitement d'images, la reconnaissance vocale et l'analyse prédictive. Les algorithmes de deep learning en profondeur sont empilés dans une hiérarchie de complexité et d'abstraction croissantes, ce qui les rend plus efficaces et plus faciles à automatiser l'analyse prédictive. Diverses méthodes peuvent être utilisées pour créer de solides modèles de deep learning en profondeur, notamment la dégradation du taux d'apprentissage, l'apprentissage par transfert, la formation à partir de zéro et l'abandon. Deep learning en profondeur est un élément important de la science des données, qui comprend les statistiques et la modélisation prédictive. [21]

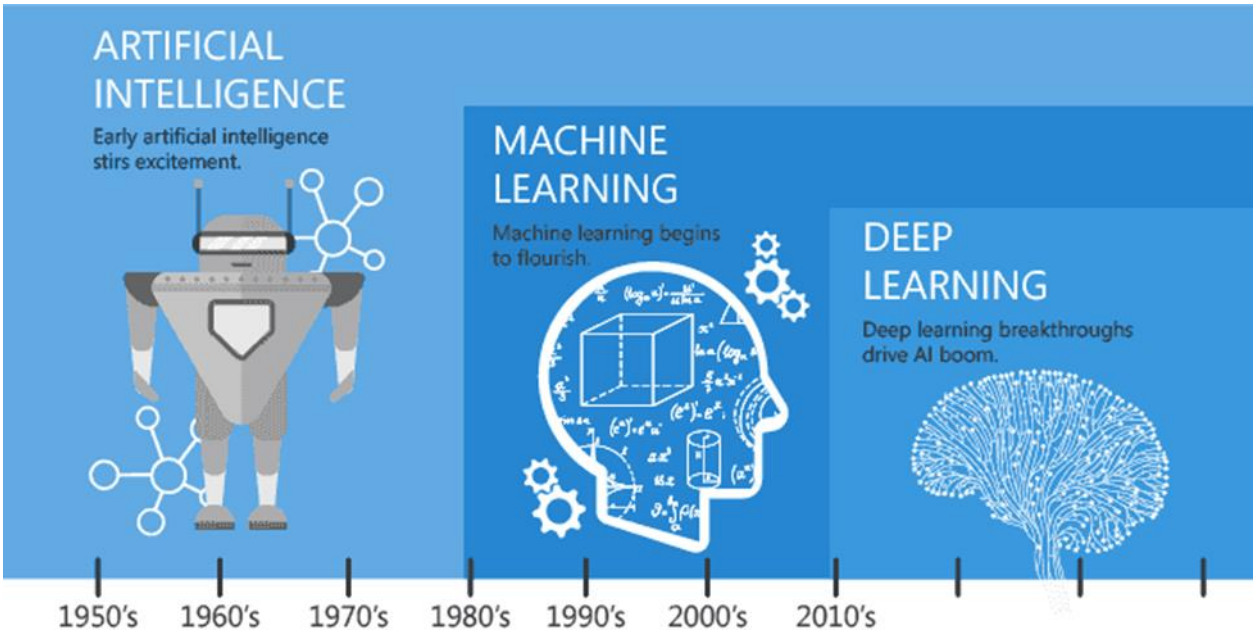



**Figure 2. 1 : les domaines de l'intelligence artificielle [29]**

## II.3 Deep Learning vs. Machine learning

Le Deep Learning et le Machine Learning sont deux sous-domaines de l'intelligence artificielle (IA) qui se concentrent sur l'apprentissage automatique à partir de données. Cependant, ils diffèrent dans leurs approches et leurs applications.

Le Machine Learning utilise des algorithmes pour analyser des données, apprendre des modèles et des relations, puis utiliser ces modèles pour faire des prédictions ou prendre des décisions sur de nouvelles données. Il existe plusieurs types d'algorithmes de Machine Learning, tels que la régression, la classification et le clustering. Le Machine Learning nécessite généralement une intervention humaine pour l'ingénierie des caractéristiques et la correction des erreurs.

Le Deep Learning, en revanche, est un sous-ensemble du Machine Learning qui utilise des réseaux de neurones artificiels avec plusieurs couches pour analyser des modèles complexes et des relations dans les données. Inspiré par la structure et le fonctionnement du cerveau humain, le Deep Learning a été couronné de succès dans diverses tâches, telles que la reconnaissance d'images, la reconnaissance vocale et la traduction automatique. Contrairement au Machine Learning, le Deep Learning est capable d'apprendre automatiquement des caractéristiques à partir de données brutes sans intervention humaine.

En résumé, le Deep Learning est un sous-ensemble plus avancé du Machine Learning qui utilise des réseaux de neurones profonds pour résoudre des problèmes plus complexes.

Tandis que le Machine Learning repose sur des algorithmes plus simples et nécessite souvent une intervention humaine pour l'ingénierie des caractéristiques. [22]

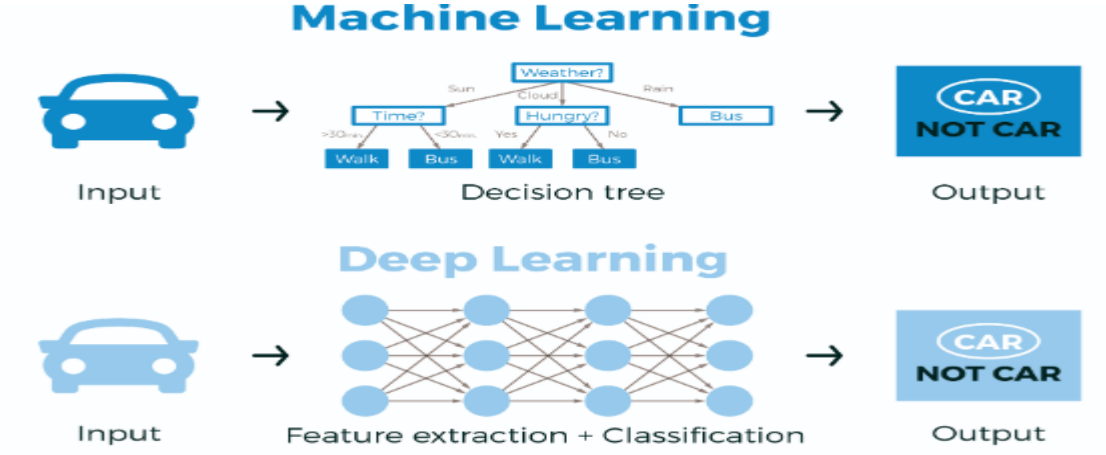



**Figure 2. 2: example Deep learning vs. Machine learning [28]**

## 2.1 Type d'apprentissage

Les algorithmes de Deep Learning se répartissent en deux types principaux, apprentissage supervisé et apprentissage non-supervisé :

### II.3.1 Apprentissage supervisé

L'apprentissage supervisé est la méthode d'apprentissage la plus utilisée en Machine Learning et en Deep Learning. Dans le contexte de l'intelligence artificielle, l'apprentissage supervisé consiste à surveiller l'apprentissage de la machine en lui présentant des exemples de ce qu'elle doit effectuer. Les données d'entrée sont étiquetées et le modèle est entraîné à prédire une sortie en fonction de ces étiquettes. Les réseaux de neurones profonds sont utilisés pour apprendre des représentations hiérarchiques des données, ce qui permet de capturer des caractéristiques complexes et non linéaires des données. Les applications de l'apprentissage supervisé en Deep Learning sont nombreuses, telles que la reconnaissance vocale, la vision par ordinateur, la classification et la régression. [23]

### II.3.2 Apprentissage non-supervisé

L'apprentissage non supervisé est une méthode d'apprentissage automatique qui utilise des algorithmes pour analyser et regrouper des ensembles de données non étiquetés. Contrairement à l'apprentissage supervisé, qui repose sur des données étiquetées pour entraîner des modèles prédictifs, l'apprentissage non supervisé découvre des modèles cachés ou des groupements de données sans nécessiter d'intervention humaine.

Dans le contexte du Deep Learning, l'apprentissage non supervisé peut être utilisé pour des tâches telles que la classification, l'association et la réduction de la dimensionnalité.

Les algorithmes d'apprentissage non supervisé sont souvent utilisés en combinaison avec des algorithmes d'apprentissage supervisé pour améliorer les performances des modèles prédictifs. [24]

## II.4 Détection des anomalies

La détection des anomalies dans la donnée est une méthode utilisée pour identifier les points de données, événements ou observations qui diffèrent significativement de la majorité des données et ne correspondent pas à une notion bien définie de comportement normal. Les anomalies peuvent être causées par des erreurs de données, des erreurs expérimentales ou humaines, ou des données corrompues. La détection des anomalies est utilisée dans de nombreux domaines, notamment la cyber sécurité, la médecine, la vision par ordinateur, les statistiques, les sciences neurologiques, l'application de la loi et la fraude financière. Il existe plusieurs algorithmes de détection des anomalies, tels que les techniques statistiques simples comme la moyenne, la médiane et les quantiles, et des algorithmes plus avancés tels que l'apprentissage automatique non supervisé. La détection des anomalies est une étape importante dans l'analyse des données pour identifier les points de données qui ne correspondent pas au comportement normal et qui peuvent affecter les performances du modèle. [25]

## II.5 Les réseaux de neurones

### II.5.1 Neurone biologique

Un neurone biologique est une cellule électriquement excitable qui émet des signaux électriques appelés potentiels d'action et communique avec d'autres cellules via des connexions spécialisées appelées synapses qui utilisent généralement d'infimes quantités de neurotransmetteurs chimiques pour transmettre le signal électrique de l'axone terminal à la dendrite du prochain neurone. Un neurone typique se compose d'un corps cellulaire (soma), de dendrites et d'un seul axone. Les dendrites sont des fibres à ramification courte qui convertissent les informations chimiques d'autres neurones ou cellules réceptrices en signaux électriques, tandis que les axones sont des fibres allongées qui transmettent des signaux électriques aux régions terminales pour communiquer avec d'autres neurones ou effecteurs. Le soma est une structure compacte, et l'axone et les dendrites sont des filaments extrudés du soma. Les neurones sont hautement spécialisés pour le traitement et la transmission des signaux cellulaires et ont une grande variété de formes, de tailles et de propriétés électrochimiques. [26]

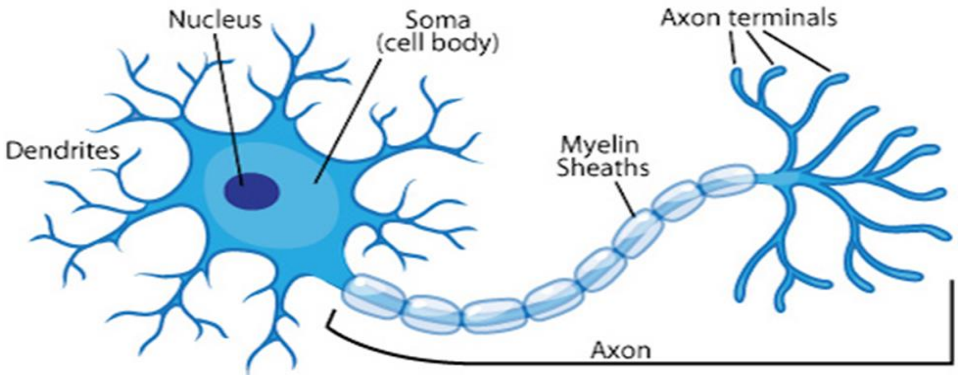



**Figure 2. 3 : Neurone biologique [26]**

### II.5.2 Des réseaux de neurones artificiels

Les réseaux de neurones artificiels (ANN) sont des modèles informatiques inspirés des réseaux de neurones biologiques. Ils sont constitués de couches nodales, comprenant une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. Chaque nœud, ou neurone artificiel, se connecte à un autre et possède un poids et un seuil associés. Les ANN sont utilisées pour imiter le comportement du cerveau humain et résoudre des problèmes courants dans les domaines de l'intelligence artificielle, de l'apprentissage automatique et de l'apprentissage en profondeur. [27]

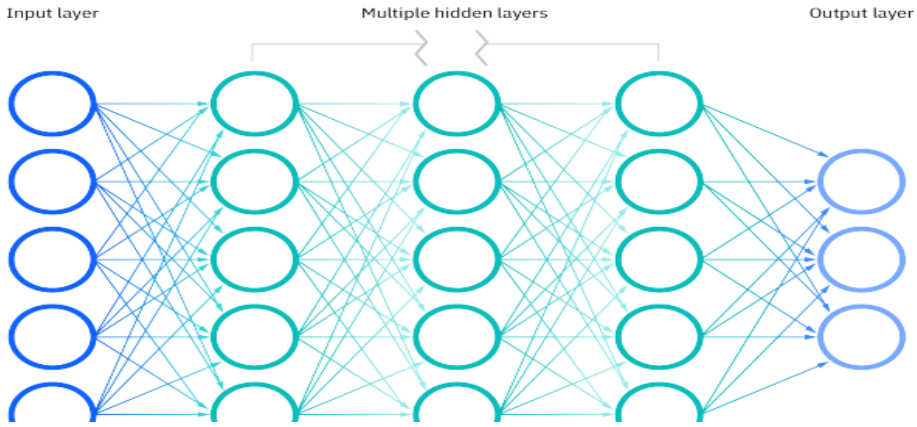



**Figure 2. 4 : réseaux de neurones artificiels [27]**

## II.6 Perceptron

### II.6.1 Un perceptron

Un Perceptron est un algorithme utilisé pour l'apprentissage supervisé de classificateurs binaires. Les classificateurs binaires décident si une entrée, généralement représentée par une série de vecteurs, appartient à une classe spécifique.

En bref, un perceptron est un réseau de neurones à une seule couche. Ils se composent de quatre parties principales, notamment les valeurs d'entrée, les pondérations et le biais, la somme nette et une fonction d'activation. [30]

### II.6.2 Fonctionne Un Perceptron

Le processus commence en prenant toutes les valeurs d'entrée et en les multipliant par leurs poids. Ensuite, toutes ces valeurs multipliées sont additionnées pour créer la somme pondérée. La somme pondérée est ensuite appliquée à la fonction d'activation, produisant la sortie du perceptron. La fonction d'activation joue le rôle essentiel de s'assurer que la sortie est mappée entre les valeurs requises telles que (0,1) ou (-1,1). Il est important de noter que le poids d'une entrée est indicatif de la force d'un nœud. De même, la valeur de biais d'une entrée donne la possibilité de décaler la courbe de la fonction d'activation vers le haut ou vers le bas. [30]

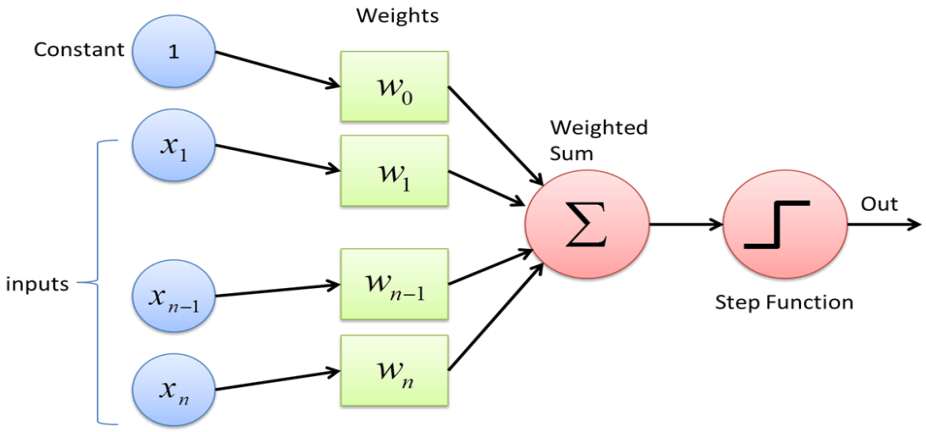



**Figure 2. 5 : un perceptron [30]**

### II.6.3 Les fonctions d'activation

Les fonctions d'activation sont des éléments clés des réseaux de neurones profonds (deep Learning) et jouent un rôle crucial dans la conception des réseaux de neurones. Elles sont utilisées pour décider si un neurone s'active ou non, en supprimant les neurones dont les entrées ne sont pas significatives pour l'application globale du réseau de neurones. Le choix de la fonction d'activation dépend du type de problème de prédiction et de la couche du réseau de neurones (cachée ou de sortie).[31]

#### II.6.3.1 Types de fonctions d'activation :

- Fonction d'étape binaire
- Fonction d'activation linéaire
- Fonctions d'activation non linéaires (un certain nombre)

#### II.6.3.1.1 Fonction d'étape binaire

Il s'agit de la fonction d'activation là moins utilisée et vous n'en avez peut-être même pas entendu parler. La fonction d'activation d'étape binaire est un simple classificateur de seuil. Si l'entrée prise par la fonction d'étape binaire est supérieure à la condition de seuil, le neurone sera activé, ce qui signifie que la sortie du neurone sera transmise avec succès à la ou aux couches suivantes. Si ce n'est pas le cas, le neurone ne s'activera pas et deviendra essentiellement inutile pour le réseau neuronal global. [31]

$$\boldsymbol{f(x) = 0 \qquad \text{pour } x < 0 \text{ ou } 1 \text{ pour } x \geq 0} \qquad \text{Éq 5}$$

#### II.6.3.1.2 Fonction d'activation linéaire

À partir de l'équation, il devient tout à fait évident que l'activation est directement proportionnelle à l'entrée reçue par la fonction. Le gradient ici restera le même pour chaque étape de calcul à travers le réseau et ne change pas. Pour cette raison, les fonctions d'activation linéaire doivent être utilisées pour les applications de régression simples où la non-linéarité n'est pas requise [31] :

$$\boldsymbol{f(x) = x} \qquad \text{Éq 6}$$

#### II.6.3.1.3 Fonctions d'activation non linéaires

Il existe un tas de fonctions d'activation non linéaires utiles qui peuvent être utilisées pour différentes applications [31] :

- **Fonction d'activation Relu (unité linéaire rectifiée) :**

$$\boldsymbol{f(x) = max(0, x)} \qquad \text{Éq 7}$$

Il s'agit d'une fonction linéaire par morceaux qui produira l'entrée directement si elle est positive, sinon, elle produira zéro. On parle donc d'unité linéaire "rectifiée". Il s'agit de la fonction d'activation la plus couramment utilisée pour les couches cachées entre les couches d'entrée et de sortie d'un réseau, car elle est simple à mettre en œuvre et se traduit souvent par de meilleures performances.

Cependant, puisque toutes les entrées négatives sont mappées sur 0, le gradient de cette activation devient également nul. De ce fait, lors de la rétro propagation, les poids de certains neurones peuvent ne pas être mis à jour, ce qui les rend inutiles. C'est ce qu'on appelle le "problème Dying Relu".

**Fonction d'activation Relu qui fuit :**

$$\boldsymbol{f(x) = 1(x < 0)(\alpha x) + 1(x \geq 0)(x)} \qquad \text{Éq 8}$$

**α:** une petite constante

La fonction d'activation Leaky Relu tente d'atténuer le problème Dying Relu qui se produit dans Relu avec une petite pente positive qui ne conduit pas à un problème de gradient nul. Ainsi, lors de la rétro propagation, les neurones de la région négative sont également pris en compte.

D'autres fonctions d'activation rarement utilisées basées sur Relu incluent :

- L'unité linéaire exponentielle paramétrique (PELU).
- L'unité linéaire exponentielle (ELU).
- L'unité linéaire d'erreur gaussienne (GELU).
- L'unité linéaire exponentielle mise à l'échelle (SELU).

Ils ne sont pas explorés ici car ils ne sont presque jamais utilisés dans le modèle de réseau neuronal le plus courant et cet article vise à être aussi court que possible.

- **Fonction d'activation sigmoïde :**

$$f(x) = \frac{1}{(1+e^{-x})} \qquad \text{Éq 9}$$

Cette fonction accepte des entrées négatives ou positives et renvoie toujours des sorties dans la plage (0, 1). La plus grande entrée est mappée très près de 1 tandis que la plus petite entrée est mappée très près de 0. Ceci est couramment utilisé pour les applications où la probabilité d'une classe est requise comme sortie. Il s'agit de la fonction d'activation largement utilisée pour la couche de sortie d'un réseau neuronal de classification binaire.

- **Fonction d'activation Tanh :**

$$f(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \qquad \text{Éq 10}$$

La fonction d'activation tanh suit la même courbe de gradient que la fonction sigmoïde mais ici, la fonction produit des résultats dans la plage (-1, 1). En raison de cette plage, puisque la fonction est centrée sur zéro, elle est principalement utilisée dans les couches cachées d'un réseau de neurones. La sortie de cette fonction peut également être négative. Par conséquent, les résultats peuvent être fortement mappés aux prochains neurones d'entrée.

- **Fonction d'activation Softmax :**

La fonction d'activation soft max est basée sur la fonction sigmoïde qui donne les sorties sous forme de probabilité dans la plage (0, 1). La fonction softmax peut être définie comme une simple combinaison de plusieurs fonctions sigmoïdes.

$$f(x) = \frac{exp(x)}{\sum k\ exp(y)} \quad \text{Éq 11}$$

$k$ : le nombre de classes.

$y$ : le vecteur de sortie.

Alors que la fonction sigmoïde est utilisée pour la classification binaire et donne une probabilité de sortie, la fonction soft max est toujours utilisée pour les applications de classification multi-classes car elle se traduit par les probabilités de sortie de chacune des 'k' classes dans l'application qui s'additionnent toutes jusqu'à 1.

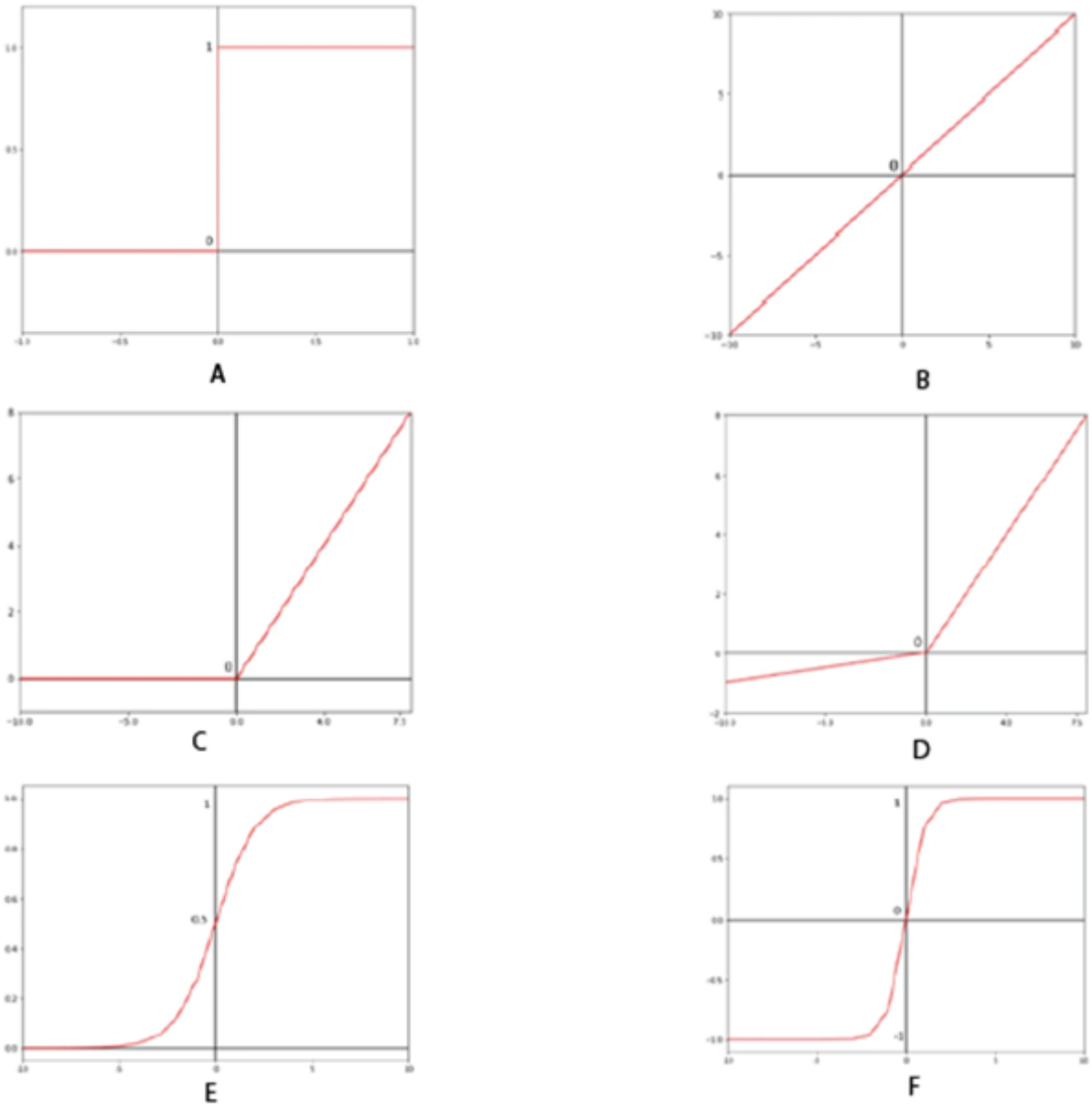



**Figure 2. 6 : Les fonctions d'activation [31]**

A: fonction d'étape binaire, B: Fonction d'activation linéaire,
C: Fonction d'activation Relu (U L R), D: Fonction d'activation Relu qui fuit,
E: Fonction d'activation sigmoïde, F: Fonction d'activation Tanh

## II.7 Algorithmes Deep Learning

Il existe plusieurs algorithmes de Deep Learning populaires et largement utilisés dans divers domaines tels que la vision par ordinateur et le traitement du langage naturel :

### II.7.1 Multi layer Perceptrons

Un perceptron multicouche (MLP) est un type de réseau de neurones artificiels entièrement connecté et orienté vers l'avant. Il est composé d'au moins trois couches de nœuds : une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. À l'exception des nœuds d'entrée, chaque nœud est un neurone qui utilise une fonction d'activation non linéaire.

Le MLP est capable de classer des ensembles de données qui ne sont pas linéairement séparables en utilisant une architecture plus robuste et complexe pour apprendre des modèles de régression et de classification pour des ensembles de données difficiles. La technique d'apprentissage supervisée utilisée pour entraîner un MLP est appelée rétro propagation, qui est basée sur la règle de la chaîne.

Les MLP sont à la base de tous les réseaux de neurones et ont considérablement amélioré la puissance des ordinateurs lorsqu'ils sont appliqués à des problèmes de classification et de régression. [32]

#### II.7.1.1 Fonctionne l'algorithme d'apprentissage en profondeur MLP

- Le fonctionnement de MLP est le même que celui dont nous avons discuté ci-dessus dans notre exemple de données MNIST. La première couche, à savoir la couche d'entrée, prend les entrées et la dernière couche produit la sortie basée sur les couches cachées. Chaque nœud est connecté à chaque nœud de la couche suivante, de sorte que les informations sont constamment transmises entre les multiples couches, c'est pourquoi on parle de réseau à anticipation. [33]
- MLP utilise une technique d'apprentissage supervisé répandue appelée rétro propagation pour la formation **Figure 2. 7**.

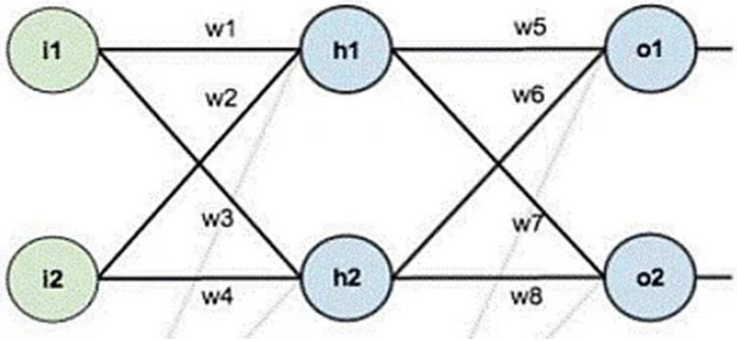



**Figure 2. 7 : Multi-layer Perceptrons [33]**

- Chaque couche cachée est alimentée avec des poids (valeurs attribuées au hasard). La combinaison des poids et de l'entrée est fournie à une fonction d'activation qui est ensuite transmise à la couche suivante pour déterminer la sortie. Si nous n'arrivons pas à la sortie attendue, nous calculons la perte (erreur) et revenons en arrière pour mettre à jour les pondérations. Il s'agit d'un processus itératif jusqu'à ce que la sortie prédite soit obtenue (essai et erreur). Il est essentiel de former le modèle d'apprentissage en profondeur, car les pondérations correctes détermineront votre résultat final.
- Les MLP utilisent couramment les fonctions sigmoïdes, l'unité linéaire rectifiée (Relu) et tanh comme fonctions d'activation. [33]

#### II.7.1.2 Avantages du MLP

- Ils ne font aucune hypothèse concernant les fonctions de densité de probabilité, contrairement aux autres modèles basés sur la probabilité.
- Capacité à fournir la fonction de décision directement en entraînant le perceptron.

#### II.7.1.3 Inconvénients du MLP

- En raison de la fonction de transfert à limite stricte, les perceptrons ne peuvent donner des sorties que sous la forme de 0 et 1.
- Lors de la mise à jour des pondérations dans les couches, le réseau MLP peut être bloqué dans un minimum local, ce qui peut nuire à la précision. [33]

### II.7.2 Récurrent Neural Networks

Les réseaux de neurones récurrents (RNN) sont une classe de réseaux de neurones artificiels dans lesquels les connexions entre les nœuds peuvent créer un cycle, permettant à la sortie de certains nœuds d'affecter l'entrée ultérieure des mêmes nœuds. Contrairement aux réseaux de neurones feedforward, les RNN peuvent utiliser leur état interne (mémoire) pour traiter des séquences d'entrées de longueur variable. Cela les rend applicables à des tâches telles que la reconnaissance de l'écriture manuscrite non segmentée et connectée ou la reconnaissance de la parole.

Les RNN sont couramment utilisés dans diverses applications, telles que la modélisation du langage, la génération de texte, la reconnaissance de la parole, la génération de descriptions d'images et l'étiquetage vidéo. [27]

### II.7.2.1 Fonctionnent les algorithmes d'apprentissage en profondeur RNN

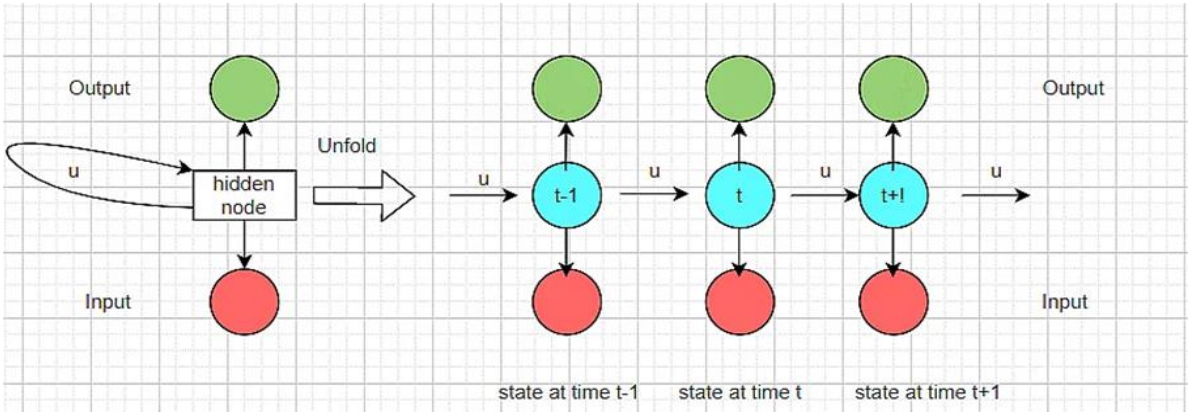



**Figure 2. 8 : fonctionnent les algorithmes d'apprentissage en profondeur RNN [33]**

### II.7.2.2 La structure de base de RNN

La figure ci-dessus montre les différentes étapes pour chaque état temporel d'un réseau de neurones récurrent. Les RNN ne tiennent pas seulement compte des poids des unités cachées pour déterminer la sortie, mais utilisent également les informations tirées de leurs entrées précédentes. Les RNN sont un type spécial de réseau neuronal profond qui peut se souvenir de ces caractères grâce à leur mémoire interne. Une sortie est produite, qui est copiée et renvoyée au réseau neuronal profond comme une boucle. C'est pourquoi l'entrée peut produire une sortie différente en fonction des entrées précédentes dans les couches connectées. [33]

### II.7.2.3 Avantages de RNN

La capacité des modèles RNN à se souvenir des informations tout au long de la période de formation joue un rôle central dans la prédiction des séries chronologiques. [33]

### II.7.2.4 Inconvénients de RNN

- Le calcul est chronophage du fait de son caractère récurrent.
- Difficile de traiter de longues séquences dans l'ensemble de données d'apprentissage, principalement lorsque nous utilisons Relu ou tanh comme fonctions d'activation. [33]

### II.7.3 Long Short-Term Memory Networks (LSTM):

Il s'agit d'un type spécial de RNN qui est capable de capturer des dépendances à long terme dans les séquences de données grâce à des mécanismes de portes spécifiques [33] :

### II.7.3.1 Fonctionnent les algorithmes d'apprentissage en profondeur LSTM

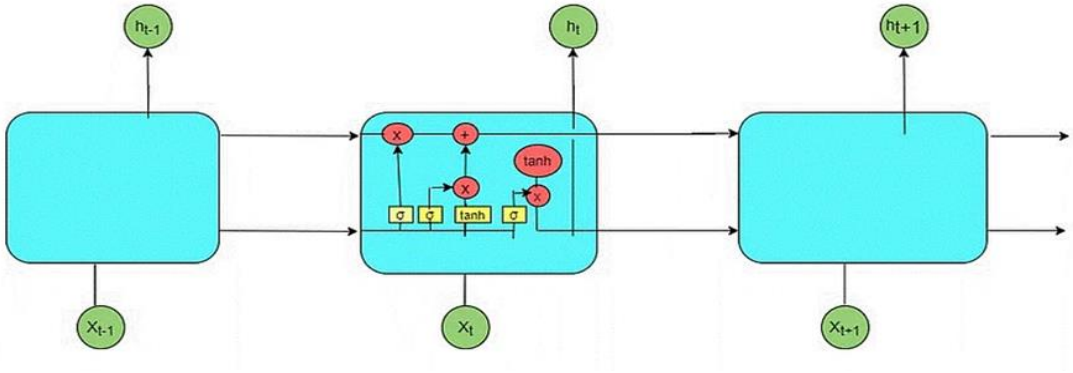



**Figure 2. 9 : fonctionnent les algorithmes d'apprentissage en profondeur LSTM [33]**

Le réseau LSTM se compose de différents blocs de mémoire appelés cellules (les blocs rectangulaires ci-dessus).

L'état de la cellule et l'état caché sont transférés à la cellule suivante. Comme son nom l'indique, les blocs de mémoire se souviennent des choses et les modifications apportées à ces blocs de mémoire sont effectuées via des mécanismes appelés portes.

La clé des LSTM est l'état de la cellule (la ligne horizontale en haut qui traverse le diagramme).

- Étape 1 : - LSTM décide quelles informations doivent être conservées intactes et quelles informations doivent être supprimées de l'état de la cellule. La couche sigmoïde est chargée de prendre cette décision.
- Étape 2 : - LSTM décide des nouvelles informations à conserver et remplace celles non pertinentes identifiées à l'étape 1 - le tanh et la fonction sigmoïde jouent un rôle important dans l'identification des informations pertinentes.
- Étape 3 : - La sortie est déterminée à l'aide de l'état de la cellule qui sera désormais une version filtrée en raison des fonctions sigmoïde et tanh appliquées. [33]

### II.7.3.2 Avantages de LSTM

Comparés aux RNN conventionnels, les LSTM sont très pratiques pour modéliser les séquences chronologiques et les dépendances à longue portée. [33]

### II.7.3.3 Inconvénients de LSTM

Des calculs et des ressources élevées sont nécessaires pour former le modèle LSTM, et c'est aussi un processus qui prend beaucoup de temps. Ils sont sujets au sur ajustement. [33]

### II.7.4 Convolution Neural Networks

Les réseaux de neurones convolutés (CNN) sont un type d'architecture de réseau de neurones d'apprentissage en profondeur couramment utilisé dans la vision par ordinateur. Les CNN sont conçus pour apprendre automatiquement et de manière adaptative les hiérarchies spatiales des caractéristiques grâce à un algorithme de rétro propagation.

Le bloc de construction central d'un CNN est la couche convolutive, où se produit la majorité des calculs. La couche convolutive nécessite quelques composants, notamment des données d'entrée, un ensemble de filtres apprenables, une valeur de foulée et une valeur de remplissage. Les filtres sont utilisés pour balayer les données d'entrée et effectuer des opérations de convolution pour détecter des caractéristiques dans l'image.

Les CNN sont composés de plusieurs blocs de construction, tels que des couches de convolution, des couches de regroupement et des couches entièrement connectées. Le prétraitement requis dans un CNN est beaucoup plus faible par rapport aux autres algorithmes de classification. Les CNN sont largement utilisés dans diverses tâches de vision par ordinateur, telles que la reconnaissance d'objets, la détection de visage et la classification d'images. [34]

#### II.7.4.1 Opération de convolution

L'opération de convolution est une opération mathématique utilisée dans les réseaux de neurones convolutifs (CNN) pour filtrer les données d'entrée et produire une carte de caractéristiques. L'opération de convolution est appliquée aux données d'entrée à l'aide d'un ensemble de filtres apprenables, également appelés noyaux ou noyaux. Comportent des détecteurs de faible largeur, hauteur et profondeur.

Le but de l'opération de convolution est d'extraire et de préserver les caractéristiques importantes des données d'entrée, telles que les bords, les coins et les textures. L'opération de convolution est la première et l'une des étapes les plus importantes du fonctionnement d'un CNN.

L'opération de convolution est souvent suivie d'autres opérations, telles que la mise en commun, l'aplatissement, le remplissage, les couches entièrement connectées et les fonctions d'activation. La combinaison de ces opérations forme les blocs de construction d'un CNN. [34,35]

### II.7.4.2 Les couches de CNN

Convolutional Neural Networks (CNNs) se compose de plusieurs couches, chacune avec sa propre fonction, La combinaison de ces couches forme les éléments constitutifs d'un CNN. [36]

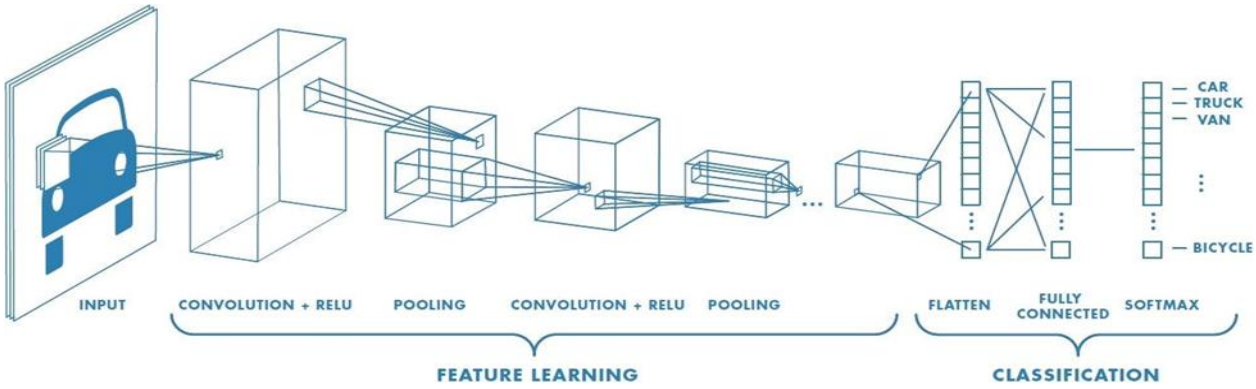



**Figure 2. 10 : A simple CNN architecture [36]**

#### II.7.4.2.1 La couche d'entrée

La couche d'entrée dans un réseau de neurones convolutif (CNN) est la première couche qui prend en entrée les données, qui sont généralement la matrice de pixels de l'image dans le cas du traitement d'image. [36]

#### II.7.4.2.2 La couche de convolution

La couche de convolution est l'une des principales couches utilisées dans les réseaux de neurones convolutifs (CNN). Les couches de convolution sont formées de ce qu'on appelle des filtres. Les filtres sont des tableaux de valeurs appelées feature maps. Chaque couche de convolution prend en entrée une image et produit une feature map. Chaque feature map est obtenue en appliquant le filtre à l'image. Par exemple, si l'image est de taille 5x5 et que le filtre est de taille 3x3, la feature map sera de taille 3x3. La couche de convolution reçoit donc en entrée plusieurs images et calcule la convolution de chacune d'entre elles avec chaque filtre. En résumé, la couche convolutive dans un CNN est responsable de la détection des caractéristiques de l'image en appliquant un filtre de convolution à l'image [38,39]

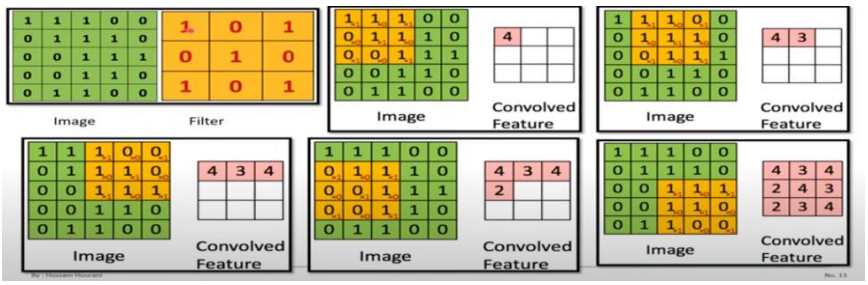



**Figure 2. 11 : les étapes de couche convolution**

#### II.7.4.2.2.1 Les avantages de la couche de convolution sont les suivants :

- Extraction des caractéristiques locales : La convolution permet de capturer les caractéristiques locales de l'image, telles que les bords, les coins et les textures. Ces caractéristiques sont essentielles pour la reconnaissance d'objets et d'autres tâches de vision par ordinateur.
- Partage des poids : Les filtres sont partagés sur l'ensemble de l'image, ce qui permet de réduire le nombre de paramètres à apprendre et rend le modèle plus efficace en termes de mémoire et de calcul.
- Invariance aux translations : La couche de convolution est capable de reconnaître les motifs indépendamment de leur position dans l'image. Cela signifie que même si un objet se déplace dans l'image, le réseau neuronal peut toujours le reconnaître.
- Réduction de la dimensionnalité : La convolution réduit la dimension de l'image en utilisant des filtres de petite taille. Cela permet de réduire la complexité du modèle et d'accélérer les calculs ultérieurs.

#### II.7.4.2.2.2 Couche d'activation dans cnn

Dans un réseau neuronal convolutif (CNN), la couche d'activation est le dernier composant de la couche convolutive qui augmente la non-linéarité de la sortie. Il est utilisé pour introduire une non-linéarité dans la sortie d'un neurone. La fonction d'activation est généralement un composant distinct de la couche convolutive. La fonction d'activation utilisée dans les couches cachées est généralement choisie en fonction du type d'architecture de réseau neuronal, et la fonction d'activation par défaut moderne pour les couches cachées est la fonction ReLU. [40]

#### II.7.4.2.3 Mise en commun maximale (Max pooling)

La mise en commun maximale est un type d'opération de mise en commun couramment utilisée dans les réseaux de neurones convolutifs (CNN) pour réduire la dimensionnalité des images en réduisant le nombre de pixels dans la sortie de la couche convolutive précédente. Cela fonctionne en sélectionnant la valeur maximale de chaque pool. La mise en commun maximale conserve les fonctionnalités les plus importantes de la carte des fonctionnalités. La mise en commun maximale est effectuée pour éviter le sur ajustement en fournissant une forme abstraite de la représentation et pour réduire la quantité de calculs dans le réseau. [41]

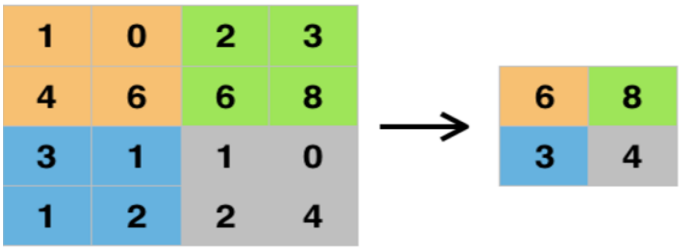



**Figure 2. 12 : Max pooling**

### II.7.4.2.4 Couche d'aplatissement (Flattenning layer)

La couche de flattening (aplanissement) est une couche couramment utilisée dans les réseaux de neurones, en particulier dans les réseaux de neurones convolutifs (CNN). Son rôle principal est de convertir les données multidimensionnelles en un format unidimensionnel, ce qui permet de les utiliser comme entrée pour les couches de neurones entièrement connectées qui suivent. L'aplatissement est utilisé pour convertir tous les tableaux bidimensionnels résultants des cartes d'entités regroupées en un seul long vecteur linéaire continu. La matrice aplatie est alimentée en entrée de la couche entièrement connectée pour classer l'image. L'étape d'aplatissement est une étape simple et rafraîchissante impliquée dans la construction d'un CNN. Après l'étape

D'aplatissement, vous retrouvez avec un long vecteur de données d'entrée que vous passez ensuite à travers le réseau de neurones artificiels. [42]

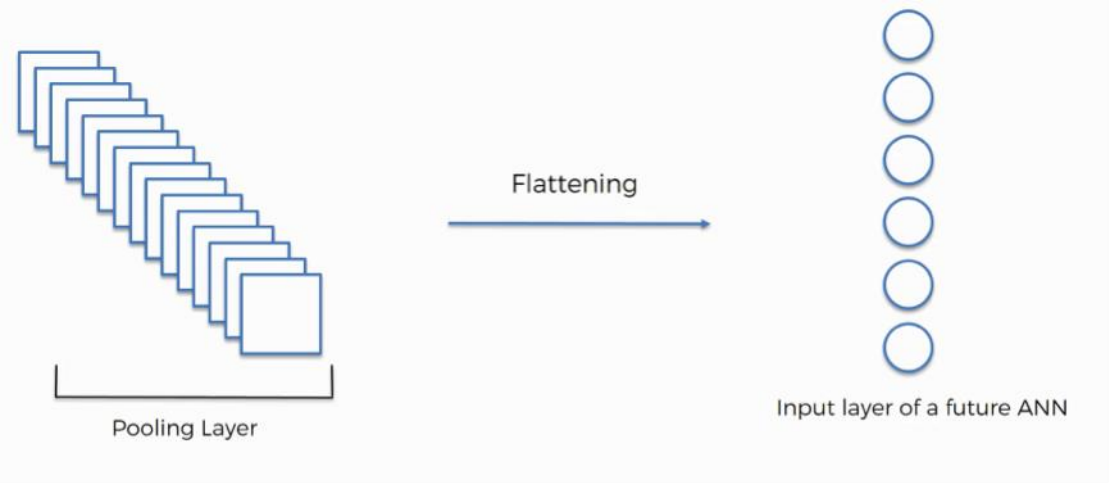



**Figure 2. 13 : couche d'aplatissement (flattening layer)**

#### II.7.4.2.5 Entièrement connecté (Fully connected)

Dans un réseau neuronal convolutif (CNN), la couche entièrement connectée (FC) est la dernière couche du réseau. L'entrée de la couche entièrement connectée est la sortie de la couche de regroupement ou de convolution finale, qui est aplatie puis introduite dans la couche entièrement connectée. Dans les couches entièrement connectées, chaque neurone applique une transformation linéaire au vecteur d'entrée via une matrice de pondérations, et une transformation non linéaire est ensuite appliquée au produit. La couche entièrement connectée est utilisée pour apprendre des combinaisons non linéaires des fonctionnalités de haut niveau dans les données. [43]

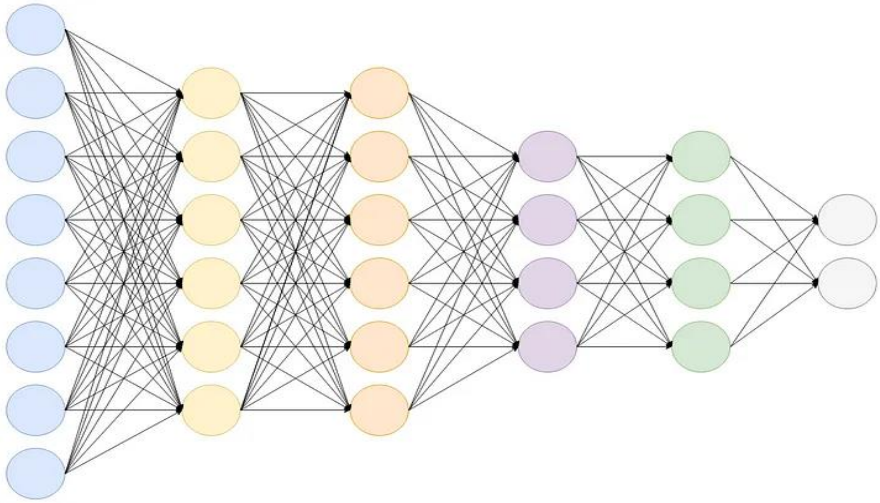


**Figure 2. 14 : Réseau entièrement connecté**

### II.7.4.3 La normalisation par lots (BN)

La normalisation par lots (BN) est une technique utilisée dans les réseaux de neurones convolutifs (CNN) pour normaliser les entrées d'une couche pour chaque mini-lot. BN définit essentiellement les pixels de toutes les cartes d'entités dans une couche de convolution sur une nouvelle moyenne et une nouvelle écart type. Cela se fait entre les couches d'un réseau de neurones plutôt que dans les données brutes, et cela se fait le long de mini-lots au lieu de l'ensemble de données complet. BN rend la formation des réseaux de neurones profonds (DNN) plus rapide et plus stable, et il assure la régularisation, évitant le sur ajustement. [44]

### II.7.4.4 Abandon (Dropout)

La couche d'abandon est une technique de régularisation utilisée dans les réseaux de neurones convolutifs (CNN) pour empêcher le sur ajustement. Il définit de manière aléatoire les unités d'entrée sur 0 avec une fréquence de taux à chaque étape pendant le temps d'entraînement. La couche Dropout agit comme un masque, éliminant les contributions de certains neurones à la couche suivante lors de la formation. Dropout est implémenté par couche dans un réseau de neurones et peut être utilisé avec la plupart des types de couches, telles que

les couches denses entièrement connectées, les couches convolutives, et des couches récurrentes telles que la couche réseau de mémoire longue à court terme. [45]

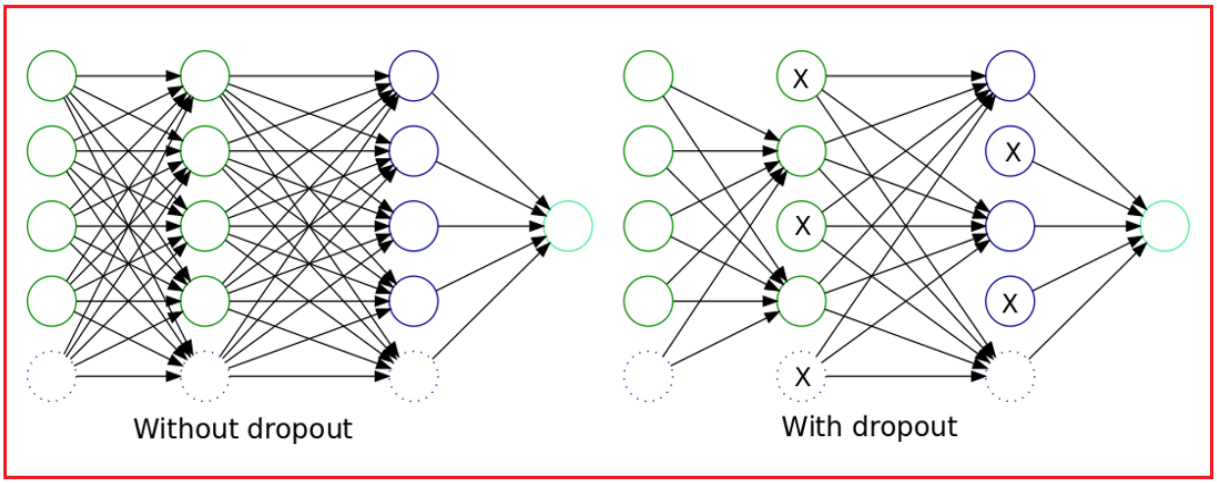



**Figure 2. 15 : La technique d'abandon**

### II.7.4.5L'optimisation des performances des réseaux de neurones convolutifs (CNN)

L'optimisation des performances des réseaux de neurones convolutifs (CNN) peut être obtenue grâce à diverses méthodes. Voici quelques façons d'optimiser les performances des CNN [46] :

- Optimisation des performances de calcul : cette méthode réduit les noyaux de convolution redondants d'un CNN avec des contraintes de performances et d'architecture. Il élague les noyaux et réduit le temps de calcul.
- Réglage fin des paramètres : cette technique d'apprentissage par transfert peut améliorer les performances d'un CNN par rapport à l'entraînement à partir de zéro.
- Optimisation de l'architecture : Cette méthode consiste à optimiser l'architecture du CNN, comme l'ajustement du nombre de couches, de filtres et de neurones.
- Optimisation de l'inférence : Cette méthode optimise le processus d'inférence du CNN sur les processeurs, ce qui peut améliorer ses performances.
- Optimisation de la formation : cette méthode consiste à optimiser le processus de formation du CNN, par exemple en utilisant l'augmentation des données, la régularisation et l'arrêt précoce.
- Comprendre les performances de généralisation et d'optimisation : cette méthode établit des garanties d'optimisation pour les algorithmes d'entraînement basés sur la descente de gradient et aide à comprendre les performances de généralisation d'un CNN.

### II.7.4.6 Facteurs peuvent affecter les performances des réseaux de neurones convolutifs (CNN)

Plusieurs facteurs peuvent affecter les performances des réseaux de neurones convolutifs (CNN), notamment [47] :

- Architecture CNN : la profondeur, la largeur et la cardinalité du réseau peuvent toutes avoir un impact sur ses performances.
- Taux d'apprentissage : le taux d'apprentissage détermine la rapidité avec laquelle le réseau ajuste ses pondérations pendant l'entraînement. Un taux d'apprentissage élevé peut amener le réseau à dépasser les poids optimaux, tandis qu'un faible taux d'apprentissage peut entraîner une convergence lente.
- Couche de regroupement : La couche de regroupement peut exercer une influence significative sur la capacité prédictive du modèle CNN. L'ajustement de la couche convolutive peut également avoir un impact sur les performances.
- Méthode d'optimisation : la méthode d'optimisation utilisée lors de l'entraînement, comme AdaGrad ou la descente de gradient stochastique, peut affecter la vitesse de convergence et les performances du CNN.
- Apprentissage par transfert : le réglage fin des modèles pré-entraînés peut améliorer les performances d'un CNN par rapport à un apprentissage à partir de zéro.
- Résolution d'image : La résolution des images d'entrée peut affecter les performances d'un CNN, les résolutions plus élevées nécessitant généralement plus de ressources de calcul.
- Apprentissage de l'intelligence artificielle : La capacité du réseau à apprendre et à généraliser à partir des données d'apprentissage est cruciale pour sa performance.
- Extraction de caractéristiques : la capacité du réseau à extraire les caractéristiques pertinentes des données d'entrée est également importante pour ses performances.

### II.7.4.7 Un exemple dans CNN

Pour spécifier un exemple dans CNN, vous devez :

#### II.7.4.7.1 Base de données

Un jeu de données est un ensemble de données organisées de manière spécifique. Il peut être utilisé à diverses fins telles que la modélisation, la compréhension et la participation à un dialogue de recherche d'informations.

#### II.7.4.7.2 La formation

La formation d'un CNN implique plusieurs étapes, notamment la définition de l'architecture du réseau, la définition d'hyper paramètres, la préparation de l'ensemble de données et l'optimisation du processus de formation.

##### II.7.4.7.2.1 L'entropie croisée catégorielle

L'entropie croisée catégorielle est une fonction de perte utilisée dans les problèmes de classification multi-classes où il existe deux étiquettes de sortie ou plus. Il est utilisé pour entraîner un CNN à générer une probabilité sur les classes C pour chaque image.

##### II.7.4.7.2.2 Adam

Adam est un algorithme d'optimisation utilisé dans la formation de modèles d'apprentissage en profondeur, qui est une extension de la descente de gradient stochastique. Il est efficace lorsque vous travaillez avec de gros problèmes impliquant beaucoup de données ou de paramètres et nécessite moins de mémoire.

##### II.7.4.7.2.3 La précision (Accuracy)

La précision est une métrique utilisée pour évaluer les performances d'un modèle CNN dans les problèmes de classification.

##### II.7.4.7.2.4 Epoch

Epoch est un terme utilisé dans la formation des réseaux neuronaux convolutionnels (CNN) et se réfère à un cycle à travers l'ensemble de données de formation complète. Dans une époque, toutes les données sont traitées une fois individuellement de l'avant et de l'arrière vers le réseau.

##### II.7.4.7.2.5 La taille du lot (Batch size)

La taille du lot est un hyperparamètre qui définit le nombre d'échantillons qui sont transmis simultanément au réseau de neurones pendant la formation.

##### II.7.4.7.2.6 Le nombre d'itérations

Le nombre d'itérations dans la formation CNN équivaut au nombre de lots nécessaires pour terminer une époque.

##### II.7.4.7.2.7 La division de validation

La division de validation est une technique de partitionnement des données en ensembles d'apprentissage et de validation. L'ensemble de validation est utilisé pour évaluer les performances du modèle pendant la formation et éviter le surajustement.

#### II.7.4.7.2.8 L'initialisation du poids

L'initialisation du poids est un aspect important de la conception d'un modèle de réseau neuronal d'apprentissage en profondeur. Les valeurs initiales des pondérations peuvent affecter la précision du modèle et éviter des problèmes tels que le problème de gradient de fuite ou le problème de gradient d'explosion.

#### II.7.4.7.2.9 La normalisation par lots

La normalisation par lots est une technique utilisée dans les réseaux de neurones d'apprentissage en profondeur pour normaliser les sorties inter couches d'un réseau de neurones. Cela se fait en mini-lots plutôt que dans les données brutes.

### II.7.5 CIFAR CNN Exemple

Pour la classification, nous avons utilisé le jeu de données Fruit-360, qui contient 82 213 images (100 × 100 pixels) de fruits et légumes de 120 classes, déjà subdivisées en ensembles d'apprentissage et de test. Nous avons sélectionné six catégories (trois types de pommes et de poires) dans l'ensemble de données : Apple Golden 1, Apple Pink Lady, Apple Red 1, Pear Red, Pear Williams et Pear Monster. Une fois les catégories sélectionnées pour réaliser l'étude de classification, l'étape suivante consiste à concevoir l'architecture CNN. La figure 38 illustre l'architecture CNN adoptée dans ce cas. Nous avons conçu un CNN avec une architecture assez simple composée de la couche d'entrée, de trois couches convolutionnelles, d'une couche d'aplatissement, d'une couche entièrement connectée et de la couche de sortie.

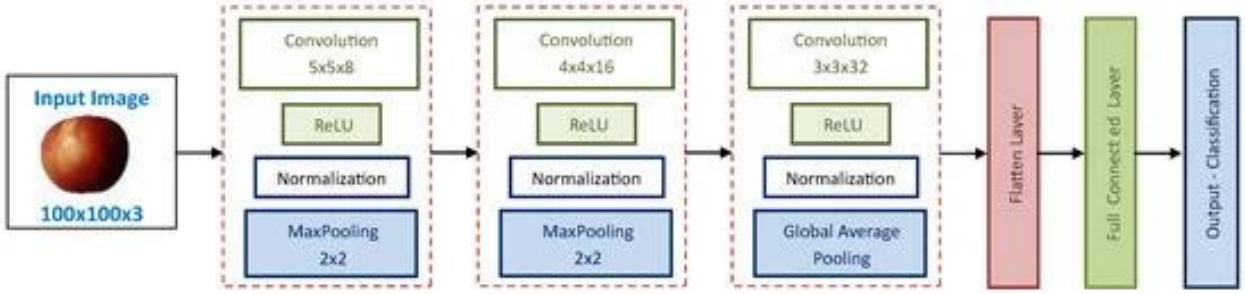



**Figure 2. 16 : Architecture CNN conçue pour la classification des fruits**

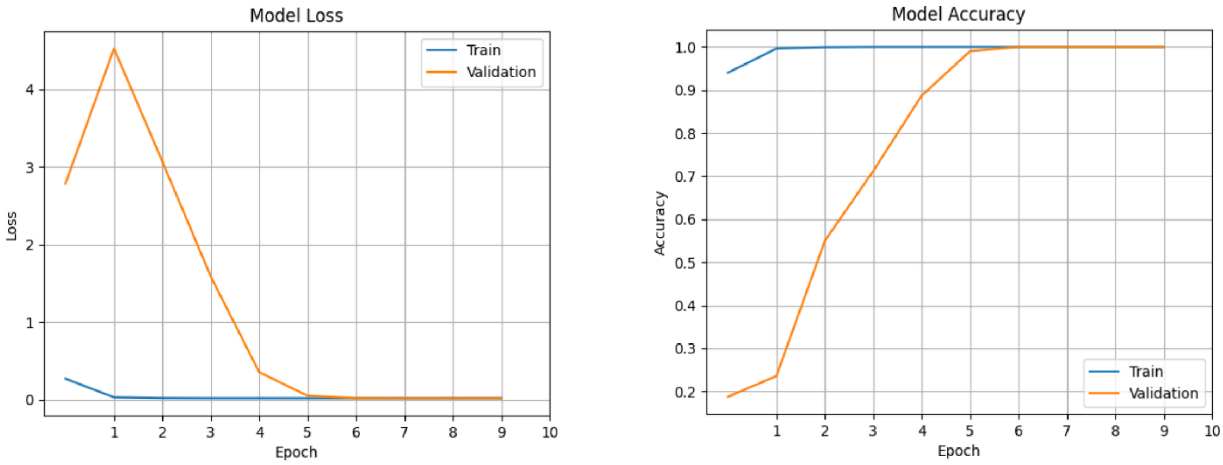
Nous formons notre modèle par 10 époques et en utilisant l'optimiseur Stochastic Gradient Descent (SGD) avec momentum. Tout d'abord, tous les filtres sont initialisés aléatoirement à partir d'une distribution normale et tous les biais sont égaux à zéro. Certains des paramètres d'ajustement du réseau sont indiqués dans le tableau 4, qui ont été fixés en appliquant une stratégie de grille grossière à fine sur une petite partie de l'ensemble d'apprentissage. Tout d'abord, un ensemble de paramètres a été défini, qui a varié en combinaison au fur et à mesure qu'un petit nombre d'époques d'entraînement (3 à 5) ont été exécutées. Selon les résultats préliminaires obtenus, les paramètres ont été affinés pour ensuite exécuter un plus grand nombre

d'époques d'entraînement et définir celles pour lesquelles les meilleures performances d'entraînement ont été obtenues.

**Tableau 2. 1 : paramétres du modèle CNN pour la classification des fruits**

| Weight Decay | DropOut | Learning Rate | Mmentum | Batch Size |
|---|---|---|---|---|
| 5*10^-4 | 0.5 | 0.001 | 0.9 | 32 |

Nous effectuons une validation croisée de 10 fois sur la base de données visant à obtenir des résultats impartiaux dans nos expériences. En outre, nous séparons 10 % des données d'entraînement à des fins de validation. Ainsi, notre ensemble de formation était composé de 2779 images et 307 images dans l'ensemble de validation. La figure 39 montre les courbes de perte moyenne et de précision pendant le processus de formation du modèle. L'axe des x représente le nombre d'époques d'entraînement. Il convient de noter qu'à partir de la 5ème époque, la valeur de perte est proche de 0 et le résultat de précision est proche de 1 pour l'entraînement et la validation. Ces résultats d'entraînement et de validation semblent parfaits en raison de la faible variabilité du jeu de données utilisé.




(a) (b)

**Figure 2. 17 : Courbes de (a) perte moyenne et (b) précision moyenne lors de la formation du modèle pour la classification des fruites**

Après le processus de formation, nous évaluons notre modèle sur 1034 images de test. Sur la figure 40, nous montrons les cartes d'activation obtenues par la 3ème couche de convolution pour une Apple Pink Lady et une Pear Williams. On remarque les différences entre les deux fruits en fonction de leurs cartes d'activation, qui sont ensuite utilisées par les couches entièrement connectées pour faire les prédictions de classe. La figure 41 illustre la matrice de confusion pour les résultats de classification sur l'ensemble d'images de test, où l'on peut noter les erreurs de classification entre les classes Pear Red, Apple Red 1 et Apple Pink Lady. La précision moyenne finale était de 95,45 % avec un score F1 moyen de 0,96.

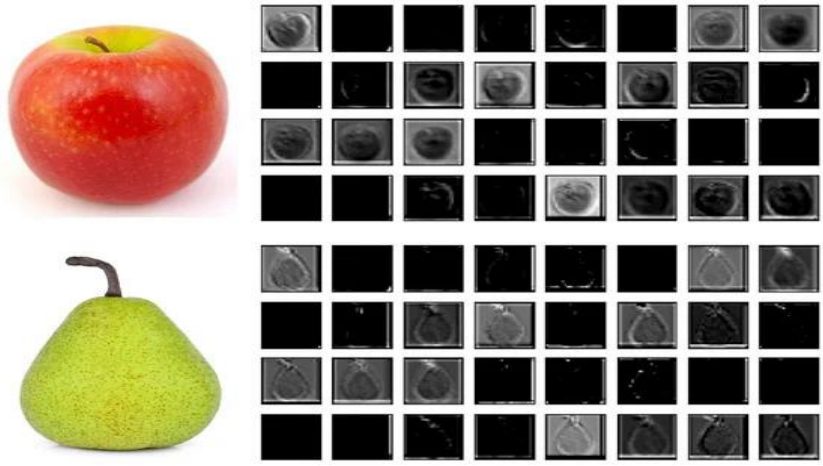


**Figure 2. 18 : Exemple de cartes d'activation de la 3e couche de convolution pour deux images du jeu de données Fruit-360**

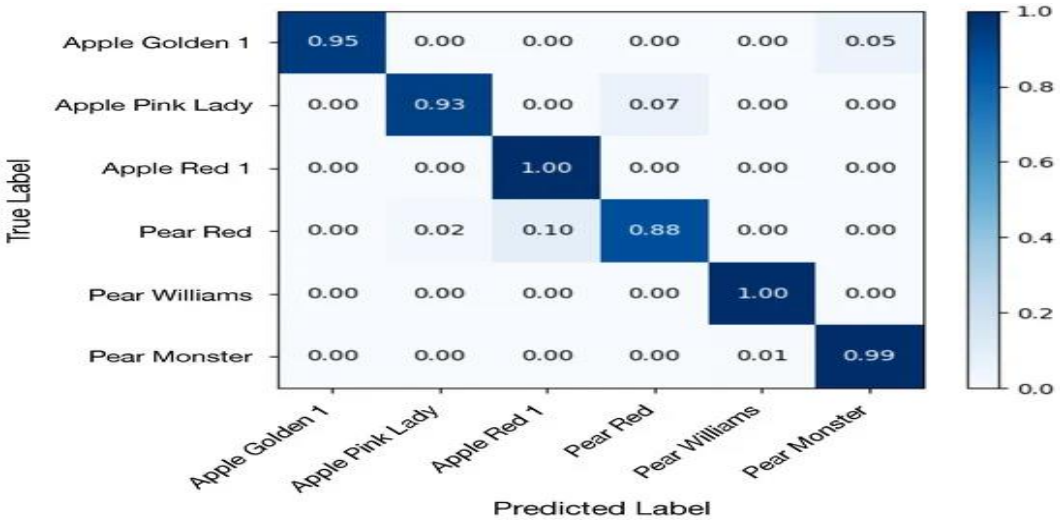



**Figure 2. 19 : Matrice de confusion pour les résultats de la classification des fruits sur l'ensemble de données Fruit 360**

En plus de l'exemple précédent et dans le but de comparer les performances, nous avons implémenté des solutions CNN plus approfondies basées sur des modèles pré-formés bien connus avec l'ensemble de données Imagenet pour comparer leurs performances. Pour cela, nous avons utilisé des architectures de différentes profondeurs, dont les modèles les plus utilisés dans la littérature pour la classification des fruits. Bien que le jeu de données utilisé soit différent d'Imagenet, afin d'apporter un exemple aussi simple que possible, nous avons évité d'optimiser le modèle pré-entraîné pour le jeu de données utilisé. Par conséquent, nous avons chargé chaque modèle préformé avec les poids appris sur Imagenet et gelé leur base convolutive. Ensuite, nous avons changé leurs couches FC supérieures en ajoutant une couche FC et en modifiant le nombre de sorties à six classes.

Tableau suivante résume les résultats pour chaque modèle. Nous effectuons le processus de formation en utilisant les mêmes paramètres que dans l'exemple initial pour obtenir des résultats comparables, ce qui signifie la même forme d'entrée et la même taille de lot, une validation croisée de 10 fois, la même répartition formation-validation-test, 10 époques de formation et l'optimiseur SGD avec le même taux d'apprentissage et la même dynamique. Il convient de noter qu'à mesure que la complexité (c'est-à-dire la profondeur) du modèle augmente, le surajustement augmente également, même en appliquant l'augmentation des données et en ajoutant l'abandon. Ce comportement est dû au fait que l'ensemble de données utilisé est petit sans variabilité suffisante. Par conséquent, les résultats montrent que la complexité d'un modèle CNN doit correspondre à la complexité de la tâche de classification et à la quantité de données disponibles. Pour cette raison, des modèles plus simples comme l'exemple proposé et AlexNet obtiennent d'aussi bons résultats. Illustration 8a. [67]

**Tableau 2. 2 : Comparaison entre l'exemple proposé et les modèles pré-entrainés pour la classification des fruits sur l'ensemble de données Fruit-360**

| CNN Model | DEpth | Training | | Validation | | Testing | |
|---|---|---|---|---|---|---|---|
| | | Loss | Accuracy | Loss | Accuracy | Accuracy | F1_Score |
| Proposed example | 6 | 0.0167 | 100% | 0.0165 | 100% | 95.45% | 0.96 |
| AlexNet | 8 | 1.2*10^-5 | 100% | 4.9*10^-5 | 100% | 100% | 1 |
| VGG16 | 16 | 0.2067 | 96.32% | 0.2070 | 95.94% | 91.32% | 0.89 |
| Mobile Net | 88 | 0.0799 | 97.45% | 0.7201 | 73.86% | 70.02% | 0.67 |
| Inception V3 | 159 | 0.5592 | 80.49% | 1.1755 | 62.82% | 54.49% | 0.49 |
| ResNet50 | 168 | 0.1436 | 99.17% | 0.9102 | 66.96% | 57.74% | 0.48 |

## II.8 Extraction des caractéristiques des signaux acoustiques

L'extraction de caractéristiques est un processus de réduction de dimension d'un signal qui rend le signal plus adapté aux algorithmes d'apprentissage automatique. Dans le contexte des signaux acoustiques, l'extraction de caractéristiques est utilisée pour mettre en évidence les composantes les plus dominantes et les plus discriminantes du signal. L'extraction de caractéristiques a pour but d'illustrer un signal de parole par un nombre prédéterminé de composantes du signal. Certains algorithmes d'extraction de caractéristiques couramment utilisés pour les signaux vocaux incluent le centroïde spectral.

La réduction du bruit est le fondement du prétraitement du signal acoustique, et l'extraction de caractéristiques pour les signaux de réduction du bruit peut obtenir des informations utiles à partir du signal. En pratique, la détection acoustique traite de grands volumes de données puisqu'elle acquiert des signaux audios sous des fréquences d'échantillonnage relativement élevées. [48,49]

### II.8.1 Kurtogramme

Le kurtogramme est un outil d'analyse spectrale du quatrième ordre utilisé pour détecter et caractériser les non-stationnarités dans un signal. Il est utilisé dans le diagnostic de défaut de roulement pour extraire le signal le plus impulsif du signal. Le kurtogramme montre les résultats de kurtosis pour une plage de longueurs de fenêtre et de fréquences, où un niveau élevé de kurtosis correspond à un niveau élevé de comportement non stationnaire ou non gaussien dans le signal. Kurtogram peut sélectionner de manière adaptative la bande de fréquence de résonance, puis la fréquence de défaut caractéristique peut être obtenue en analysant la bande sélectionnée. [50]

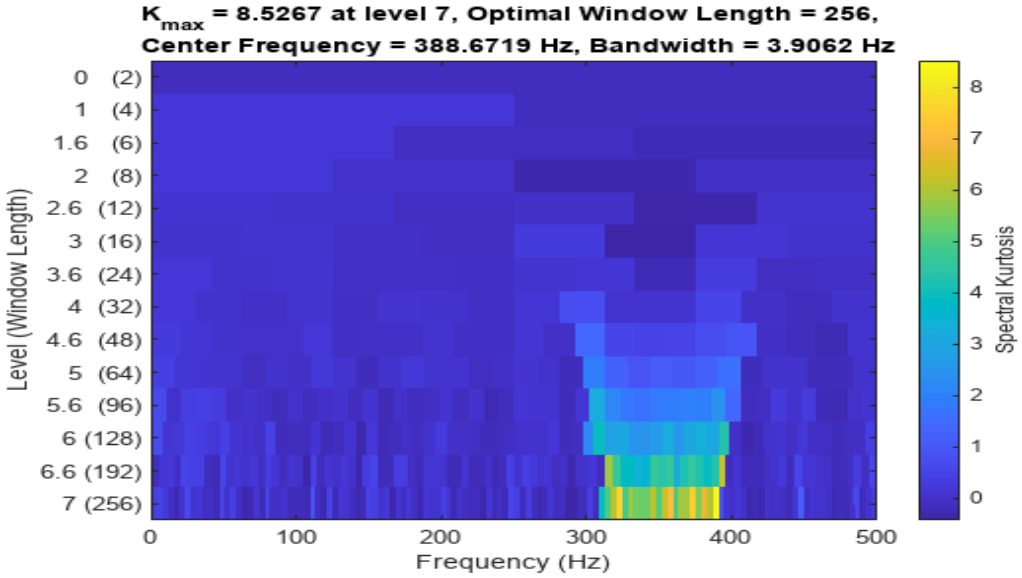



**Figure 2. 20 : kurtogramme**

### II.8.2 Spectrogramme

Un spectrogramme est une représentation visuelle du spectre des fréquences d'un signal tel qu'il varie dans le temps. Il cartographie le son de la même manière qu'une partition musicale, en cartographiant uniquement la fréquence plutôt que les notes de musique. Un spectrogramme est une manière visuelle de représenter la force du signal, ou « intensité », d'un signal dans le temps à diverses fréquences présentes dans une forme d'onde particulière. Les spectrogrammes sont largement utilisés dans les domaines de la musique, de la linguistique, du sonar, du radar, du traitement de la parole, de la sismologie et autres.

Les spectrogrammes audios peuvent être utilisés pour identifier phonétiquement les mots prononcés et pour analyser les différents cris d'animaux. Un spectrogramme peut être généré par un spectromètre optique, un banc de filtres passe-bande, par transformée de Fourier ou par transformée en ondelettes (auquel cas on l'appelle aussi scalogramme où scalogramme). Les spectrogrammes sont généralement représentés sous forme de cartes thermiques, c'est-à-dire sous forme d'images dont l'intensité est indiquée en faisant varier la couleur ou la luminosité. [51]

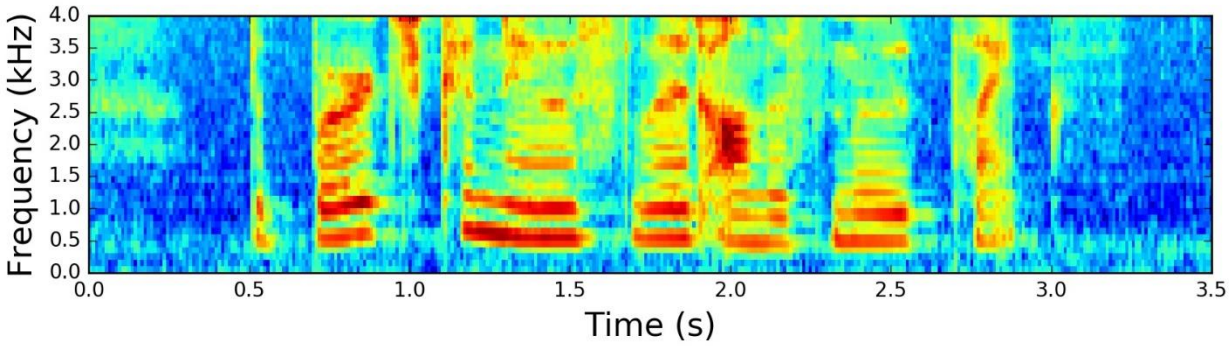



**Figure 2. 21 : spectrogramme**

### II.8.3 MFCC

Les coefficients cepstraux de fréquence Mel (MFCC) est un ensemble de caractéristiques utilisées dans le traitement du son pour représenter le spectre de puissance à court terme d'un son. Ils ont été développés au MIT à la fin des années 1960 pour étudier les modèles de parole. Les MFCC sont des coefficients qui constituent collectivement un cepstre de fréquence Mel (MFC), qui est dérivé d'un type de représentation cepstrale du clip audio. La différence entre le cepstre et le cepstre à fréquence mel est que dans le MFC, les bandes de fréquences sont également espacées sur l'échelle mel, qui est une échelle non linéaire de fréquence. Le calcul des MFCC consiste à produire des sorties de banque de filtres logarithmiques et à les multiplier par 20 pour obtenir des enveloppes spectrales en décibels, puis à prendre la transformée en cosinus discrète (DCT) de l'enveloppe spectrale 1. Les MFCC sont largement utilisés dans divers domaines, notamment le traitement de la parole et de la musique. [52,53]

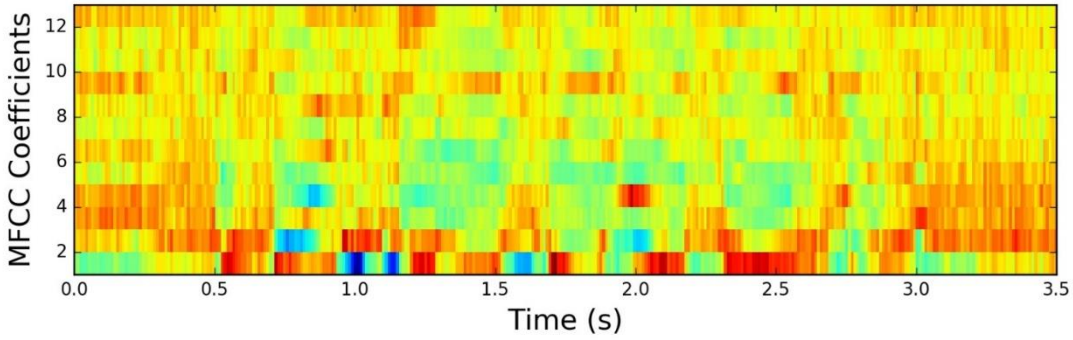



**Figure 2. 22 : MFCC**

### II.8.3.1 Étapes de calcul de la matrice MFCC

Pour calculer la matrice des coefficients cepstraux de fréquence Mel (MFCC), les étapes suivantes sont généralement suivies :

- Cadrage : le signal audio est divisé en trames courtes, généralement d'une durée de 20 à 40 ms, avec un chevauchement entre les trames adjacentes.
- Fenêtrage : une fonction de fenêtrage, telle que la fenêtre de Hamming, est appliquée à chaque image pour réduire les fuites spectrales.
- Transformée de Fourier rapide (FFT) : le spectre de puissance de chaque image est calculé à l'aide de l'algorithme FFT.
- Banc de filtres Mel : Un ensemble de filtres triangulaires espacés sur l'échelle Mel est appliqué au spectre de puissance pour obtenir un ensemble d'énergies de banc de filtres Mel.
- Logarithme : le logarithme de l'énergie de chaque banc de filtres Mel est utilisé pour compresser la plage dynamique du spectre.
- Transformée discrète en cosinus (DCT) : la DCT est appliquée aux énergies du banc de filtres logarithmiques pour obtenir un ensemble de MFCC. [53]

Différentes bibliothèques et outils peuvent avoir des implémentations légèrement différentes de ces étapes, mais ils suivent généralement cette procédure générale.

#### II.8.3.1.1 Cadrage en mfcc (segmentation)

Le cadrage est une technique fondamentale de traitement du signal qui consiste à diviser un long signal en morceaux courts appelés trames. Le nombre d'échantillons dans une trame est appelé longueur de trame et le nombre d'échantillons entre les trames est appelé longueur de saut. Le processus de cadrage est illustré dans Figure II-20. Le but du cadrage est d'analyser chaque pièce séparément car les signaux réels ont tendance à être approximativement stationnaires si nous ne regardons que de courts fragments. Le cadrage est couramment utilisé dans l'analyse par transformée de Fourier à court terme (STFT) et par coefficients cepstraux à fréquence Mel (MFCC) des signaux vocaux.

Les tailles de trame typiques dans le traitement de la parole vont de 20 ms à 40 ms avec un chevauchement de 50 % (+/-10 %) entre les trames consécutives. Les paramètres populaires sont 25 ms pour la taille de trame frame_size = 0.025 et une foulée de 10 ms (chevauchement de 15 ms) frame_stride = 0.01.

Le nombre d'échantillons dans une trame est connu sous le nom de **longueur de trame**, que nous désignerons par$N_F$.

La longueur de trame détermine la durée du fragment analysé $(t = \frac{N_F}{f_8})$

En plus de la longueur de la trame, nous devons également décider du nombre de trames à prendre. Ceci est contrôlé par la **longueur de saut** : le nombre d'échantillons entre les trames, que nous noterons$N_H$.[54]

Le processus de cadrage est illustré par la **Figure 2. 23**

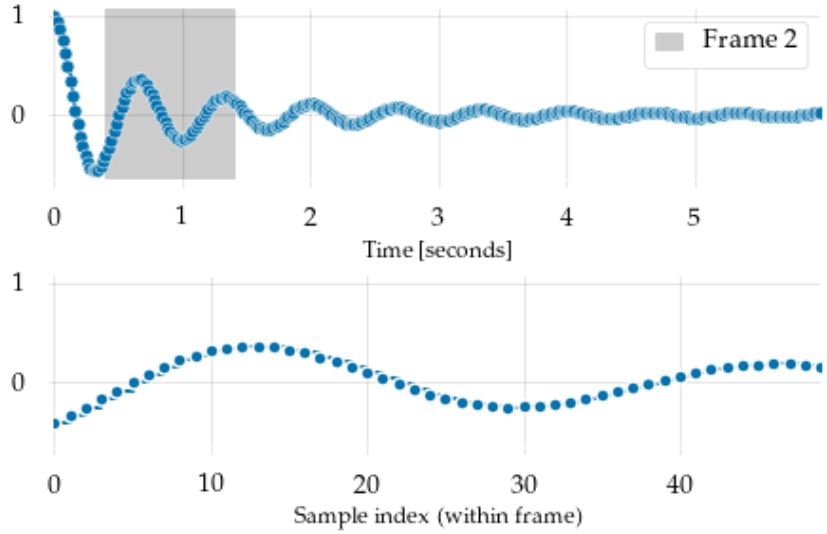



**Figure 2. 23 : Cadrage en MFCC (segmentation)**

**Comptage des cadres**

Un signe de $N$ échantillons, avec longueur de trame $N_F \leq N$ et la longueur du houblon $N_H$ produira $k$ cadres, où

$$\boldsymbol{K = 1 + \left\lfloor \frac{N - N_F}{N_H} \right\rfloor} \qquad \text{Éq 12}$$

**II.8.3.1.2 Fenêtrage**

Le fenêtrage est une technique utilisée dans le calcul du MFCC pour réduire l'effet de fuite spectrale. La fenêtre de Hamming ou de Hanning est appliquée aux trames vocales brutes pour réduire l'effet de fuite spectrale. La nouvelle famille proposée de techniques de fenêtrage peut être utilisée pour calculer le MFCC pour la reconnaissance automatique du locuteur. Le bloc Audio MFCC d'Edge Impulse Studio extrait les coefficients d'un signal audio à l'aide d'une fenêtre glissante pour la normalisation de la moyenne cepstrale locale. La fonction mfcc de MATLAB partitionne la parole en trames et calcule les caractéristiques cepstrales pour chaque trame à l'aide d'une fonction de fenêtre [55]

#### II.8.3.1.2.1 Hamming

La fenêtre de Hamming est un type de fonction de fenêtre utilisée dans le traitement du signal pour réduire les fuites spectrales lors de l'exécution d'une transformée de Fourier sur un signal. Voici quelques points clés sur la fenêtre de Hamming [56] :

$$w(n) = 0.54 - 0.46\,cos\left(\frac{2\pi n}{N-1}\right) \qquad \text{Éq 13}$$

**N** : la longueur de la fenêtre

**n** : l'indice de l'échantillon.

La fenêtre de Hamming est optimisée pour minimiser le lobe latéral le plus proche, ce qui la rend utile pour la mise en forme des impulsions.

La fenêtre de Hamming est souvent utilisée en combinaison avec d'autres fonctions de fenêtre, telles que les fenêtres Blackman-Harris ou Kaiser, pour obtenir une meilleure résolution spectrale.

La fenêtre Hamming peut être générée à l'aide de fonctions intégrées dans MATLAB et Python, telles que hamming () dans MATLAB et scipy. Signal.hamming() dans Python

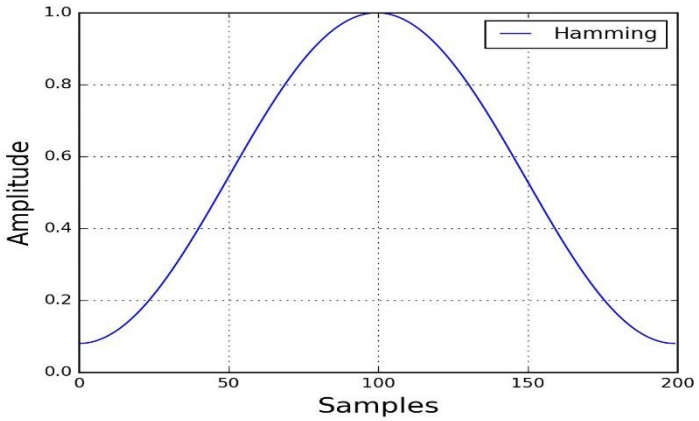



**Figure 2. 24 :fenêtre de hamming**

#### II.8.3.1.3 Transformée de Fourier rapide (FFT)

La transformée de Fourier rapide (FFT) est une technique d'analyse du signal utilisée pour extraire et compresser certaines caractéristiques du signal vocal sans perdre aucune information pertinente afin que le traitement de la parole devienne plus facile. La FFT est utilisée pour décomposer le signal en un signal sinusoïdal. Cependant, le MFCC est une caractéristique motivée par la perception qui correspond à la façon dont les humains perçoivent le son. C'est une technique utilisée dans les systèmes d'analyse de la parole et de l'audio qui utilise la transformée discrète en cosinus (DCT) car elle a l'avantage d'être une transformée réel-réel, et parce que les coefficients résultants sont en pratique décarrelés les uns des autres. [23] Par conséquent, MFCC utilise DCT au lieu de FFT.[57]

L'équation suivante puis calculez le spectre de puissance :

$$\boldsymbol{p} = \frac{|\boldsymbol{FFT}(\boldsymbol{X_{je}})|^2}{\boldsymbol{N}} \quad \text{Éq 14}$$

### II.8.3.1.4 Banc de filtres Mel

Les coefficients cepstraux de fréquence Mel (MFCC) utilisent une échelle non linéaire appelée échelle Mel pour diviser la bande de fréquences en filtres triangulaires. [58]

La dernière étape du calcul des bancs de filtres consiste à appliquer des filtres triangulaires, généralement 40 filtres, n_filt = 40 sur une échelle Mel au spectre de puissance pour extraire les bandes de fréquences. L'échelle Mel vise à imiter la perception non linéaire de l'oreille humaine du son, en étant plus discriminante aux basses fréquences et moins discriminative aux hautes fréquences. Nous pouvons convertir entre Hertz (F) et Mel (m) en utilisant les équations suivantes :

$$m = 2595 log_{10}(1 + \frac{F}{700}) \quad \text{Éq 15}$$

Avec :

$$F = 700(10^{\frac{m}{2595}} - 1) \quad \text{Éq 16}$$

Chaque filtre du banc de filtres est triangulaire avec une réponse de 1 à la fréquence centrale et diminue linéairement vers 0 jusqu'à ce qu'il atteigne les fréquences centrales des deux filtres adjacents où la réponse est de 0, comme le montre cette figure :

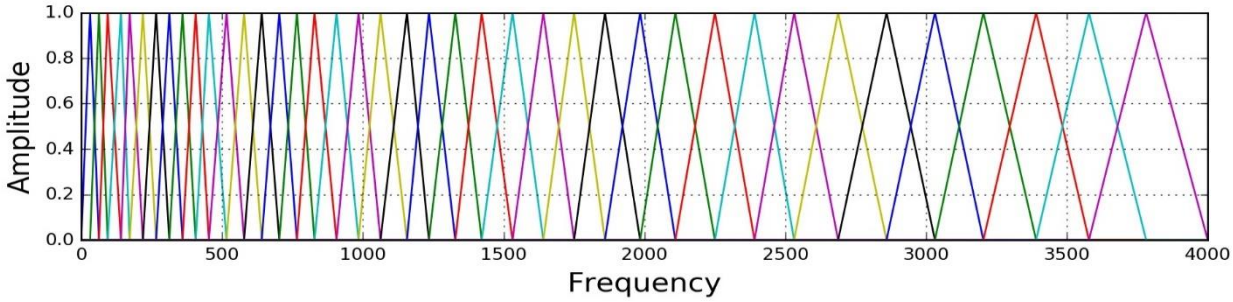



**Figure 2. 25 : Filter Bank on a Mel-Scale**

Cela peut être modélisé par l'équation suivante :

$$H_m(k) = \begin{cases} 0 & k < f(m-1) \\ \frac{k-f(m-1)}{f(m)-f(m-1)} & f(m-1) \le k < f(m) \\ 1 & k = f(m) \\ \frac{f(m+1)-k}{f(m+1)-f(m)} & f(m) < k < f(m+1) \\ 0 & k > f(m+1) \end{cases} \quad \text{Éq 17}$$

Après application du banc de filtres sur le spectre de puissance (périodogramme) du signal, on obtient le spectrogramme suivant :

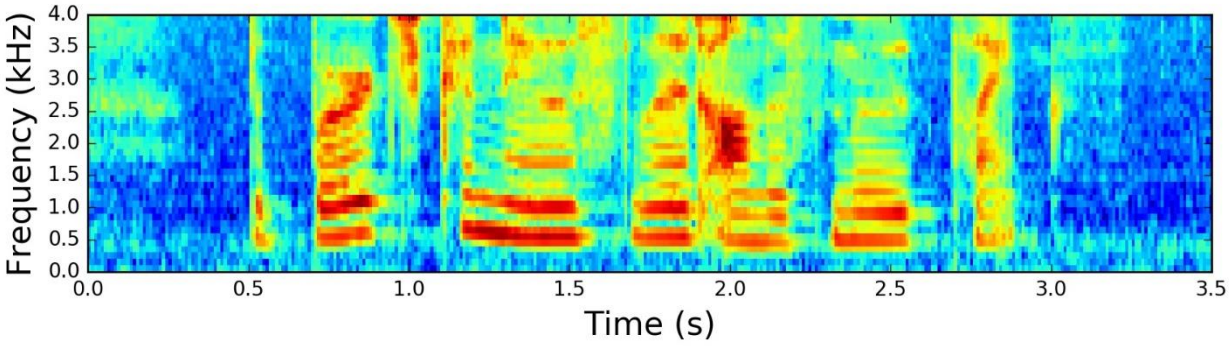



**Figure 2. 26 : Spectrogramme du signal**

#### II.8.3.1.5 Le logarithme

Le logarithme est utilisé dans le calcul des coefficients cepstraux de fréquence Mel (MFCC) pour transformer une multiplication en une addition. Les sorties logarithmiques du banc de filtres sont produites et multipliées par 20 pour obtenir des enveloppes spectrales en décibels. Les MFCC sont obtenus en prenant la transformée en cosinus discrète (DCT) de l'enveloppe spectrale. Les coefficients du cepstre sont obtenus comme le ième coefficient MFCC, où i=1, 2, ..., L, et L est le nombre de coefficients MFCC que nous voulons calculer. Le logarithme de l'énergie est calculé sur tout le segment pour coder la caractéristique globale du signal [59].

#### II.8.3.1.6 Transformée discrète en cosinus (DCT)

La transformée en cosinus discrète (DCT) est utilisée dans le processus d'extraction MFCC pour transformer l'enveloppe spectrale en un ensemble de coefficients cepstraux. La DCT est utilisée à la place de la transformée de Fourier inverse car elle a l'avantage d'être une transformée réel-réel et les coefficients résultants sont décorrélés les uns des autres. Le DCT est appliqué aux sorties logarithmiques du banc de filtres qui sont produites et multipliées par 20 pour obtenir des enveloppes spectrales en décibels. Les coefficients du cepstre sont obtenus comme le ième coefficient MFCC, où i=1, 2,...,L, et L est le nombre de coefficients MFCC que nous voulons calculer.[58,60,61]

Après avoir calculé la DFT et le spectre de magnitude logarithmique (pour obtenir le cepstre réel), nous calculons les sorties du banc de filtres, puis utilisons une transformée en cosinus discrète :

$$c(n) = \sum_{m=0}^{M-1} S(m)\cos\left(\pi n(m - \frac{1}{2})/M\right) \qquad \text{Éq 17}$$

Où

$$s(m) = \ln\left(\sum_{k=0}^{N-1} |X_\alpha(k)|^{2H_m}(k)\right) \quad 0 < m \leq M \qquad \text{Éq 18}$$

#### II.8.3.1.7 Puissance spectrale

La puissance spectrale, ou spectre de puissance, n'est pas directement utilisée dans le calcul des coefficients cepstraux de fréquence Mel (MFCC). Au lieu de cela, le calcul MFCC consiste à prendre les sorties logarithmiques du banc de filtres et à les multiplier par 20 pour obtenir des enveloppes spectrales en décibels. Les MFCC sont alors obtenus en prenant la transformée en cosinus discrète (DCT) de l'enveloppe spectrale [62].

$$s[k] = |X[k]|^2 \qquad \text{Éq 19}$$

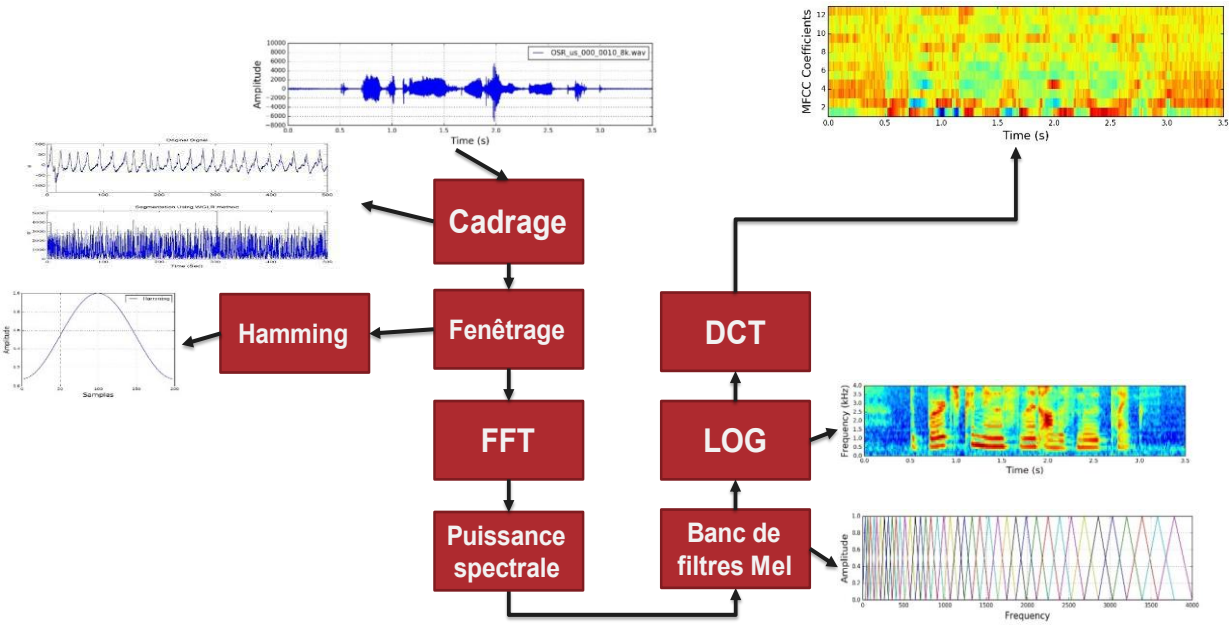



**Figure 2. 27 : Etapes de calcul de la matrice MFCC**

## II.9 Application de la méthode MFCC sur les signaux acoustiques

### II.9.1 Base de données choisie

Pour tester notre méthode de classification des défauts de roulement, nous avons choisi des données acoustiques et vibratoires des cas de défaut de la pompe centrifuge, de l'installation de banc d'essai, du capteur et du dispositif d'acquisition de données situés au Laboratoire de métrologie de précision, Département de génie mécanique de l'Institut d'ingénierie et de technologie de Sant Longowal Longowal, Inde.

#### II.9.1.1 Banc d'essai

L'expérience a été réalisée sur une pompe centrifuge monobloc de marque CRI (modèle : ACM-0 (AF)) fonctionnant à une vitesse de 43 Hz. La spécification de la pompe centrifuge

est donnée dans le **Tableau 2. 3**. Le schéma du banc d'essai est illustré à la **Figure 2. 28** une photographie typique du banc d'essai est présentée à **Figure 2. 29.** Le banc d'essai se compose d'un rotor soutenu par deux paliers (Roulement 1 et Roulement 2). Le roulement le plus proche de la roue est désigné comme le roulement 1 (est le roulement d'essai) et porte le numéro de roulement 6203-ZZ. Les spécifications du roulement d'essai sont données dans le tableau 2. Le roulement éloigné de la turbine est désigné comme roulement 2 et porte le numéro 6202-ZZ. La roue est en porte-à-faux par rapport à l'arbre du rotor et est maintenue dans le carter de la roue. La roue se compose de 3 aubes rotatives. Les aubes rotatives aspirent le fluide axialement à travers l'œil de la turbine et transmettent de l'énergie cinétique au fluide, faisant s'écouler le fluide vers l'extérieur dans la direction radiale à travers le carter. [63,37]

**Tableau 2. 3 : Spécification de la pompe centrifuge**

| | |
|---|---|
| Source de tension | 230/240 V |
| Puissance du moteur | 373 Watt |
| Décharge | 1.61 liter/sec |
| Type de roue | Fermée |
| Diamètre de la roue | 118.88 mm |
| Aubes de roue | 3 |

**Tableau 2. 4: Spécification du roulement à billes profond à une rangée à l'étude (NBC : 6203ZZ)**

| Diamètre du pas (D)in mm | Diamètre (bague intérieure) | Diameter (Bague extérieure) | Diamètre de bille (d) in mm | Number d'éléments roulants (N) | Contact angle ($\varphi^{\circ}$) |
|---|---|---|---|---|---|
| 28.5 | 17 | 40 | 6.74 | 9 | 0 |

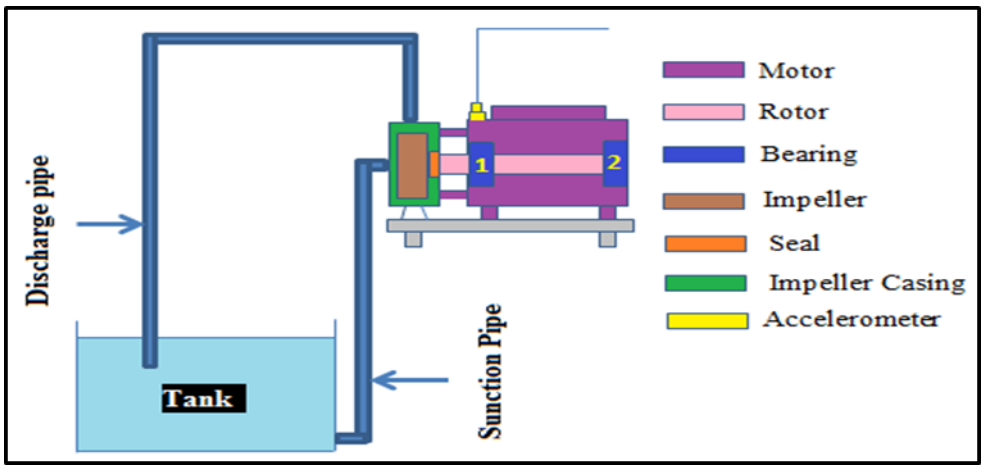



**Figure 2. 28 : Schéma du banc d'essai avec accéléromètre**

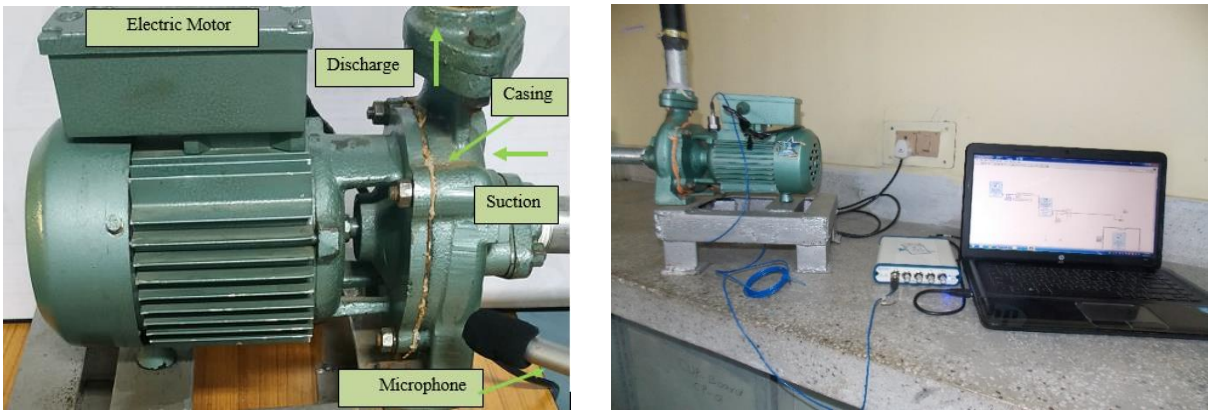



**Figure 2. 29 : Une photographie typique du banc d'essai (A) avec capteur acoustique (B)**

**A** **B**

- État des défauts : L'étude est réalisée pour la pompe dans différentes conditions comme spécifié dans le. Des photographies typiques des composants défectueux sont présentées dans la figure suivant :

**Tableau 2. 5 : Différentes condition de pompe à l'étude**

| S. No | Conditions à étude |
|---|---|
| 1 | Defect free (Sans défaut). |
| 2 | Broken impeller (roue cassé). |
| 3 | Clogged impeller (roue obstrué). |
| 4 | Roulement avec défaut de bague intérieure (IR). |
| 5 | Roulement avec défaut de bague extérieure (OR). |

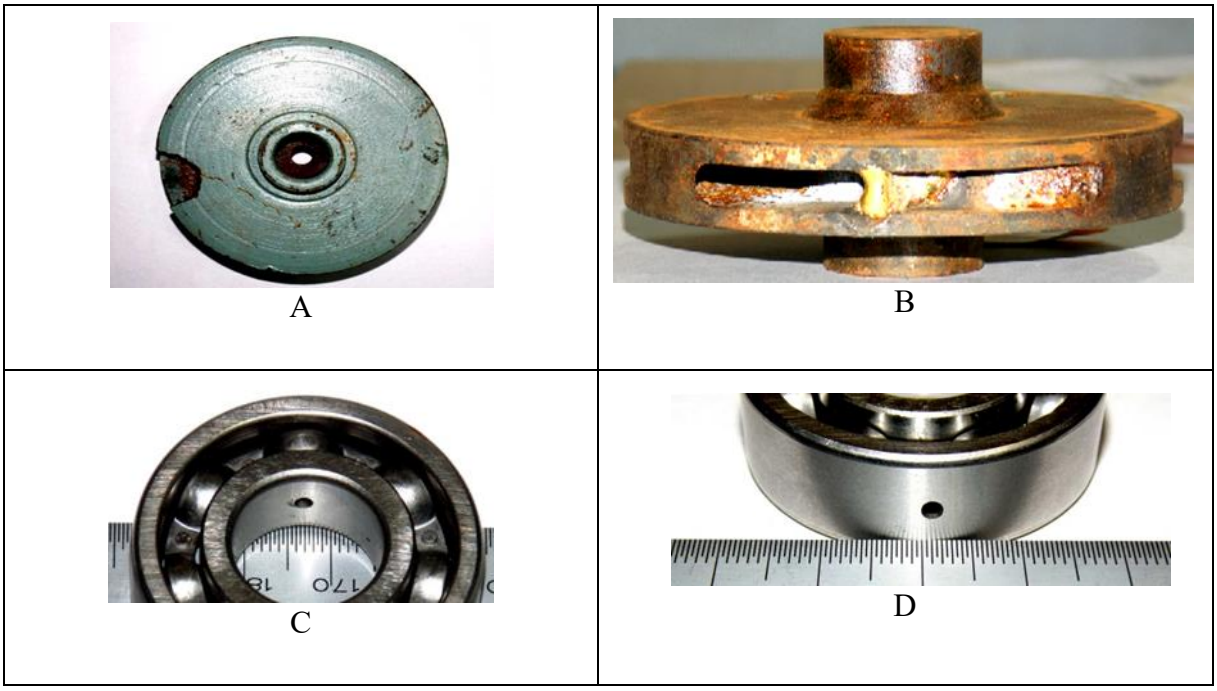



**Figure 2. 30 : Photographies du composant défectueux à l'étude (A) Roue cassée (BI) (B) Roue obstruée (CI) (C) Roulement avec défaut de bague intérieure (IR) (D) Roulement avec défaut de bague extérieure (OR)**

### II.9.2 Le Modèle CNN

Pour profiter des avantages offerts par l'utilisation du Deep Learning, nous avons décidé de créer notre propre modèle CNN. Pour ce faire, nous avons fait demi-tour au langage de programmation Python et à diverses bibliothèques de Deep Learning propose.

#### II.9.2.1 Python

Python est un langage de programmation polyvalent de haut niveau largement utilisé dans divers domaines tels que le développement Web, l'informatique scientifique, l'intelligence artificielle et l'apprentissage automatique. Python met l'accent sur la lisibilité du code avec l'utilisation d'une indentation significative via la règle hors-jeu. Il prend en charge plusieurs paradigmes tels que la programmation orientée objet, procédurale (impérative), fonctionnelle, structurée et réflexive. Python possède une communauté importante et active qui fournit une assistance, des didacticiels et des ressources pour les développeurs débutants et expérimentés. Il existe plusieurs plates-formes en ligne telles que LearnPython.org, Codecademy et GeeksforGeeks qui proposent des didacticiels Python interactifs gratuits pour les personnes qui souhaitent apprendre Python. Le site officiel de Python.org fournit de la documentation pour la bibliothèque standard de Python ainsi que des tutoriels et des guides. La Python Software Foundation promeut, protège et fait progresser le langage de programmation Python et soutient la croissance d'une communauté diversifiée et internationale de programmeurs Python.[64]

### II.9.2.2 TensorFlow et Keras

TensorFlow et Keras sont tous deux des modules d'apprentissage automatique utilisés dans le domaine de la science des données. TensorFlow est un framework qui offre des API de haut et de bas niveau, tandis que Keras est une API de haut niveau qui s'exécute au-dessus de TensorFlow, CNTK et Theano. Keras est conçu pour être modulaire, rapide et facile à utiliser, et il est parfait pour des implémentations rapides. D'autre part, TensorFlow est idéal pour la recherche en apprentissage profond et les réseaux complexes. TensorFlow peut être utilisé pour les modèles hautes performances, tandis que Keras peut être utilisé pour les modèles peu performants. Keras dispose d'une API conviviale qui facilite la compréhension des débutants, tandis que TensorFlow a une meilleure représentation graphique pour des données plutôt que toute autre plate-forme supérieure. TensorFlow offre un écosystème flexible et complet de ressources communautaires, de bibliothèques et d'outils qui facilitent la création et le déploiement d'applications de machine learning. [65]

### II.9.2.3 Librosa

Librosa est une bibliothèque Python pour l'analyse des signaux audio et musicaux. Il fournit des fonctionnalités pour l'extraction de caractéristiques, le suivi des battements, l'estimation de la hauteur, etc. Librosa est utilisé pour analyser et traiter les signaux audios dans diverses applications telles que la génération de musique, la reconnaissance automatique de la parole, la prédiction du genre, la prédiction du genre musical et l'identification vocale. Il est facile à utiliser et prend en charge divers formats de fichiers audios tels que .wav et .mp3. Librosa peut être utilisé pour les algorithmes de détection de hauteur et la détection d'activité vocale. C'est un outil utile pour créer des systèmes de recherche d'informations musicales. [66]

## II.10 Conclusion

Dans ce chapitre, nous avons discuté d'une méthode d'extraction de caractéristiques à partir de signaux acoustiques. Cette méthode a été développée pour convertir des données audios unidimensionnelles en une matrice bidimensionnelle ou une représentation d'image. Nous avons également exploré différents types de réseaux de neurones, dont le réseau de neurones convolutifs (CNN), et ses caractéristiques spécifiques. Pour notre projet, nous nous sommes appuyés sur le réseau CNN pour analyser les signaux acoustiques et effectuer les tâches souhaitées.

# Chapitre III : implémentation, test et résultats

## III.1 Introduction

Dans ce chapitre, nous aborderons la mise en œuvre pratique de notre méthode de détection et de classification des défauts de roulement. Tout d'abord, nous allons commencer par mettre en place la base de données nécessaire pour former et tester nos modèles. Cette base de données contiendra les enregistrements audio des roulements intacts et défectueux, avec les annotations correspondantes.

Ensuite, nous appliquerons notre méthode d'extraction des caractéristiques MFCC aux enregistrements audios de la base de données.

Une fois les MFCC extraits, nous procéderons à la construction de réseaux convolutifs (CNN) pour identifier et classer les défauts de roulement. Réseaux CNN. Nous formerons plusieurs CNN à l'aide de nos données d'entraînement et évaluerons leurs performances à l'aide de nos données de test. A partir des résultats obtenus, nous sélectionnerons le meilleur réseau pour une utilisation ultérieure dans notre étude expérimentale finale.

Une fois la phase de test terminée sur l'ordinateur, nous passerons à l'exécution sur la carte Raspberry Pi 3b+. Nous allons reproduire les mêmes étapes pour préparer les données, extraire les MFCC et les afficher sur le même modèle de choix sur le Raspberry Pi. Cela nous permettra de comparer les performances des modèles entre un environnement informatique et un environnement Raspberry Pi 3b+, ce qui est essentiel pour évaluer la faisabilité d'une exécution en temps réel sur un appareil embarqué.

## III.2 La base de donnes

Dans la base de données, nous avons cinq classes qui correspondent aux différents types de défauts de roulement. Chaque classe représente une catégorie spécifique de défaut. Les cinq classes sont les suivantes :

1. Classe de roulement Sans défaut ;
2. Classe de défaut interne ;
3. Classe de défaut externe ;
4. Classe de Roue cassé ;
5. Classe de Roue obstrué ;

Chaque enregistrement de la base de données est associé à l'une de ces cinq classes, ce qui permet à nos modèles de classification de distinguer et d'identifier le type de défaut de roulement présent dans les données d'entrée. Chacun des signaux acoustiques des défauts de

roulements est enregistré pendant une période de 14s avec une fréquence d'échantillonnage de 70KHz.

### III.3 Segmentation les signaux

Dans la base de données, chaque classe contient un signal acoustique composé de 943719 points de données. Pour faciliter le traitement et l'analyse, nous avons divisé chaque signal acoustique en segments de 4000 points chacun. Cette segmentation permet de créer des échantillons plus gérables pour l'extraction de caractéristiques et la formation de modèles. En conséquence, pour chaque classe, nous avons 470 segments, chaque segment contenant 4000 points de données. Cette segmentation permet une manipulation plus facile et une analyse plus approfondie dans le processus de détection et de classification des défauts de roulement.

Après avoir appliqué la segmentation sur le signal acoustique du roulement sans aucun défaut, la **Error! Reference source not found.** affiche le résultat du premier segment. Ce segment est représenté sur **Error! Reference source not found.**, montrant la forme d'onde ou le spectrogramme de la partie segmentée du signal acoustique. Les figures représentent visuellement l'étape initiale de division du signal acoustique en segments et illustrent le segment spécifique considéré.




**Figure 3. 1 : signal de cas roulement Sans défaut**

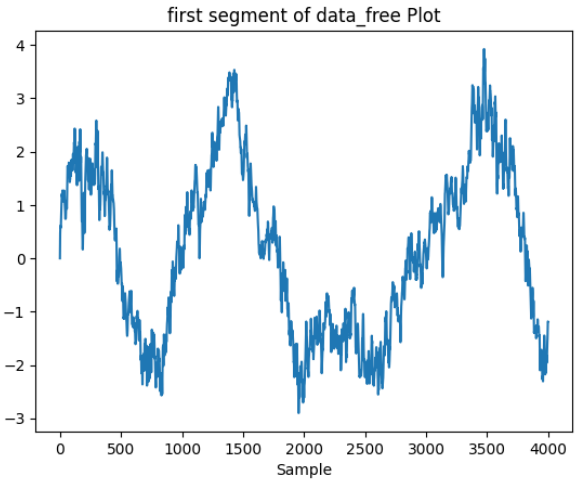



**Figure 3. 2 : 1er segment de signal cas roulement Sans défaut**

### III.4 Traitement MFCC sur les signaux acoustiques

Les trames résultantes sont ensuite converties en images en utilisant la méthode MFCC. Les paramètres choisis pour le prétraitement sont :

1. Taille des trames N = 4000 points ;
2. Taille des segments L = 2048 points ;
3. Taille des sauts de chevauchements entre chaque segment H = 512 points ;

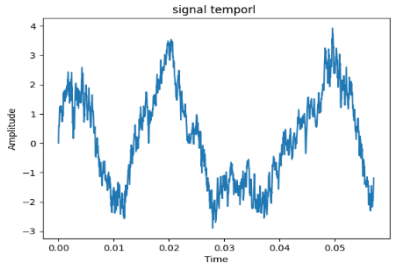



**Figure 3. 3 : signal temporel**

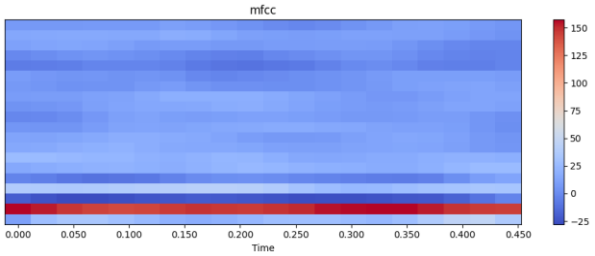



**Figure 3. 4 : matrice MFCC**

La **Figure 3. 4** affiche le résultat du traitement MFCC effectué sur la première trame du signal acoustique d'un roulement sans défaut **Figure 3. 3.** Les paramètres mentionnés plus loin dans le texte ont été utilisés pour ce traitement. La sortie résultante est une image ou une matrice de taille fixe, en particulier 20 lignes et 20 colonnes. Cette image ou matrice représente les coefficients MFCC extraits du signal acoustique et fournit une représentation visuelle des caractéristiques extraites à l'aide du traitement MFCC pour la trame donnée.

## III.5 Décomposition

Après avoir converti 2350 signaux acoustiques du dataset en image MFCC, il en résulte un nouveau dataset intermédiaire composé des images MFCC et des différentes classes de défauts associés.

Le reste des signaux acoustiques (51) n'ont pas été converti et servirons pour tester l'efficacité de l'implémentation finale, prétraitement MFCC et classificateur CNN sur ordinateurs

**Tableau 3. 1 : Partage des images du la base de données des images MFCC**

<table>
<tr><td colspan="4">Ensemble des images MFCC (2401)</td></tr>
<tr><td>Training<br>80 %</td><td>Validation<br>10%</td><td>Test<br>10%</td><td>Test final</td></tr>
<tr><td colspan="3">Images utiliser pour entrainer le modèle CNN (2350)</td><td>51 images jamais vus par le modèle</td></tr>
</table>

## III.6 Modèle CNN

Pour exploiter les avantages offerts par l'apprentissage en profondeur, nous avons décidé de développer notre propre modèle CNN. À cette fin, nous avons utilisé le langage de programmation Python ainsi que la gamme diversifiée de bibliothèques d'apprentissage en profondeur disponibles dans son écosystème. Cela nous a permis de tirer parti des puissants outils et fonctionnalités fournis par ces bibliothèques pour construire et former notre modèle CNN personnalisé. En développant notre propre modèle, nous avons pu l'adapter spécifiquement à notre tâche d'identification et de classification des défauts de roulement à partir de signaux acoustiques, optimisant ainsi ses performances pour notre application unique.

## III.7 Le modèle CNN proposé

Afin de classer les défauts des roulements à billes, nous avons conçu un modèle CNN compact composé de deux couches convolution elles et de deux couches entièrement connectées. Ce modèle a été délibérément gardé de petite taille avec un nombre limité de paramètres de couche cachés. Le choix de ce modèle a été principalement motivé par la taille des images d'entrée, qui sont relativement petites (20 lignes et 20 colonnes). De plus, nous avons cherché à développer un modèle léger qui pourrait être déployé efficacement sur une plateforme embarquée. En optant pour une architecture compacte, nous avons cherché à trouver un équilibre entre les performances du modèle et l'utilisation des ressources, en assurant une

classification efficace des défauts tout en tenant compte des contraintes de la plate-forme de déploiement cible.

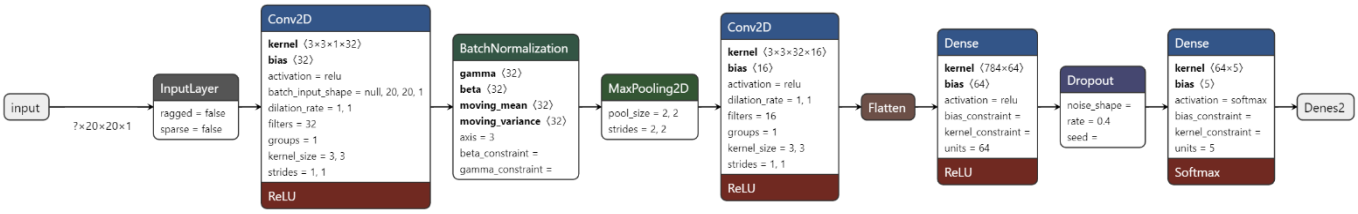



**Figure 3. 5 : modèle CNN proposé**

## III.8 Résultats de classification

Initialement, nous avons séparé les données en segments, chaque segment étant composé de 1200 points de données. Cependant, lors de la formation de notre modèle à l'aide de ces segments plus petits, nous avons observé un faible taux de précision de 40 % et un taux de perte élevé de 60 %, ce qui ne répondait pas aux critères de performance souhaités. Pour résoudre ce problème, nous avons décidé d'augmenter la taille du segment à 4 000 points de données. Cet ajustement a été effectué dans le but d'améliorer la précision du modèle et de réduire les cas d'abandon pendant la formation. En utilisant des segments plus grands, nous nous attendions à capturer des informations plus complètes et à améliorer le processus d'apprentissage, ce qui se traduirait par une meilleure précision et une réduction des pertes pendant la formation.

Pour le test (10% dataset), nous avons appliqué le modèle et donné un taux de précision impressionnant de 99.14 % et un taux de perte minimal de 1,02 %. Ces résultats exceptionnels indiquent l'exactitude et l'efficacité élevées du modèle.

La capacité du modèle à classer avec précision les images avec une telle précision et une perte minimale démontre sa robustesse et son adéquation à la tâche à accomplir.

**Tableau 3. 2 : résultat de prédiction de test**

| | Précision | Erreur absolue |
|---|---|---|
| Résultats | 0.9914 | 0.0102 |

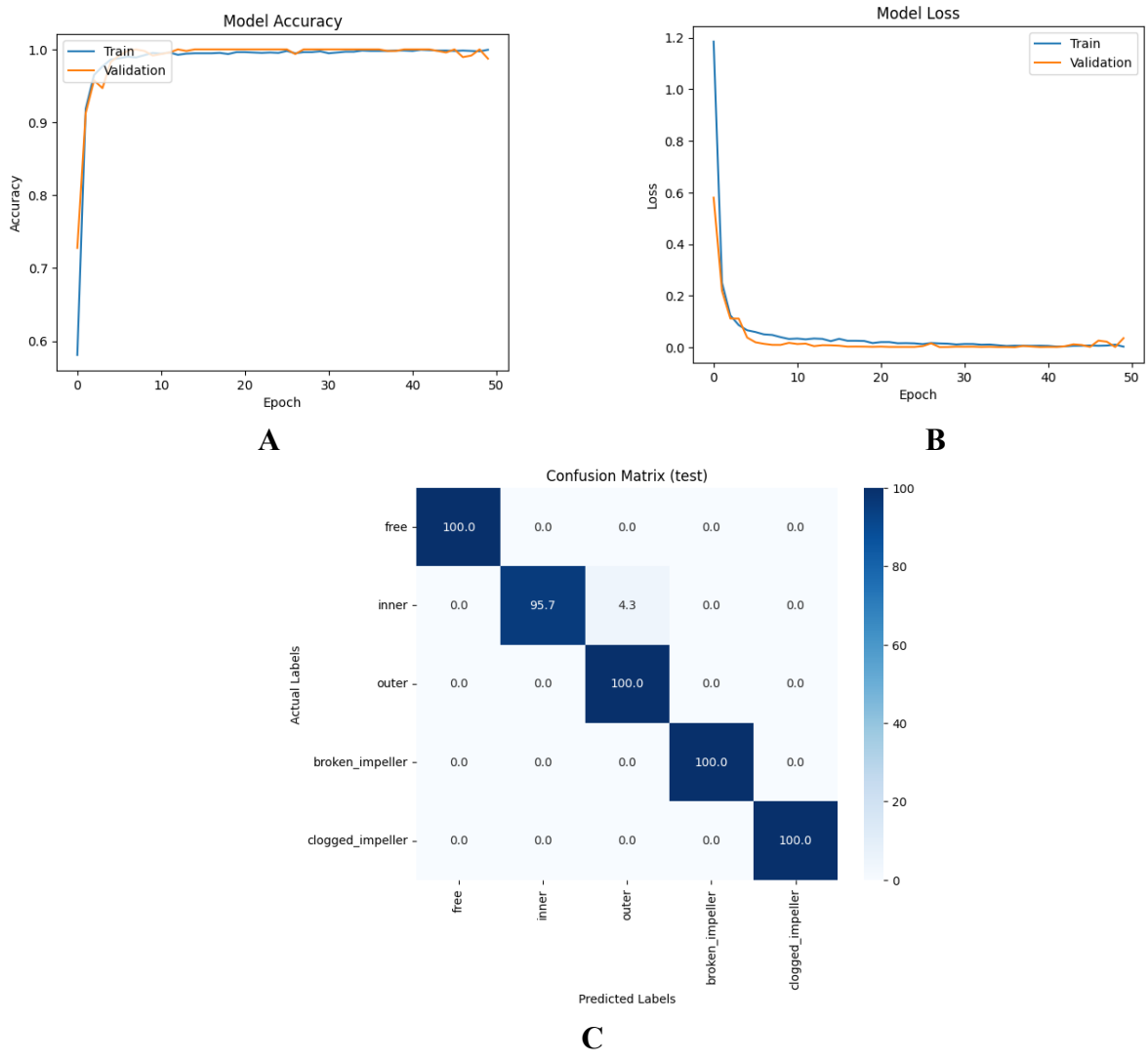



**Figure 3. 6 : Résultat du modèle CNN proposé**

**A : graphe de l'évolution de la précision, B : graphe de l'évolution de l'erreur , C : résultat de classification des signaux de tests (10% du dataset)**

Afin de valider et de confirmer la fiabilité et l'efficacité du modèle, nous avons effectué un test final sur un ensemble de 51 images préalablement préparées et jamais présentées au modèle lors de sa phase d'apprentissage. Ce test indépendant visait à certifier les capacités de généralisation du modèle, tel qu'évalué sur des données inédites. En examinant les performances du modèle sur cet ensemble de données nouveau et invisible, nous pouvons évaluer sa capacité à classer avec précision et à faire des prédictions sur des instances inédites. Le résultat de ce test sert de validation de la validité du modèle et de son efficacité dans la gestion des scénarios du monde réel.

Où il en est résulté :

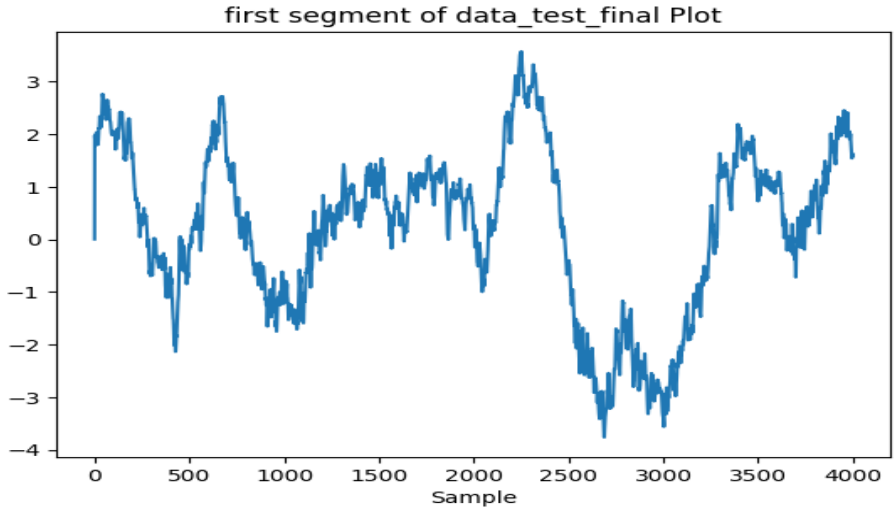



**Figure 3. 7 : premier segment des données de test finales**

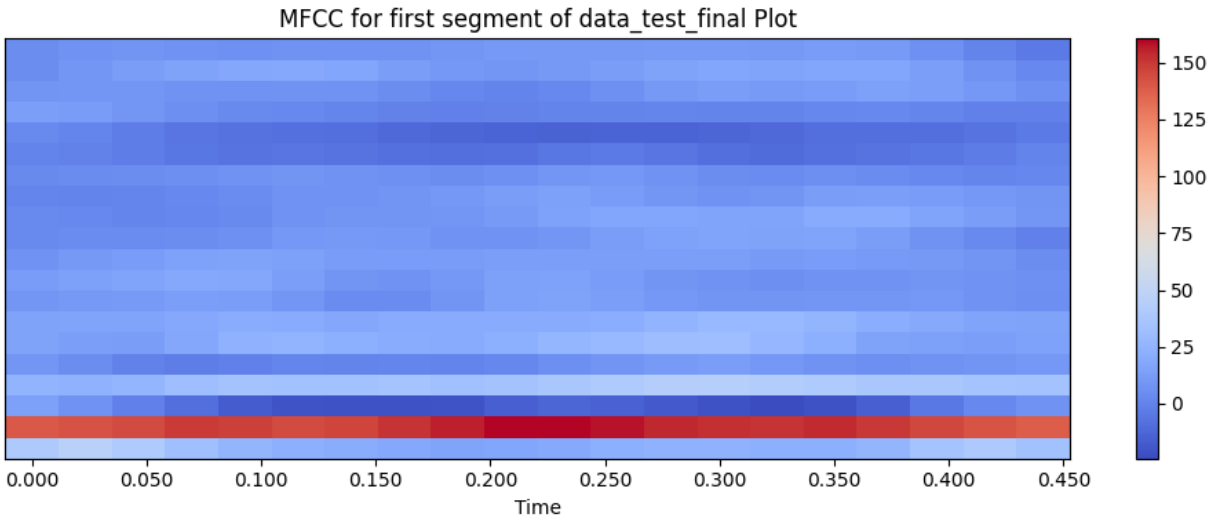



**Figure 3. 8 : Calculez les caractéristiques MFCC pour le premier segment des données de test finales.**

**A**

```
extract MFCC feature time: 0.6439721584320068 seconds
```

**B**

```
MFCC shape: (20, 20)
2/2 [==============================] - 1s 9ms/step
prediction time: 1.5294148921966553 seconds
Predictions complete.
```

```
['segment 1: Class: free, Percentage: 99.99219%',
 'segment 2: Class: free, Percentage: 100.00000%',
 'segment 3: Class: free, Percentage: 100.00000%',
 'segment 4: Class: free, Percentage: 99.99999%',
 'segment 5: Class: free, Percentage: 99.99954%',
 'segment 6: Class: free, Percentage: 100.00000%',
 'segment 7: Class: free, Percentage: 100.00000%',
 'segment 8: Class: free, Percentage: 99.99998%',
 'segment 9: Class: free, Percentage: 99.99999%',
 'segment 10: Class: free, Percentage: 99.99971%',
 'segment 11: Class: free, Percentage: 99.99999%',
 'segment 12: Class: free, Percentage: 99.99998%',
 'segment 13: Class: free, Percentage: 99.99995%',
 'segment 14: Class: free, Percentage: 99.99999%',
 'segment 15: Class: free, Percentage: 99.99998%',
 'segment 16: Class: free, Percentage: 100.00000%',
 'segment 17: Class: free, Percentage: 100.00000%',
 'segment 18: Class: free, Percentage: 99.99962%',
 'segment 19: Class: free, Percentage: 100.00000%',
 'segment 20: Class: free, Percentage: 100.00000%',
 'segment 21: Class: free, Percentage: 100.00000%',
 'segment 22: Class: free, Percentage: 99.99998%',
 'segment 23: Class: free, Percentage: 100.00000%',
 'segment 24: Class: free, Percentage: 99.99995%',
 'segment 25: Class: free, Percentage: 99.99986%',
 'segment 26: Class: free, Percentage: 99.99900%',
 'segment 27: Class: free, Percentage: 99.99992%',
 'segment 28: Class: free, Percentage: 99.99992%',
 'segment 29: Class: free, Percentage: 99.99994%',
 'segment 30: Class: free, Percentage: 99.99988%',
 'segment 31: Class: free, Percentage: 99.99986%',
 'segment 32: Class: free, Percentage: 99.99999%',
 'segment 33: Class: free, Percentage: 99.99993%',
 'segment 34: Class: free, Percentage: 100.00000%',
 'segment 35: Class: free, Percentage: 99.99999%',
 'segment 36: Class: free, Percentage: 99.99994%',
 'segment 37: Class: free, Percentage: 99.99996%',
 'segment 38: Class: free, Percentage: 100.00000%',
 'segment 39: Class: free, Percentage: 99.99999%',
 'segment 40: Class: free, Percentage: 100.00000%',
 'segment 41: Class: free, Percentage: 100.00000%',
 'segment 42: Class: free, Percentage: 100.00000%',
 'segment 43: Class: free, Percentage: 99.99983%',
 'segment 44: Class: free, Percentage: 100.00000%',
 'segment 45: Class: free, Percentage: 100.00000%',
 'segment 46: Class: free, Percentage: 99.99998%',
 'segment 47: Class: free, Percentage: 99.99990%',
 'segment 48: Class: free, Percentage: 99.99807%',
 'segment 49: Class: free, Percentage: 100.00000%',
 'segment 50: Class: free, Percentage: 99.99999%',
 'segment 51: Class: free, Percentage: 99.99981%']
```

**C**

**D**

**E**

**Figure 3. 9 : les résultats de test final**

**A : temps de traitement MFCC, B : temps de prédiction, C, D et E : résultat de classification avec le percentage**

Chaque segment comprend 4000 points de données.

Le temps de traitement et la prédiction pour les données de test finales :

- MFCC feature time : 0.6439 seconds.

- prediction time: 1.5294 seconds.

Nous avons utilisé une approche supplémentaire en convertissant le modèle du format h5 en tflite. Cela impliquait l'utilisation de TensorFlow Lite, qui est un cadre pour exécuter des modèles d'apprentissage automatique sur des appareils à ressources limitées. En convertissant le modèle au format tflite, nous avons obtenu des résultats similaires au modèle d'origine tout en réduisant considérablement le temps de traitement et la latence de prédiction pour les données de test finales. Cette optimisation nous a permis d'améliorer l'efficacité et les performances en temps réel du modèle, le rendant plus adapté au déploiement sur des appareils aux capacités de calcul limitées.

```
extract MFCC feature time: 0.7316725254058838
seconds
prediction time is 10.945100000071761 ms
```

**Figure 3. 10 : temps de traitement par modele tflite**

Où il a donné :

- MFCC feature time : 0.7316 seconds.
- prediction time is 10.94 ms.

TFLite donne le même résultat avec le modèle H5, mais il y a une différence de temps de calcul pour le MFCC et le temps de prédiction

Nous allons effectuer un autre test en utilisant un contenant 3 défauts, puis afficher les résultats obtenus.

```
extract MFCC feature time: 0.6549382209777832 seconds
```

**A**

```
MFCC shape: (20, 20)
2/2 [==============================] - 0s 8ms/step
prediction time: 0.22514081001281738 seconds
Predictions complete.
```

**B**

C

```
['segment 1: Class: inner, Percentage: 99.99976%',
 'segment 2: Class: outer, Percentage: 97.48288%',
 'segment 3: Class: outer, Percentage: 99.99996%',
 'segment 4: Class: outer, Percentage: 99.33851%',
 'segment 5: Class: free, Percentage: 99.99954%',
 'segment 6: Class: free, Percentage: 100.00000%',
 'segment 7: Class: free, Percentage: 100.00000%',
 'segment 8: Class: free, Percentage: 99.99998%',
 'segment 9: Class: free, Percentage: 99.99999%',
 'segment 10: Class: free, Percentage: 99.99971%',
```

D

```
'segment 11: Class: free, Percentage: 99.99999%',
'segment 12: Class: free, Percentage: 99.99998%',
'segment 13: Class: free, Percentage: 99.99995%',
'segment 14: Class: free, Percentage: 99.99999%',
'segment 15: Class: free, Percentage: 99.99998%',
'segment 16: Class: free, Percentage: 100.00000%',
'segment 17: Class: free, Percentage: 100.00000%',
'segment 18: Class: free, Percentage: 99.99962%',
'segment 19: Class: free, Percentage: 100.00000%',
'segment 20: Class: free, Percentage: 100.00000%',
'segment 21: Class: free, Percentage: 100.00000%',
'segment 22: Class: free, Percentage: 99.99998%',
'segment 23: Class: free, Percentage: 100.00000%',
```

E

```
'segment 24: Class: free, Percentage: 99.99995%',
'segment 25: Class: free, Percentage: 99.99986%',
'segment 26: Class: free, Percentage: 99.99900%',
'segment 27: Class: free, Percentage: 99.99992%',
'segment 28: Class: free, Percentage: 99.99992%',
'segment 29: Class: free, Percentage: 99.99994%',
'segment 30: Class: free, Percentage: 99.99988%',
'segment 31: Class: free, Percentage: 99.99986%',
'segment 32: Class: free, Percentage: 99.99999%',
'segment 33: Class: free, Percentage: 99.99993%',
'segment 34: Class: free, Percentage: 100.00000%',
'segment 35: Class: free, Percentage: 99.99999%',
'segment 36: Class: free, Percentage: 99.99994%',
'segment 37: Class: free, Percentage: 99.99996%',
'segment 38: Class: free, Percentage: 100.00000%',
'segment 39: Class: free, Percentage: 99.99999%',
'segment 40: Class: free, Percenta
```

F

```
'segment 41: Class: free, Percentage: 100.00000%',
'segment 42: Class: free, Percentage: 100.00000%',
'segment 43: Class: free, Percentage: 99.99983%',
'segment 44: Class: free, Percentage: 100.00000%',
'segment 45: Class: free, Percentage: 100.00000%',
'segment 46: Class: free, Percentage: 99.99998%',
'segment 47: Class: free, Percentage: 99.99990%',
'segment 48: Class: free, Percentage: 99.99807%',
'segment 49: Class: free, Percentage: 100.00000%',
'segment 50: Class: free, Percentage: 99.99999%',
'segment 51: Class: free, Percentage: 99.99981%']
```

**Figure 3. 11 : résultat de prédiction avec percentage pour chaque image (segment) multi class**

**A : temps de traitement MFCC, B : temps de prédiction, C, D, E et F : résultat de classification avec le percentage**

Le temps de traitement et la prédiction pour les données de test :

- MFCC feature time: 0.6549 seconds.
- prediction time: 0.2251 seconds.

Nous allons effectuer le même test en utilisant le modèle TFLite et afficher les résultats obtenus.

```
extract MFCC feature time: 0.7901084423065186
seconds
prediction time is 10.197199999993245 ms
```

**Figure 3. 12 : Temps de traitement par modèle tflite**

Le temps de traitement et la prédiction pour les données de test :

- MFCC feature time: 0.6549 seconds.
- prediction time: 0.2251 seconds.

### III.9 Implémentation

Les étapes de l'implémentation sont représentées dans l'organigramme de la **Figure 3. 13**

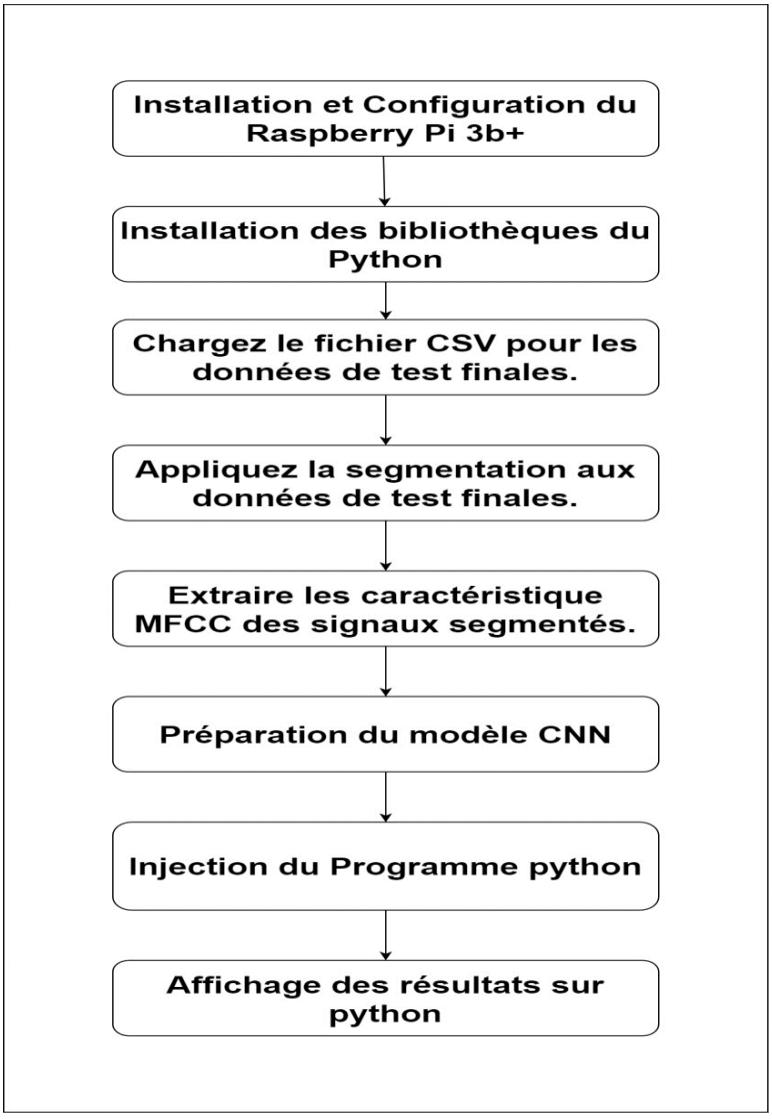



**Figure 3. 13 : Organigramme des étapes d'implémentation**

### III.9.1 Raspberry Pi 3B+

Le Raspberry Pi 3 modèle B + est le dernier produit de la gamme Raspberry Pi 3, doté d'un processeur quad core 64 bits fonctionnant à 1,4 GHz, d'un réseau local sans fil bi-bande 2,4 GHz et 5 GHz, Bluetooth 4.2 / BLE, Ethernet plus rapide et PoE capacité via un HAT PoE séparé Le LAN sans fil bi-bande est livré avec une certification de conformité modulaire, permettant à la carte d'être conçue dans des produits finaux avec des tests de conformité LAN sans fil considérablement réduits, améliorant à la fois le coût et le délai de mise sur le marché.



**Figure 3. 14 : Raspberry Pi 3 Model B+**

Le Raspberry Pi 3 modèle B+ conserve la même empreinte mécanique que le Raspberry Pi 2 modèle B et le Raspberry Pi 3 modèle B. [68]

### III.9.2 Tensorflow

Tensor Flow Lite est un ensemble d'outils qui permet l'apprentissage automatique sur appareil en aidant les développeurs à exécuter leurs modèles sur des appareils mobiles, intégrés et périphériques. Il est utilisé pour convertir des modèles Tensor Flow pré-formés en un format de modèle d'apprentissage automatique plus petit et plus efficace, optimisé pour l'apprentissage automatique sur l'appareil. Tensor Flow Lite est optimisé pour répondre à cinq contraintes clés : la latence, la confidentialité, la connectivité, la taille et la consommation d'énergie. Il est utilisé pour convertir des modèles pré-formés en un format portable efficace spécial appelé FlatBuffers, qui offre plusieurs avantages par rapport au format de modèle de tampon de protocole de TensorFlow, tels qu'une taille réduite et une inférence plus rapide. TensorFlow Lite est utilisé pour convertir des modèles pré-entraînés du format "h5" au format "tflite" qui est plus léger et adapté aux appareils embarqués comme Raspberry Pi. Il est également livré avec une boîte à outils pour la quantification post-formation et la formation des modèles prenant en compte la quantification, avec des options pour accélérer l'interface intégrée ou assurer une grande précision lors de l'utilisation de modèles quantifiés. [69]

### III.9.3 Préparation du modèle

Tout d'abord, nous avons converti le modèle Keras du format CNN, initialement enregistré sous ".h5", au format ".tflite" pour le rendre compatible avec le Raspberry Pi. Ce processus de conversion était nécessaire pour adapter le modèle au Raspberry Pi avec succès.

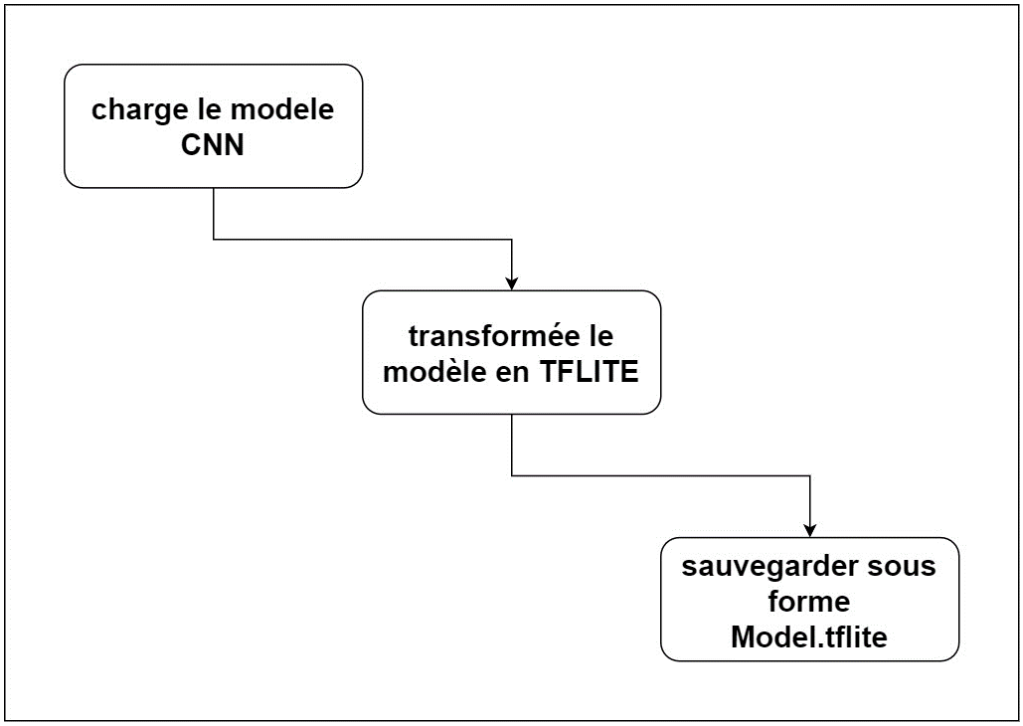



**Figure 3. 15 : Préparation des données**

Afin de charger notre modèle CNN personnalisé, nous avons utilisé le fichier de commande "code final". Pour rendre le modèle compatible avec Raspberry Pi, nous avons utilisé la bibliothèque "tflite.TFLiteConverter.from_keras_model". À la suite de ce processus de conversion, le modèle Keras a été converti de son format d'origine ".h5" au format ".tflite". Le modèle converti a été enregistré dans un fichier "model.tflite" créé à partir du modèle converti.

### III.9.4 Résultat d'implémentation

Tout d'abord, nous avons importé les bibliothèques nécessaires pour obtenir le résultat souhaité. Ensuite, nous avons fourni un fichier CSV extrait de la base de données, qui a été utilisé à des fins de formation pour effectuer la numérisation.

Après avoir fourni le fichier CSV segmenté, nous l'avons converti en signaux audio représentés par les coefficients cepstraux de fréquence Mel (MFCC). Pour effectuer cette conversion, nous avons utilisé la bibliothèque de traitement audio populaire appelée **librosa.** Cette bibliothèque a simplifié le processus d'extraction de MFCC en fournissant des paramètres tels que la fréquence d'échantillonnage, N_mfcc et la longueur de saut.

A l'aide de la commande suivante, nous avons extrait le MFCC sur le Raspberry Pi :

mfcc = librosa.feature.mfcc() [70]

Pour nos calculs, nous fixons les valeurs suivantes :

- Fréquence d'échantillonnage : 70 000 Hz
- Longueur de saut : 210
- N_mfcc : 20

La sortie résultante était la suivante :

Pour remodeler la matrice MFCC, nous avons utilisé l'instruction

`y6 = mfcc1.reshape(mfcc.shape[0], mfcc.shape[1], mfcc.shape[2], 1)`

Cette commande modifie la forme du tableau selon les dimensions spécifiées.

Dans ce cas, la forme est changée en

`(mfcc. shape[0], mfcc.shape[1], mfcc.shape[2], 1)`, où

`mfcc.shape[0]` Représente le nombre de trames

`mfcc.shape[1]` indique le nombre de coefficients par trame,

`mfcc.shape[2]` indique le nombre de canaux et l'ajout de `1` introduit une nouvelle dimension de canal.

Étant donné les paramètres fournis :

- Nombre des segments : 51
- Nombre de coefficients : 20
- Nombre de canaux : 20
- Le `1` une nouvelle dimension de canal.

Par conséquent, `Y6.shape = (51, 20, 20, 1) `.

Après avoir extrait le MFCC, nous l'avons introduit dans le modèle TFLite pour faire des prédictions et obtenir des résultats pour le signal audio ainsi que les pourcentages associés.

De plus, des efforts considérables ont été investis dans la mesure du temps nécessaire au processus de prédiction.

La procédure de prédiction consistait à comparer les 51 image de MFCC au modèle, ce qui a donné les défauts trouvés dans chaque image ainsi que leurs pourcentages respectifs. Le

résultat obtenu a ensuite été analysé pour déterminer la cause de tout dysfonctionnement ou pour confirmer l'exactitude du résultat.

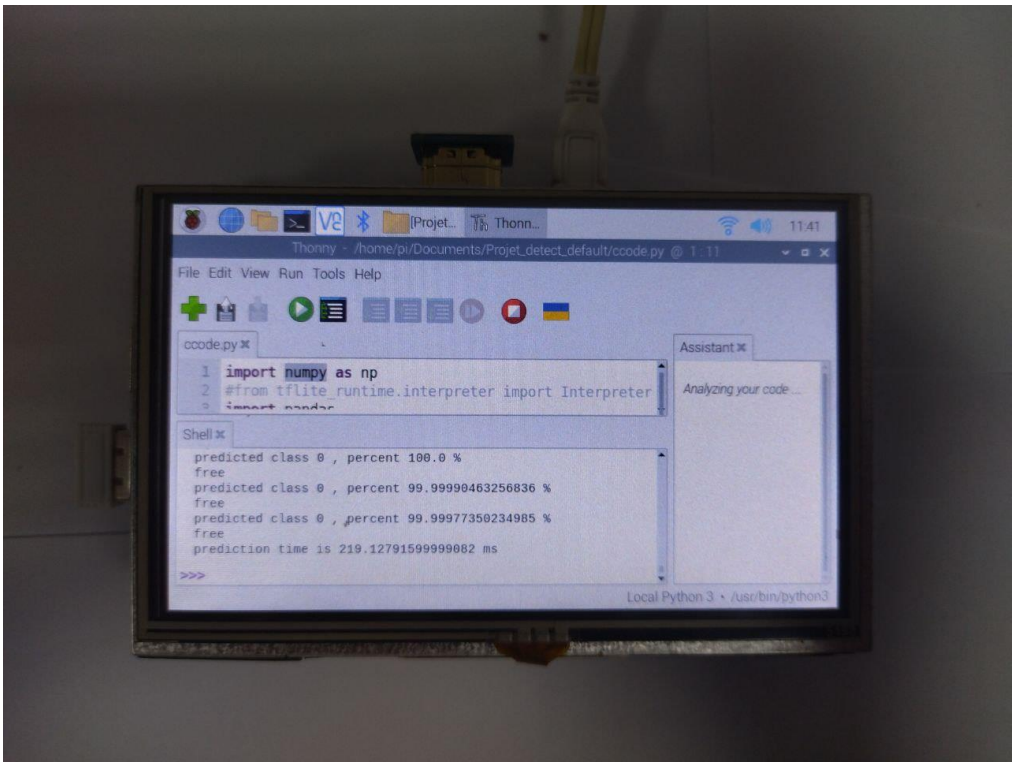



**Figure 3. 16 : affichage sur Écran HDMI 5 pouces (Annexe D)**

```
mfcc featurs time is 24280.33776600023 ms
(51, 20, 20, 1)
INFO: Created TensorFlow Lite XNNPACK delegate for CPU.
[ 1 20 20  1]
predicted class 0 , percent 99.99219179153442 %
free
predicted class 0 , percent 100.0 %
free
predicted class 0 , percent 100.0 %
```

**A**

```
predicted class 0 , percent 99.99219179153442 %
free
predicted class 0 , percent 100.0 %
free
predicted class 0 , percent 100.0 %
free
predicted class 0 , percent 99.99998807907104 %
free
```

**B**

```
predicted class 0 , percent 100.0 %
free
predicted class 0 , percent 99.99998807907104 %
free
predicted class 0 , percent 99.99980926513672 %
free
prediction time is 350.4476390007767 ms
```

**C**

**Figure 3. 17 : Résultat de prédiction**

**A : temps de traitement MFCC, B : résultat de classification avec le percentage, C : temps de prédiction**

Le temps de traitement et la prédiction pour les données de test final :

- MFCC feature time: 2.4280 seconds.
- prediction time: 350.44 ms.

Nous allons effectuer un autre test en utilisant un contenant 3 défauts, puis afficher les résultats obtenus.

**A**

```
mfcc featurs time is 19047.2699259999 ms
(51, 20, 20, 1)
INFO: Created TensorFlow Lite XNNPACK delegate for
[ 1 20 20  1]
predicted class 1 , percent 99.9997615814209 %
Inner
predicted class 2 , percent 97.48287200927734 %
Outer
```

**B**

```
predicted class 1 , percent 99.9997615814209 %
Inner
predicted class 2 , percent 97.48287200927734 %
Outer
predicted class 2 , percent 99.99996423721313 %
Outer
predicted class 2 , percent 99.33851361274719 %
Outer
```

**C**

```
predicted class 2 , percent 99.33851361274719 %
Outer
predicted class 0 , percent 99.99953508377075 %
free
predicted class 0 , percent 100.0 %
free
predicted class 0 ; percent 100.0 %
free
```

**D**

```
predicted class 0 , percent 99.99998807907104 %
free
predicted class 0 , percent 99.99980926513672 %
free
prediction time is 267.84513700022217 ms
```

**Figure 3. 18 : résultat de prédiction avec percentage pour chaque image (segment) multi class**

**A : temps de traitement MFCC, B et C : résultat de classification en pourcentage, D : temps de prédiction**

Le temps de traitement et la prédiction pour les données de test final :

- MFCC feature time: 1.9047 seconds.
- prediction time: 267.84 ms.

**Tableau 3. 3 : temps de traitement (test final)**

| | Pc | | Raspberry Pi |
|---|---|---|---|
| | Modèle h5 | Modèle tflite | tflite |
| **MFCC** | **0.6439 s** | **0.7316 s** | **2.4280 s** |
| **Prédiction** | **1.5294 s** | **0.1094 s** | **0.3504 s** |

**Tableau 3. 4 : temps de traitement (test multi classe)**

| | Pc | | Raspberry Pi |
|---|---|---|---|
| | Modèle h5 | Modèle tflite | tflite |
| **MFCC** | **0.6549 s** | **0.7901 s** | **1.9047 s** |
| **Prédiction** | **0.2251 s** | **0.1019 s** | **0.2678 s** |

## III.10 Conclusion

Dans ce chapitre, nous avons discuté du processus que nous avons suivi pour déterminer les résultats du test que nous avons effectué.

En comparant ces résultats, il est évident qu'il y avait une différence de performances entre l'environnement PC et l'environnement Raspberry Pi 3b+. Malgré cette différence, le modèle a tout de même atteint un taux de précision élevé de 99,99977 % sur les deux systèmes. Ces résultats sont très satisfaisants car ils permettent d'évaluer la faisabilité d'une exécution en temps réel sur un appareil embarqué comme le Raspberry Pi 3b+.

# Conclusion générale

# *Conclusion générale*

Dans ce travail, l'accent était mis sur l'utilisation d'images MFCC générées à partir de signaux acoustiques pour identifier les défauts de roulement. La procédure impliquait l'extraction de signaux acoustiques à partir d'une base de données fournie et l'application de techniques d'extraction de caractéristiques pour convertir les données audios en représentations raster bidimensionnelles.

Le projet a commencé par explorer le concept des pompes centrifuges, y compris leurs composants, leurs utilisations, leur fonctionnement et leurs avantages. L'analyse acoustique des machines tournantes, avec un accent particulier sur l'identification des problèmes de roulement, a également été abordée. Cette connaissance a servi de base pour une analyse et un traitement plus poussés des signaux acoustiques.

Une méthode d'extraction de caractéristiques des signaux acoustiques a été mise en œuvre à l'aide d'une architecture de réseau neuronal convolutionnel (CNN). Le CNN a été spécifiquement choisi pour sa capacité à traiter efficacement les représentations d'images des signaux audio.

Les performances du modèle développé ont été évaluées en mesurant le temps de calcul MFCC et le temps de prédiction à la fois sur un système informatique et sur un appareil Raspberry Pi 3B+. Alors que le Raspberry Pi 3B+ présentait des temps de traitement plus lents par rapport au système informatique, le modèle a atteint un taux de précision impressionnant de 99,99977 % sur les deux plates-formes. Ces résultats démontrent la faisabilité d'une exécution en temps réel sur des appareils embarqués tels que le Raspberry Pi 3B+.

En conclusion, ce projet a démontré avec succès le potentiel de l'utilisation d'images MFCC dérivées de signaux acoustiques pour l'identification des défauts de roulement. Les résultats mettent en évidence l'importance de l'analyse acoustique dans la maintenance et le diagnostic des machines tournantes. En outre, le travail met en évidence le caractère pratique de la mise en œuvre de telles techniques d'analyse à la fois sur les systèmes informatiques traditionnels et sur les dispositifs embarqués à ressources limitées. Le projet contribue au domaine de la maintenance prédictive et ouvre des possibilités d'applications futures dans des domaines similaires.

# Références bibliographiques

## Références bibliographiques


1. Chitale, S., et al., *Parameters affecting efficiency of centrifugal pump-A review.* 2021. **8**(6): p. 49-58.
2. zone, p. *Centrifugal Pump*. Available from: https://powerzone.com/resources/glossary/centrifugal-pump.
3. Volk, M., *Pump characteristics and applications*. 2013: CRC Press.
4. Karassik, I. and J.T. McGuire, *Centrifugal pumps*. 2012: Springer Science & Business Media.
5. ecademy, G. *PRINCIPAUX TYPES DE ROUE*. Available from: https://www.grundfos.com/content/dam/global/page-assets/learn/ecademy/pdfs/fr-36-module-3-principaux-types-de-roue.pdf.
6. *Cours pompe centrifuge: roue, diffuseur et volute*. Available from: https://www.azprocede.fr/Cours_GC/pompe_centrifuge_roue.html.
7. Harnoy, A., *Bearing design in machinery: engineering tribology and lubrication*. 2002: CRC press.
8. Engineering, A. *Mechanical Seals in Centrifugal Pumps*. Available from: https://www.advantageengineering.com/fyi/227/advantageFYI227.php.
9. Ghoneim, A.J.E.c. and management, *Design optimization of photovoltaic powered water pumping systems.* 2006. **47**(11-12): p. 1449-1463.
10. Zhu, Z., et al., *Axial thrust instability analysis and estimation theory of high speed centrifugal pump.* 2022. **34**(7): p. 075118.
11. Li, Y., et al., *Dynamic Characteristic Analysis of Centrifugal Pump Impeller Based on Fluid-Solid Coupling.* 2022. **10**(7): p. 880.
12. Stewart, M., *3 - Centrifugal pumps*, in *Surface Production Operations*, M. Stewart, Editor. 2019, Gulf Professional Publishing: Boston. p. 61-309.
13. Nelik, L., *Centrifugal & rotary pumps: fundamentals with applications*. 1999: CRC press.
14. Weaver, W. and E. Bradner. *Pumps in Industry: Theory and Application*. in *ASME Citrus Engineering Symposium*. 1998. American Society of Mechanical Engineers.
15. Suman, A., C. Kumar, and P.J.A.A. Suman, *Early detection of mechanical malfunctions in vehicles using sound signal processing.* 2022. **188**: p. 108578.
16. *Acoustique - cours*. Available from: https://www.studocu.com/fr/document/institut-des-sciences-et-technologies-de-paris/sciences-de-lingenieur/acoustique-cours/53527141.
17. ERIKS. *Notions d'ingenierie : guide des roulements*. 2023; Available from: https://shop.eriks.lu/fr/guide-roulements/.
18. SKF. *Roulements*. Juillet 2019 Available from: https://www.skf.com/binaries/pub43/Images/0901d1968094ddb3-Rolling-bearings---17000_1-FR_tcm_43-121486.pdf.

19. AYED, A., *Identification des défauts des roulements en régime transitoire par la méthode de la cyclostationnarité.* 2016.
20. NSK, *GUIDE POUR LE REMPLACEMENT DES ROULEMENTS.*

21. *Leventi-Peetz A-M, Östreich T. Deep learning reproducibility and explainable ai (xai). arXiv preprint arXiv:220211452. 2022.*
22. *Jindal M, Juneja N, Garg P. Artificial Intelligence & Automation: Technology Changing the World: Blue Hill Publications.*
23. *Saint-Cirgue G. Apprendre le Machine Learning en une semaine. Machine Learnia, Londres. 2019.*
24. *IBM D. unsupervised-learning 2023 [Available from: https://www.ibm.com/fr-fr/topics/unsupervised-learning.*
25. *Kumar S. Anomaly Detection Algorithms every Data Scientist should know. Dec 13,2021.*
26. *Regents AfAAABABo. Neurons [Available from: https://ib.bioninja.com.au/standard-level/topic-6-human-physiology/65-neurons-and-synapses/neurons.html.*
27. *IBM. recurrent neural networks [Available from: https://www.ibm.com/topics/recurrent-neural-networks.*
28. *Gupta S. AI vs. Machine Learning vs. Deep Learning. June 30, 2022.*
29. *Babu AD. Artificial Intelligence vs Machine Learning vs Deep Learning (AI vs ML vs DL) Nov 5, 2019 [Available from: https://medium.com/@alanb_73111/artificial-intelligence-vs-machine-learning-vs-deep-learning-ai-vs-ml-vs-dl-e6afb7177436.*
30. *Guo P, Huang K, Xu Z. Partial Differential Equations is All You Need for Generating Neural Architectures--A Theory for Physical Artificial Intelligence Systems. arXiv preprint arXiv:210308313. 2021..*
31. *Gurion DB. Activation Functions in Deep Learning Nov 29, 2021 [Available from: https://towardsdatascience.com/what-are-activation-functions-in-deep-learning-cc4f01e1cf5c.*
32. *Pan S, Brunton SL, Kutz JN. Neural Implicit Flow: a mesh-agnostic dimensionality reduction paradigm of spatio-temporal data. Apr; 2022.*
33. *projectpro. Top 10 Deep Learning Algorithms in Machine Learning [2023] 24 Apr 2023 [Available from: https://www.projectpro.io/article/deep-learning-algorithms/443..*
34. *Yamashita R, Nishio M, Do RKG, Togashi K. Convolutional neural networks: an overview and application in radiology. Insights into Imaging. 2018;9(4):611-29.*
35. *Kandpal A. Different Basic Operations in CNN 2023 [Available from: https://iq.opengenus.org/operations-in-cnn/.*

36. *Saha S. A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way December 15, 2018 [Available from: https://saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way/.*
37. *Anil Kumar, Rajesh Kumar, May 12, 2022, "Acoustic and vibration data for defect cases of the centrifugal pump", IEEE Dataport, doi: https://dx.doi.org/10.21227/8cqr-jr43.*
38. *Belaidi N. Réseaux convolutifs (CNN) le 21 juin 2022 [Available from: https://blent.ai/blog/a/cnn-comment-ca-marche.*
39. *Obam YS. Comprendre les Réseaux de Neurones Convolutifs (CNN) Dec 21, 2019 [Available from: https://yannicksergeobam.medium.com/comprendre-les-r%C3%A9seaux-de-neurones-convolutifs-cnn-d5f14d963714.*
40. *Brownlee J. How to choose an activation function for deep learning. Machine Learning Mastery. 2021;181.*
41. *Qayyum R. Introduction To Pooling Layers In CNN. Aug 12, 2022.*
42. *Biswal A. Convolutional Neural Network Tutorial. Apr 24, 2023;13.*
43. *Arc. Convolutional Neural Network Dec 25, 2018 [Available from: https://towardsdatascience.com/convolutional-neural-network-17fb77e76c05.*
44. *Brownlee J. A Gentle Introduction to Batch Normalization for Deep Neural Networks January 16, 2019 [Available from: https://machinelearningmastery.com/batch-normalization-for-training-of-deep-neural-networks/.*
45. *Brownlee J. A Gentle Introduction to Dropout for Regularizing Deep Neural Networks August 6, 2019 [Available from: https://machinelearningmastery.com/dropout-for-regularizing-deep-neural-networks/.*
46. *Habib G, Qureshi S. Optimization and acceleration of convolutional neural networks: A survey. Journal of King Saud University - Computer and Information Sciences. 2022;34(7):4244-68.*
47. *Chen F, Tsou JY. Assessing the effects of convolutional neural network architectural factors on model performance for remote sensing image classification: An in-depth investigation. International Journal of Applied Earth Observation and Geoinformation. 2022;112:102865.*
48. *Al-Quayed F, Soudani A, Al-Ahmadi S. Lightweight feature extraction method for efficient acoustic-based animal recognition in wireless acoustic sensor networks. EURASIP Journal on Wireless Communications and Networking. 2020;2020(1):256.*
49. *Sharma G, Umapathy K, Krishnan S. Trends in audio signal feature extraction methods. Applied Acoustics. 2020;158:107020.*
50. *Antoni J. Fast computation of the kurtogram for the detection of transient faults. Mechanical Systems and Signal Processing. 2007;21(1):108-24.*
51. *Green G. Seeing sound: What is a spectrogram. Audio Project Cataloguer for Unlocking our Sound Heritage. 19 September 2018.*

52. Dwivedi D, Ganguly A, Haragopal VV. 6 - Contrast between simple and complex classification algorithms. In: Goswami T, Sinha GR, editors. Statistical Modeling in Machine Learning: Academic Press; 2023. p. 93-110
53. *Abdul ZK, Al-Talabani AK. Mel Frequency Cepstral Coefficient and its Applications: A Review. IEEE Access. 2022;10:122136-58.*
54. *McFee B. Framing 2020-2022 [Available from: https://brianmcfee.net/dstbook-site/content/ch09-stft/Framing.html#framing.*
55. *Sahidullah M, Saha G. A Novel Windowing Technique for Efficient Computation of MFCC for Speaker Recognition. IEEE Signal Processing Letters. 2012;20.*
56. *Harris FJ. Chapter 3 - Multirate FIR Filters for Interpolating and Desampling. In: Elliott DF, editor. Handbook of Digital Signal Processing. San Diego: Academic Press; 1987. p. 173-287.*
57. *Ali S, Tanweer S, Khalid S, Rao N. Mel Frequency Cepstral Coefficient: A Review2021.*
58. *Fayek H. Speech Processing for Machine Learning: Filter banks, Mel-Frequency Cepstral Coefficients (MFCCs) and What's In-Between Apr 21, 2016 [Available from: https://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html.*
59. *L. M. What is the purpose of the log when computing the MFCC Jun 2, 2013 [Available from: https://dsp.stackexchange.com/questions/8866/what-is-the-purpose-of-the-log-when-computing-the-mfcc.*
60. *MEL-FREQUENCY CEPSTRUM [Available from: https://isip.piconepress.com/courses/msstate/ece_8463/lectures/current/lecture_13/lecture_13_08.html.*
61. *schnarf. How do I interpret the DCT step in the MFCC extraction process. Oct 5, 2011.*
62. *Krey S. Reproducing the feature outputs of common programs using Matlab and melfcc.m 2013/02/26 [Available from: https://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/mfccs.html.*
63. *Anil Kumar RK. Acoustic and vibration data for defect cases of the centrifugal pump 2022.*
64. *GeeksforGeeks. Python Tutorial 24 Apr, 2023 [Available from: https://www.geeksforgeeks.org/python-programming-language/.*
65. *Wolfe M. TensorFlow vs Keras: A Comparison Jul 16, 2021 [Available from: https://towardsdatascience.com/tensorflow-vs-keras-d51f2d68fdfc.*
66. *McFee B, Raffel C, Liang D, Ellis D, McVicar M, Battenberg E, et al. librosa: Audio and Music Signal Analysis in Python2015. 18-24 p.*
67. *José Naranjo-Torres . MM, Ruber Hernández-García ,Ricardo J. Barrientos ,Claudio Fredes and Andres Valenzuela A Review of Convolutional Neural Network Applied to Fruit Image Processing 2020 [Available from: https://www.mdpi.com/2076-3417/10/10/3443.*

68. *Raspberry-Pi-Model-Bplus-Product-Brief. Available from: https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf.*

69. *TensorFlow Lite for Microcontrollers. Available from: https://www.tensorflow.org/lite/microcontrollers?hl=fr.*

70. *librosa.feature.mfcc. Available from: https://librosa.org/doc/latest/generated/librosa.feature.mfcc.html#librosa.feature.mfcc.*

71. *garrec, l. (2023). "Les différents types de pompes industrielles." from https://www.legarrec.com/entreprise/differentes-pompes-industrielles/.*

# Annexes

# Annexe A

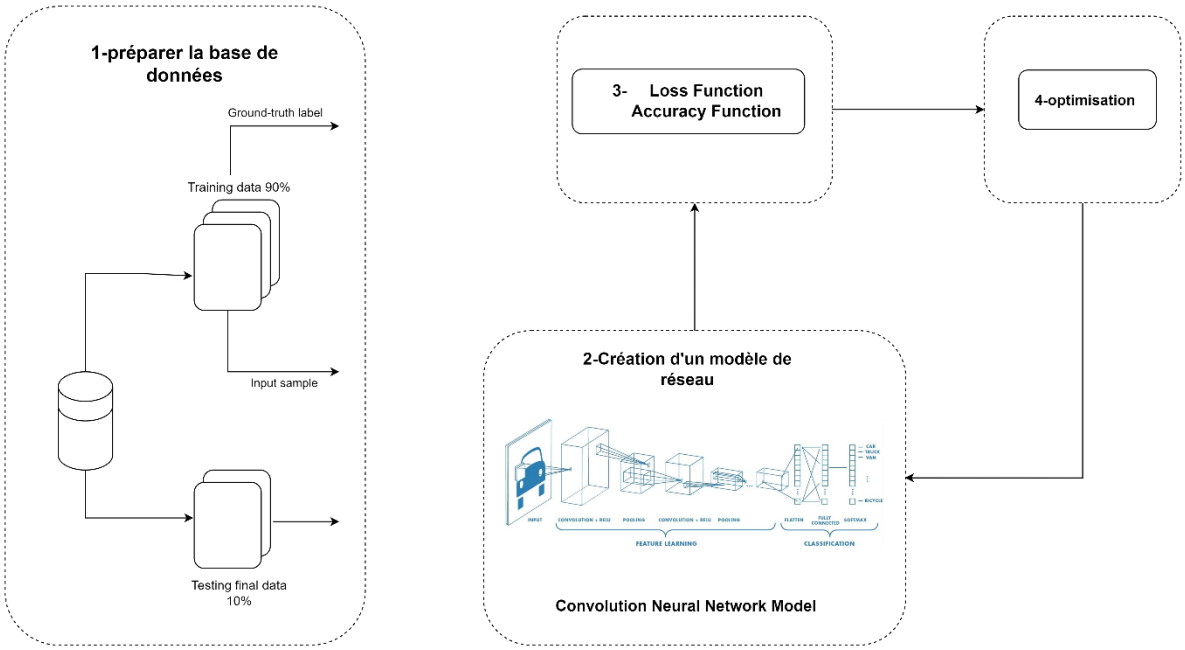



**Figure i : Diagramme schématique de procédure d'entrainement des réseaux CNN**

# Annexe B

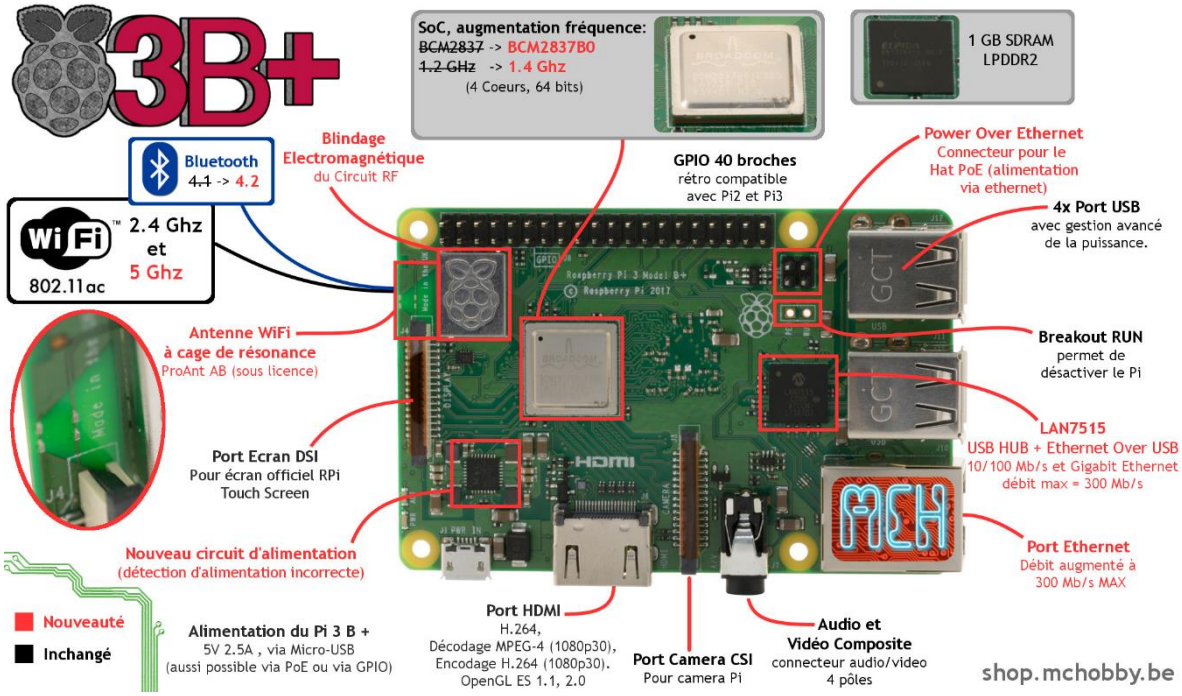



**Figure ii : interfaces d'entrées/sorties de la carte Raspberry pi 3 b+**

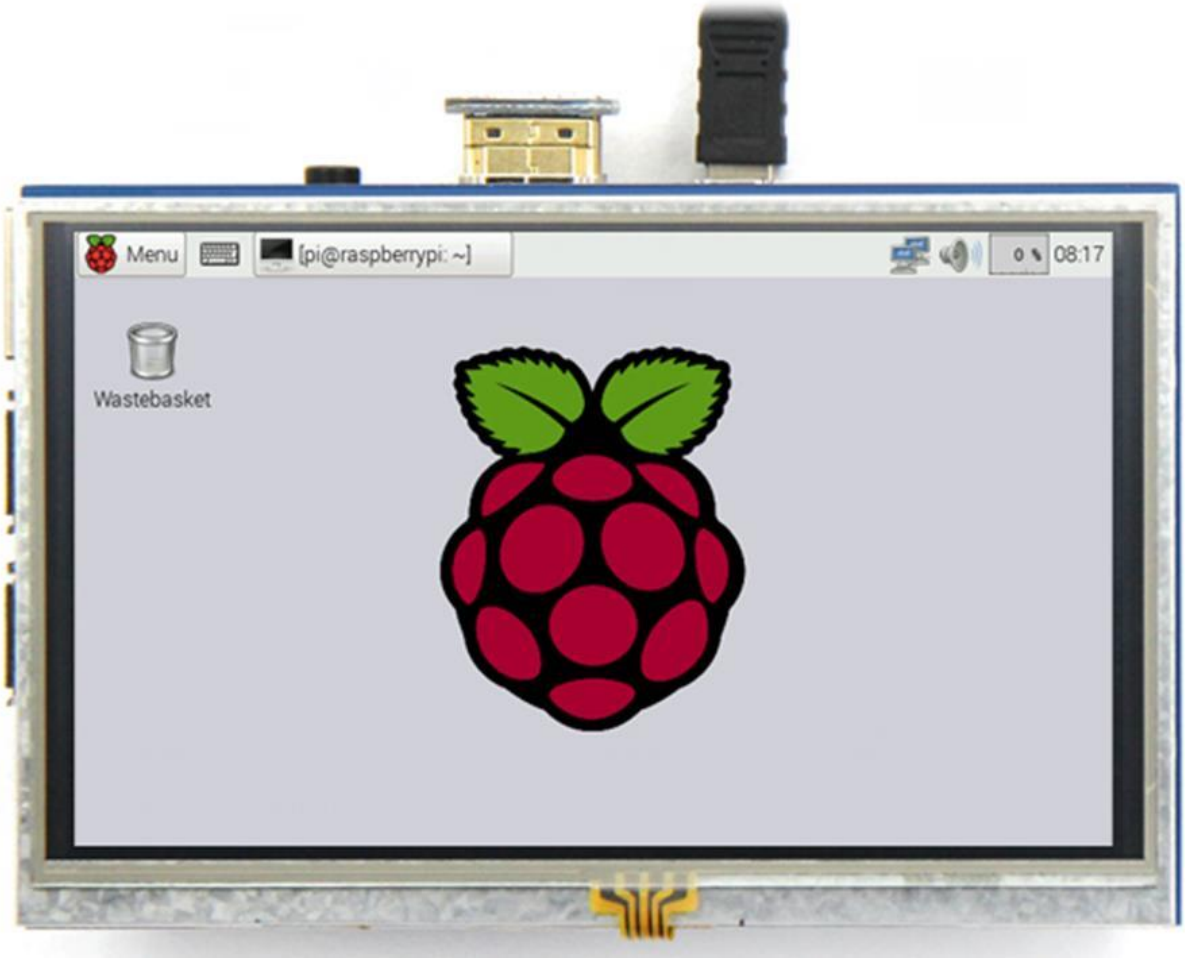
# Annexe C

**Tableau 1 : Tableau des Caractéristiques de Raspberry pi 3B+**

| | |
|---|---|
| Processeur : | Le Raspberry Pi 3B+ est équipé d'un processeur quad-core ARM Cortex-A53 cadencé à 1,4 GHz. Cela lui permet d'exécuter des tâches de calcul de manière efficace. |
| Mémoire : | Il dispose de 1 Go de mémoire vive (RAM) intégrée, ce qui lui permet d'exécuter différentes applications et tâches simultanément. |
| Connectivité : | Le Raspberry Pi 3B+ prend en charge la connectivité sans fil Wi-Fi 802.11ac et le Bluetooth 4.2. Cela permet de connecter l'appareil à des réseaux sans fil et à d'autres périphériques compatibles. |
| Ports et interfaces : | Il dispose de plusieurs ports et interfaces, notamment :<br>- Quatre ports USB 2.0 pour connecter des périphériques externes tels que des claviers, des souris ou des disques durs.<br>- Un port Ethernet pour une connexion réseau filaire.<br>- Un port HDMI pour la sortie vidéo.<br>- Un lecteur de carte microSD pour le stockage de données.<br>- Un port audio de 3,5 mm pour la sortie audio.<br>- Un port GPIO (General Purpose Input/Output) pour la connexion de composants électroniques. |
| Graphiques : | Le Raspberry Pi 3B+ est équipé d'un processeur graphique VideoCore IV 3D, ce qui lui permet d'afficher des graphiques et des vidéos de manière fluide. |
| Système d'exploitation : | Il peut exécuter divers systèmes d'exploitation, y compris des distributions Linux comme Raspbian (basé sur Debian) et d'autres systèmes d'exploitation spécialisés. |
| Alimentation : | Le Raspberry Pi 3B+ est alimenté par un adaptateur secteur standard de 5 volts via un connecteur micro USB. |

# Annexe D

**Écran HDMI 5 pouces**

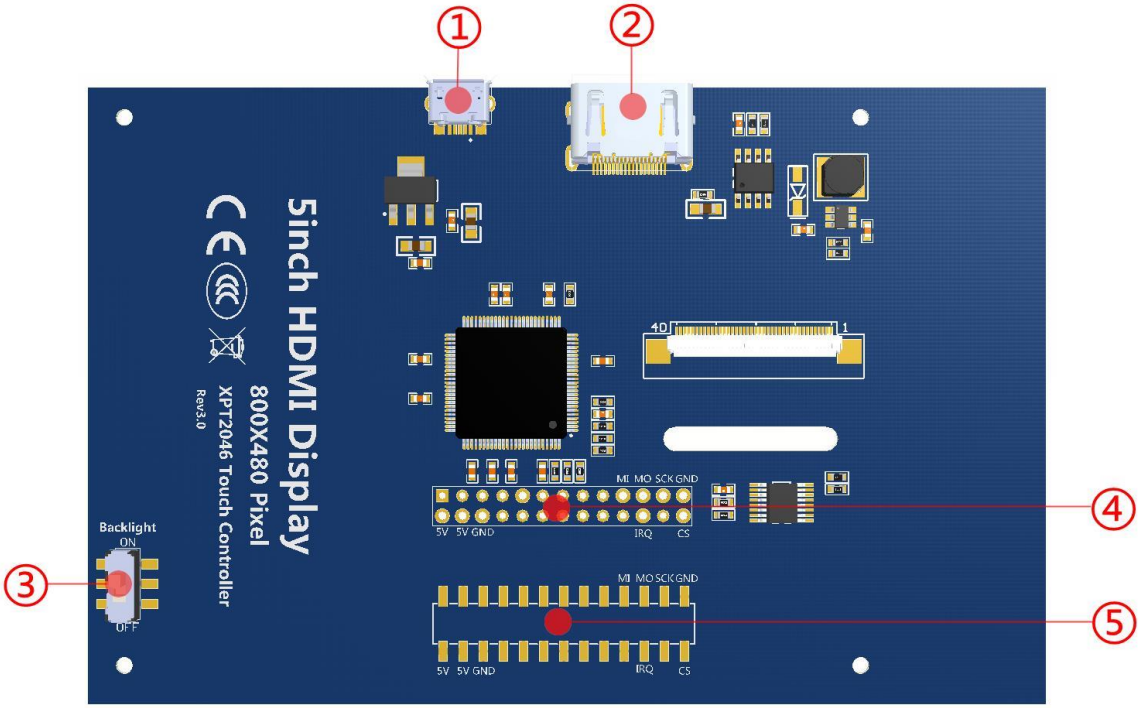



**Description du produit**

- Écran standard 5’’, résolution 800 × 480
- Avec écran tactile résistif, prise en charge du contrôle tactile
- Prend en charge le contrôle du rétroéclairage seul, le rétroéclairage peut être désactivé pour économiser de l'énergie
- Prend en charge l'entrée d'interface HDMI standard, compatible avec et peut être directement inséré avec Raspberry Pi (3ème, 2ème et 1ère génération)
- Peut être utilisé comme moniteur HDMI à usage général, par exemple : se connecter à un ordinateur HDMI comme le sous-affichage (la résolution doit pouvoir forcer la sortie pour 800 x 480)
- Utilisé comme écran de tarte aux framboises qui prend en charge Raspbian, Ubuntu, Kodi, win10 IOT (tactile résistif)

- Fonctionne comme un moniteur PC, prend en charge les systèmes XP, win7, win8, win10 (ne prend pas en charge le toucher)
- Certification CE, RoHS

**Description du matériel**




① **Interface USB :** obtenez une alimentation 5 V à partir de l'USB, si la prise ④ 13 x 2 broches a été connectée, cela l'interface USB peut être No Connecté.

② **Interface HDMI :** Pour la transmission HDMI.

③ **Interrupteur d'alimentation du rétroéclairage :** contrôle l'activation et la désactivation du rétroéclairage pour économiser de l'énergie.

④ **Prise 13 x 2 broches :** obtenez une alimentation 5 V de Raspberry Pi à LCD, en même temps, transférez le signal tactile vers Raspberry Pi.

⑤ **interface étendue :** étendu Le signal de prise ④-13*2 Pin-to-Pin.

**Carte des broches**

④ Prise 13 x 2 broches: obtenez une alimentation 5 V de Raspberry Pi à LCD, en même temps, transférez le signal tactile vers Raspberry Pi

**Tableau 2: branchement avec raspberry pi**

| Description | Pin | NO | NO | Pin | Description |
|---|---|---|---|---|---|
| Power input(5V) | 5V | 2 | 1 | 3.3V | NC |
| Power input(5V) | 5v | 4 | 3 | SDA | NC |
| Power GND | GND | 6 | 5 | SCL | NC |
| NC | TX | 8 | 7 | P7 | NC |
| NC | RX | 10 | 9 | GND | Power GND |
| NC | P1 | 12 | 11 | P0 | NC |
| Power GND | GND | 14 | 13 | P2 | NC |
| NC | P4 | 16 | 15 | P3 | NC |
| NC | P5 | 18 | 17 | 3.3V | NC |
| Power GND | GND | 20 | 19 | MI | TP SPI Bus input (MOSI) |
| TP Interrupt | IRQ | 22 | 21 | MO | TP SPI Bus output (MISO) |
| NC | CEO | 24 | 23 | SCK | TP SPI Bus Clock (SCLK) |
| TP Chip Sellect | TCS | 26 | 25 | GND | Power GND |

# Annexe E

**Martiale utiliser dons la base de donnes**

- **Dispositif d'acquisition de données**

Le périphérique d'acquisition de données utilisé pour acquérir des données acoustiques et vibratoires est le NI-USB-4431 (illustré à la Fig. 4), qui est un périphérique d'acquisition de données portable. Il s'agit d'un périphérique d'E/S analogique 24 bits à 4 canaux utilisé pour acquérir les données de vibration. Le NI-USB-4431 possède des modules de conditionnement de signal avec un filtre de Bessel passe-bas pour la réduction du repliement. L'appareil dispose de 4 canaux d'entrée (ai0, ai1, ai2 et ai3) qui peuvent acquérir une tension dans la plage de ± 10 V. Il dispose d'un canal de sortie analogique (± 3,5 V) et a un taux de mise à jour d'échantillon de 96 kS/s. L'appareil a une résolution de 24 bits et dispose d'un conditionnement de signal IEPE sélectionnable par logiciel (0 ou 2,1 mA). Le signal a été acquis à une fréquence d'échantillonnage de 70 kHz.




**Figure iii : Une photographie typique de DAQ (NI-USB-4431)**

- **Capteur acoustique**

Les données acoustiques ont été enregistrées à l'aide d'un microphone (ECM8000). Le capteur d'émission acoustique utilisé pour acquérir les données est représenté sur la figure 5 et a été connecté à tous les canaux du dispositif d'acquisition de données



**Figure iv : Microphone utilisé pour l'acquisition de données**