



Université ABBES LAGHROUR Khenchela
Faculté des Sciences et de la Technologie
Département de Génie Industriel
جامعة عباس لغرور خنشلة
كلية العلوم والتكنولوجيا
قسم الهندسة الصناعية



N° Série :

Mémoire de fin d'étude

Pour l'obtention du diplôme de Master

Filière : Télécommunications

Spécialité : Systèmes des Télécommunications

THEME

**Analyse des signaux de navigation par
satellites en utilisant les réseaux de
neurones convolutionnels (CNN)**

Réalisé par : TAFERGHOUST Amal

KETTAR Fayrouz

Soutenu le : 19/06/2023 *Devant le jury composé de:*

Dr. KHEZZAR Zaki Aissam

Président

Université Abbes Laghrour-Khenchela

Dr. DOUAK Fouzi

Encadreur

Université Abbes Laghrour-Khenchela

Dr. BOURAS Moustafa

Examineur

Université Abbes Laghrour-Khenchela

Promotion 2022/2023

*Je dédie mon diplôme aux personnes les plus chères
qui ont beaucoup enduré pour moi et ont travaillé
dur pour que j'atteigne ce stade.*

Sans aucun doute, ils constituent ma précieuse famille.

Merci à tous.

*Un grand merci également à mon amie Imane Mahrez
pour son soutien. Que Dieu vous bénisse
et vous souhaite beaucoup de succès.*

Amal

Je dédie ce mémoire :

*À ma mère pour ses sacrifices et à mon père pour sa
confiance.*

*À mes frères, mes amies qui m'ont toujours soutenu et
encouragé durant ces années.*

À toute ma famille.

Fayrouz

Remerciements

Merci à Dieu, qui nous protège et nous aide dans les épreuves de la vie et qui nous donne le courage et l'ardeur nécessaire pour l'accomplissement de ce travail.

*Nous tenons à remercier notre encadreur de mémoire **Dr. Fouzi Douak**, pour ses conseils et suggestions avisés qui m'aider à mener à bien ce travail, et d'avoir rapporté à ce mémoire ses remarques et conseils.*

*Nous adressons nos plus sincères remerciements à Monsieur **Dr. KHEZZAR Zaki Aissam**, pour nous avoir fait l'honneur de présider le jury de notre mémoire. Nous remercions également à Monsieur **Dr BOURAS Moustafa**, qui a accepté d'examiner ce travail.*

On remercie aussi tout les enseignants de la filière systèmes des télécommunications pour la qualité de leurs conseils, leurs disponibilités, leurs patiences et leurs aides tout au long de notre travail.

Enfin, nous exprimons notre remerciement à tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce travail.

Résumé

Les systèmes de navigation par satellite GNSS jouent un rôle crucial dans les services de localisation au quotidien. Cependant, leur avenir est menacé par des interférences non intentionnelles provenant d'autres systèmes de communication, ainsi que par des interférences intentionnelles, connues sous le nom de brouillage, qui visent à tromper les récepteurs. Pour relever ces défis, les chercheurs de divers domaines s'orientent vers des solutions pratiques et simples, adaptées aux avancées technologiques actuelles, plutôt que de s'appuyer sur des méthodes traditionnelles complexes et inefficaces.

Dans notre étude, nous nous concentrons sur le développement d'un mécanisme robuste et précis pour la détection et la suppression des interférences en utilisant des algorithmes d'apprentissage profond basés sur les méthodes de convolution CNN (réseau de neurones convolutionnels). Les résultats obtenus après simulation sont très satisfaisants en termes d'amélioration des performances de détection et de suppression des interférences. Les modèles CNN surpassent la méthode OCSVM (machines à vecteurs de support à une classe) en termes d'erreur (RMSE) et de coefficient de corrélation (R). Cela démontre l'efficacité des modèles CNN dans ce contexte et met en évidence leur supériorité par rapport aux approches traditionnelles.

Mots clés : GNSS, Interférence, Classification, OCSVM, Apprentissage profond, CNN.

Abstract

Satellite navigation systems GNSS play a crucial role in everyday location-based services. However, their future is threatened by unintentional interference from other communication systems, as well as intentional interference, known as jamming, which aims to deceive receivers. To address these challenges, researchers from various fields are turning to practical and simple solutions that align with current technological advancements, rather than relying on complex and ineffective traditional methods.

In our study, we focus on developing a robust and accurate mechanism for the detection and suppression of interference using deep learning algorithms based on convolutional neural network (CNN) methods. The results obtained after simulation are highly satisfactory in terms of improving interference detection and suppression performance. CNN models outperform the one-class support vector machine (OCSVM) method in terms of error (RMSE) and correlation coefficient (R). This demonstrates the effectiveness of CNN models in this context and highlights their superiority over traditional approaches.

Keywords: GNSS, Interference, Classification, OCSVM, Deep learning, CNN.

ملخص

تلعب أنظمة الملاحة بالأقمار الصناعية GNSS دورًا حاسمًا في الخدمات اليومية فيما يتعلق بتحديد المواقع. ومع ذلك، فإن مستقبلها مهدد بالتشويش غير المقصود الناتج عن أنظمة الاتصال الأخرى، بالإضافة إلى التشويش المقصود، المعروف أيضًا باسم الإعاقة، والذي يهدف إلى إضلال المستقبلات. لمواجهة هذه التحديات، يلجأ الباحثون من مجالات مختلفة إلى حلول عملية وبسيطة تتوافق مع التطورات التكنولوجية الحالية، بدلاً من الاعتماد على الطرق التقليدية المعقدة وغير فعالة. في دراستنا، نركز على تطوير آلية قوية ودقيقة للكشف عن التشويش وإزالته باستخدام خوارزميات التعلم العميق المستندة إلى طرق الشبكات العصبية المتراكبة (CNN). تعتبر النتائج التي تم الحصول عليها بعد المحاكاة مرضية للغاية من حيث تحسين أداء الكشف عن التشويش وإزالته. تفوق نماذج CNN على طريقة OCSVM من حيث الخطأ (RMSE) ومعامل الارتباط (R). يبرهن ذلك على فعالية نماذج CNN في هذا السياق ويسلط الضوء على تفوقها على النهج التقليدي.

الكلمات المفتاحية : GNSS, التداخل, التصنيف, OCSVM, التعلم العميق, CNN.

Liste des tableaux

Tableau I. 1. La différence entre l'apprentissage supervisé et non supervisé.	30
Tableau III. 1. Max et Moyenne Pooling.....	49
Tableau III. 2. Configuration réelle de réseau.	60
Tableau IV. 1. Paramètres du signal E5a1.....	68
Tableau IV. 2. Les résultats obtenus par les différentes architectures CNN (28×28, 48×48, et 64×64) sur la base de données de test des signaux GNSS.	86

Liste des figures

Figure I. 1. Principe de Trilatération.	6
Figure I. 2. Segment du système GNSS.	7
Figure I. 3. Bandes de fréquences allouées par les signaux GNSS.	11
Figure I. 4. Chaîne de récepteur GNSS.	12
Figure I. 5. Schéma global d'un récepteur GPS.	13
Figure I. 6. Exemples de résultat d'acquisition.	14
Figure I. 7. Principe d'acquisition par recherche série.	15
Figure I. 8. Acquisition par recherche parallèle sur le retard de code par (FFT).	15
Figure I. 9. Schéma bloc de la recherche parallèle sur l'espace de fréquences par FFT.	16
Figure I. 10. Schéma fonctionnel d'une boucle de poursuite du code et de la porteuse.	17
Figure I. 11. Différents types d'interférences sur les récepteurs GNSS.	18
Figure I. 12. Les brouilleurs.	24
Figure II. 1. Schéma d'un modèle supervisé.	28
Figure II. 2. Schéma d'un modèle non-supervisé.	29
Figure II. 3. Neurone artificiel.	31
Figure II. 4. L'architecture d'un réseau de neurones de type MLP.	32
Figure II. 5. Structure d'un réseau de neurone multicouche.	32
Figure II. 6. Les trois types de fonctions de transfert.	33
Figure II. 7. Architecture d'un réseau de neurones à base de fonctions radiales.	35
Figure II. 8. Transformer l'espace de départ vers un espace de dimension supérieure.	36
Figure II. 9. Séparation de deux ensembles de points par un hyperplan H.	37
Figure II. 10. Cas linéairement séparable.	38
Figure II. 11. Illustration d'un classificateur SVM dans le cadre d'une classification binaire.	38
Figure II. 12. One-class SVM.	39
Figure II. 13. Principe des OCSVM.	40
Figure II. 14. Deux cas de la classification (bi-classe et multi-classe).	42

Figure III. 1. La relation entre l'intelligence artificielle, le ML et le deep learning.....	46
Figure III. 2. Les réseaux de neurones convolutifs (CNN).....	46
Figure III. 3. Exemple d'architecture CNN.	47
Figure III. 4. Exemple de schéma de principe du CNN.....	48
Figure III. 5. Représentation de convolution sur une image de taille «4×4» pixels avec une filtre de convolution.	49
Figure III. 6. Max Pooling avec un filtre 2x2.	50
Figure III. 7. Illustration de la fonction d'activation.....	50
Figure III. 8. Fonction sigmoïde.	51
Figure III. 9. La fonction RELU.	52
Figure III. 10. La fonction Leaky RELU.	53
Figure III. 11. Représentation graphique de la fonctionnalité ReLU.	53
Figure III. 12. Couche entièrement connectée (Fully connected).....	54
Figure III. 13. La fonction Softmax.	55
Figure III. 14. Exemple montrant l'étiquette codée de la couche de sortie CNN.....	55
Figure III. 15. Architecture LeNet-5.....	57
Figure III. 16. Architecture AlexNet.....	58
Figure III. 17. Représentation de l'architecture de AlexNet.....	58
Figure III. 18. Architecture ZF Net.....	58
Figure III. 19. Architecture de VGG-16.	59
Figure III. 20. Représentation 3D de l'architecture de VGG-16.	59
Figure III. 21. Architecture MobileNet-V2.....	62
Figure III. 22. Architecture GoogLeNet/Inception-v1.....	62
Figure III. 23. Architecture de ResNet.....	63
Figure IV. 1. Présentation d'un signal E5 avec les deux composantes E5a et E5b.	68
Figure IV. 2. Amplitude du signal E5aI dans le domaine temporel sans interférence.	69
Figure IV. 3. Amplitude du signal E5aI dans le domaine temporel contaminé par une interférence DMA, ISR varie entre 10dB à 60dB.....	70
Figure IV. 4. Amplitude du signal E5aI dans le domaine fréquentiel sans interférence.....	70
Figure IV. 5. Amplitude du signal E5aI dans le domaine fréquentiel contaminé par des interférences DMA, ISR varie entre 10dB à 60dB.	71

Figure IV. 6. Classification des interférences des signaux GNSS en utilisant l'architecture CNN.....	72
Figure IV. 7. Principe de conversion des signaux en une image au niveau de gris.	73
Figure IV. 8. Structure du modèle CNN.	74
Figure IV. 9. Image en niveau de gris 2D des signaux GNSS sans interférences.....	76
Figure IV. 10. Image en niveau de gris 2D des interférences des signaux GNSS.	76
Figure IV. 11. Schéma bloc de la suppression numérique de l'interférence pulsée.....	77
Figure IV. 12. Architecture CNN de dimension 28×28, (a) Architecture et (b) Détail des couches de convolution sous MATLAB.	79
Figure IV. 13. Évolution des critères Accuracy et Loss en fonction du nombre d'itérations sur la base de données GNSS durant la phase d'apprentissage avec une entrée de dimension 28×28.....	80
Figure IV. 14. Architecture CNN de dimension 48×48, (a) Architecture et (b) Détail des couches de convolution sous MATLAB.	81
Figure IV. 15. Évolution des critères Accuracy et Loss en fonction du nombre d'itérations sur la base de données GNSS durant la phase d'apprentissage avec une entrée de dimension 48×48.....	82
Figure IV. 16. Architecture CNN de dimension 64×64, (a) Architecture et (b) Détail des couches de convolution sous MATLAB.	84
Figure IV. 17. Évolution des critères Accuracy et Loss en fonction du nombre d'itérations sur la base de données GNSS durant la phase d'apprentissage avec une entrée de dimension 64×64.....	85
Figure IV. 18. Evolution de l'architecture CNN 48×48 en terme de RMSE en fonction des différentes valeurs ISR.	87
Figure IV. 19. Evolution de l'architecture CNN 48×48 en terme de R en fonction des différentes valeurs ISR.	88
Figure IV. 20. Application de l'architecture CNN 48×48 pour la détection et la suppression des interférences, (a) ISR=10dB, (b) ISR=20dB, (c) ISR=30dB, (d) ISR=40dB, (e) ISR=50dB, (f) ISR=60dB.....	90

Liste des symboles

AWGN Additive White Gaussian Noise
ADC Analog to Digital Converter
ARNS Aeronautical Radio Navigation Service
AltBOC Alternative BOC modulation
CS Commercial Service
CWI Continuous Wave Interference
C/A Coarse / Acquisition
CNN Convolutional Neural Networks
CDMA Code Division Multiple Access
DME Equipement de Mesure de Distance
DVB-T Digital Video Broadcasting–Terrestrial
DVB-S Digital Video Broadcasting–Satellite
DME Interférence Pulsé
DLL Delay Lock Loop
DL Deep Learning
DoD Direct outward Dialing
FFT Fast Fourier Transform
FLL Frequency Lock Loop
FM Modulation de Fréquence
GNSS Global Navigation Satellite System
GPS Global Positioning System
GLONASS GLObal'naya NAVigatsionnaya Sputnikovaya Sistema
JTIDS Joint Tactical Information Distribution System
MCWI Interférence à Onde Continue Multiple
ML Machine Learning
MLP Multi-Layer Perceptron
MOE Medium Orbit Earth
MIDS Système de Distribution d'Informations Multifonction
NAVSTAR Navigation Satellite Timing And Ranging

NBI Narrow Band Interference
OCSVM One Class Support Vector Machine
PLL Phase Lock Loop
PPS Precise Positioning Service
PVT Position, Velocity and Timing
PRN Pseudo Random Noise
PED Personal Electronic Devices
RFI Radio France Internationale
RMSE Root Mean Square Error
R Corrélation
RBF Radial Basis Function
SOL Safety Of Life
SSR Surveillance Secondaire Radar
SAR Search And Rescue
SPS Standard Positioning Service
SATCOM Communications par Satellite
SVM Support Vector Machine
TACAN TACTical Air Navigation
UHF Ultra High Frequency
UWB Ultra-Wide Band
VHF Very High Frequency
VHFCOM Communication en VHF
WBI Wide Band Interference

Table des matières

Introduction Générale	1
Chapitre I	
Signaux GNSS	
I.1. Introduction	5
I.2. Principes fondamentaux du GNSS	5
I.2.1. Position, vitesse et temps	6
I.3. Segments des systèmes de navigation par satellites.....	7
I.3.1. Segment spatial	8
I.3.2. Segment de contrôle.....	8
I.3.3. Segment utilisateur	8
I.4. Systèmes mondiaux de navigation par satellite.....	9
I.4.1. Système de positionnement global GPS	9
I.4.2. Système de navigation Russe GLONASS	10
I.4.3. Système de navigation Galileo	10
I.4.4. Système de navigation Beidou.....	11
I.5. Signaux GNSS	11
I.5.1. Principe de navigation par satellites	12
I.5.2. Chaîne de réception GNSS	12
I.5.3. Architecture d'un récepteur GPS.....	13
I.5.4. Acquisition du signal GNSS	14
I.5.4.1. Acquisition par recherche en série	14
I.5.4.2. Acquisition par recherche parallèle sur le retard de code par (FFT).....	15
I.5.4.3. Acquisition par recherche parallèle sur le décalage de fréquences par (FFT)	16
I.5.5. Poursuite du signal GNSS	16
I.6. Classification de la source interférente	17
I.6.1. Caractéristiques spectrales d'interférence	18
I.6.1.1. Les interférences pulsées.....	20

1.7. Sources potentielles d'interférence.....	21
<u>1.7.1. Signaux hors bande</u>	21
<u>1.7.2. Signaux en bande</u>	22
I.8. Brouilleurs.....	23
I.9. Impact des Interférences radiofréquence sur les récepteurs GNSS.....	24
I.10. Conclusion.....	24

Chapitre II

Apprentissage automatique

II.1. Introduction.....	27
II.2. Algorithmes d'apprentissage	28
<u>II.2.1. Apprentissage supervisé</u>	28
<u>II.2.2. Apprentissage non supervisé</u>	29
<u>II.2.3. Différence entre apprentissage supervisé et non supervisé</u>	30
II.3. Méthodes de classification traditionnelle	30
<u>II.3.1. Réseaux de neurones multicouches MLP.....</u>	30
<u>II.3.1.1. Fonctions de transfert.....</u>	33
<u>II.3.1.2. Mise en œuvre des réseaux neuronaux</u>	34
<u>II.3.2. Réseaux de neurones à fonction radiales de base RBF</u>	35
<u>II.3.3. Machines à Vecteurs de Support (SVM).....</u>	36
<u>II.3.3.1. Principe des SVMs.....</u>	37
<u>II.3.4. Machine à vecteur de support à une classe OCSVM</u>	39
<u>II.3.4.1. Le principe d'OCSVM.....</u>	40
<u>II.3.5. Comparaison SVM avec OCSVM</u>	41
II.4. Optimisation des paramètres des algorithmes d'apprentissage automatique.....	41
<u>II.4.1. Validation simple.....</u>	41
<u>II.4.2. Classification binaire (Bi-classe).....</u>	41
II.5. Conclusion	42

Chapitre III

Apprentissage profond

III.1. Introduction	45
III.2. Définition de l'apprentissage profond (deep learning).....	45

III.3. Architectures de réseaux de neurones profonds	47
III.3.1. Principe d'architecture d'un réseau de neurones convolutionnel CNN	47
III.3.2. Blocs de construction	48
III.3.3. Couche de convolution (CONV).....	48
III.3.4. Couche de pooling (POOL)	49
III.3.5. Fonctions d'activation.....	50
III.3.6. Couche entièrement connectée (FC)	53
III.3.7. Couche de perte (LOSS)	54
III.3.8. Couche de sortie (output layer).....	55
III.3.9. Choix des hyperparamètres	55
III.3.10. Avantages de CNNs	56
III.4. Classification de l'apprentissage profond.....	56
III.4.1. LeNet-5	56
III.4.2. AlexNet	57
III.4.3. ZFNet	58
III.4.4. VGGNet	59
III.5. Domaines d'application de l'apprentissage profond	64
III.6. Conclusion	64

Chapitre IV

Résultats et discussion

IV.1. Introduction	67
IV.2. Description de la base de données GNSS	67
IV.3. Méthode proposée	72
IV.3.1. Conversion des signaux 1D en images 2D.....	72
VI.3.2. Réseaux de neurones convolutifs	73
VI.3.3. Génération des images aux niveaux de gris	75
IV.4. Résultats des différents modèles CNN.....	76
VI.4.1. Conception du modèle neuronal CNN 28×28.....	78
VI.4.2. Conception du modèle neuronal CNN 48×48.....	80
VI.4.3. Conception du modèle neuronal CNN 64×64.....	83
VI.4.4. Comparaison entre les différents modèles CNN.....	85

IV.5. Conclusion.....	91
Conclusion générale	93
<i>Bibliographie</i>	96

Introduction générale

Introduction générale

La navigation par satellite est une technologie largement utilisée et indispensable dans notre vie quotidienne. Les systèmes qui offrent ce service sont connus sous le nom de GNSS systèmes globaux de navigation par satellite. Il s'agit de la radionavigation basée sur des satellites qui proposent aux usagers civils des services de géolocalisations [1].

Aujourd'hui, le GNSS comprend deux principaux groupes de satellites : (1) Le GPS (Global Positioning System) des Etats-Unis, et (2) GLONASS (Globalnaya Navigatsionnaya Sputnikovaya Sistema) de la Fédération de Russie. Deux autres constellations majeures sont en cours de déploiement : Galileo pour l'Europe et les systèmes Compass/Beidou de la Chine [2]. Ces systèmes GNSS ont été conçus essentiellement pour le positionnement d'un utilisateur au terrain équipé d'un récepteur radio fréquence [3], le récepteur doit maîtriser et traiter les signaux émis par les satellites de la constellation afin d'estimer le triplet : position, vitesse et temps (PVT) à tout endroit de la terre [4].

Le positionnement par satellite possède beaucoup d'avantages par rapport aux instruments longtemps utilisés et offre aujourd'hui une précision inégalée par rapport aux autres systèmes. Néanmoins, il se heurte à des problèmes liés à la propagation dans l'atmosphère, à l'instabilité des horloges utilisées, aux obstacles dans le milieu de réception ou encore au bruit électronique du récepteur. Les erreurs occasionnées par ces phénomènes conduisent parfois à des erreurs de précision pouvant atteindre des dizaines de mètres [5].

Les constellations opérationnelles nominales GPS, GALILEO, GLONASS, et BEIDOU, se composent de plusieurs dizaines de satellites évoluant entre 20000km et 30000km d'altitude suivant des orbites équitablement réparties pour couvrir tous les continents [6]. Les signaux reçus sur terre sont de faible puissance, inférieure à celle du bruit thermique, ce qui rend les récepteurs GNSS très vulnérables aux interférences. Ainsi, le principal problème dans le domaine des signaux GNSS est la détection et la suppression des interférences.

Pour résoudre le problème des interférences, peu de méthodes de détection et de réduire les interférences au niveau des récepteurs GNSS ont été proposées dans la littérature scientifique. On peut citer l'article [7], dans leur étude ils ont proposé une méthode composée d'une série d'opérations. Tout d'abord, le signal reçu, qui est immergé dans un bruit gaussien blanc additif (AWGN), est multiplié, dans le domaine temporel, par une fenêtre Tukey. Ensuite, il est transformé dans le domaine DCT (Transformée en Cosinus Discrète). Afin de détecter et de localiser les composantes fréquentielles de l'interférence, le signal transformé est partitionné en paquets non superposés. Le seuillage d'un paquet est obtenu grâce à

l'utilisation du seuil universel de l'algorithme de DONOHO. La variance est estimée sur la base de la théorie de l'échantillonnage statistique. La dernière étape consiste à appliquer la DCT inverse pour obtenir une bonne estimation du bruit gaussien.

Dans [8], l'auteur propose une approche de détection et de classification basée sur les réseaux de neurones convolutifs (CNN). L'entrée du réseau est la représentation temps-fréquence visuelle du signal reçu, avec des caractéristiques statistiques dans les domaines temporel et fréquentiel. La représentation fréquentielle-temporelle est obtenue à l'aide de la transformée de Fourier à court terme. Dans ce travail, l'auteur utilise des modèles pré-entraînés tels que AlexNet et ResNet.

Les techniques de traitement numérique du signal ont longtemps été utilisées comme des méthodes traditionnelles de détection des interférences. Cependant, elles sont souvent basées sur des opérations complexes et coûteuses en calcul. Des recherches récentes ont montré que les méthodes d'apprentissage automatique peuvent être utilisées pour détecter les interférences. Cependant, *la question se pose de savoir quelle technique est la plus appropriée pour détecter et supprimer les interférences dans les récepteurs GNSS ?*

Au cours des dernières années, l'apprentissage profond (Deep Learning - DL) a été considéré comme l'étalon-or dans la communauté de l'apprentissage automatique (Machine Learning - ML). Il est devenu la méthode computationnelle la plus utilisée dans le domaine du ML, offrant des performances exceptionnelles sur de nombreuses tâches cognitives complexes, parfois même surpassant les performances humaines. L'un des avantages clés du DL est sa capacité à apprendre à partir de grandes quantités de données. Ce domaine s'est rapidement développé ces dernières années et a été largement utilisé avec succès dans de nombreuses applications traditionnelles [9].

Le réseau de neurones convolutionnel (CNN) est en effet l'un des réseaux de deep learning (DL) les plus largement utilisés et populaires. Grâce au CNN, le DL est devenu très populaire de nos jours. L'avantage principal du CNN par rapport à ses prédécesseurs est sa capacité à détecter automatiquement les caractéristiques importantes sans supervision humaine, ce qui le rend extrêmement utile [9]. Nous avons donc approfondi le CNN en introduisant ses principaux composants. De plus, nous avons détaillé les structures les plus courantes de CNN, en commençant par LeNet-5 et en terminant par le réseau de neurones résiduels (ResNet).

Dans ce mémoire, nous avons utilisé le modèle CNN pour entraîner des caractéristiques abstraites des signaux GNSS dans le domaine temporel. Nous avons appliqué une méthode de prétraitement des données permettant de convertir directement les signaux GNSS 1D en

images 2D. Cette approche a été utilisée pour détecter les interférences dans les récepteurs GNSS. Pour trouver une architecture adaptée à la détection d'interférences, nous avons proposé l'utilisation de trois types d'architectures CNN converties en images 2D en niveaux de gris de différentes tailles, à savoir 28x28, 48x48 et 64x64. Nous avons également intégré plusieurs types de couches de régularisation (BatchNormalization et Dropout) afin d'améliorer ces architectures. Par la suite, afin de prouver l'efficacité de notre méthode, nous avons comparé les différentes architectures avec la méthode de machine à vecteur de support à une classe (OCSVM), qui est une méthode robuste contre les valeurs aberrantes.

Ce mémoire est organisé de la manière suivante :

Dans le premier chapitre, nous présentons les systèmes mondiaux de navigation par satellites de manière générale, en abordant la définition des différents systèmes GNSS, leur principe de fonctionnement, notamment le GPS.

Le deuxième chapitre est consacré à l'introduction des techniques d'apprentissage automatique, plus précisément les méthodes de classification telles que les réseaux de neurones multicouches (MLP), les fonctions radiales de base RBF, les machines à vecteurs de support (SVM) et SVM à une classe (OCSVM).

Le troisième chapitre traite des concepts liés au deep learning, en présentant les différentes structures, les avantages et les domaines d'application associés.

Dans le dernier chapitre, nous examinons les différentes méthodes de classification, en mettant l'accent sur le classificateur CNN pour la détection et l'élimination des interférences. Enfin, nous comparons la méthode du classificateur CNN avec la méthode du classificateur OCSVM.

En conclusion, ce travail modeste se termine par une synthèse générale incluant certaines perspectives pour de futures recherches.

Chapitre I

Signaux GNSS

I.1. Introduction

Les systèmes de navigation par satellite sont devenus indispensables dans notre vie quotidienne, offrant des fonctionnalités de positionnement et de navigation précises et fiables. Le système GNSS (Global Navigation Satellite System) est un réseau mondial de satellites qui permet aux utilisateurs de déterminer leur position à tout moment aussi précisément que possible. L'un des aspects clés du système GNSS est la transmission des signaux, qui permettent aux récepteurs GNSS de recevoir les informations nécessaires pour déterminer la position de l'utilisateur en utilisant un processus appelé Trilatération. Les signaux GNSS sont également utilisés pour synchroniser les horloges des récepteurs et pour obtenir des informations sur l'état des satellites et du système de navigation [10].

Les signaux GNSS sont généralement transmis dans les bandes de fréquences radio et utilisent des techniques de modulation pour transporter les informations nécessaires pour déterminer la position et l'heure. Les signaux GNSS sont également soumis à diverses perturbations lors de leur propagation dans l'atmosphère terrestre et à différentes interférences avec d'autres systèmes de communication, ce qui peut affecter leur qualité et leur précision. Pour cela, notre objectif dans ce travail est la détection et la suppression des différents types d'interférences des récepteurs GNSS.

Dans ce chapitre, nous explorerons en détail les principes fondamentaux des signaux GNSS, notamment la détermination des positions par rapport à trois satellites et les différents segments de navigation. Nous aborderons également les systèmes mondiaux de navigation, notamment les systèmes déjà opérationnels GPS (Global Positioning System), Glonass (GLObalnaya NAVigatsionnaya Sputnikovaya Sistema), ainsi que les systèmes Galileo et BEIDOU, qui sont encore en phase de développement. Ensuite, nous détaillerons les systèmes GNSS, tels que le principe de navigation par satellites, la chaîne de réception, leur architecture, etc. Enfin, nous aborderons les problèmes majeurs des signaux GNSS, tels que les interférences.

I.2. Principes fondamentaux du GNSS

Les bases du GNSS depuis que l'humanité a commencé à explorer le monde, la localisation a toujours joué un rôle clé dans son succès. Initialement, la navigation se faisait avec des cartes utilisant des points de référence tels que des montagnes, des ruisseaux et des rivières. Le manque de points de référence est considéré comme un problème sérieux depuis le début de la navigation maritime. En conséquence, les navigateurs en sont venus à se fier aux observations du ciel pour se localiser. L'avènement de la science du chronométrage a amélioré la précision

des lectures du ciel et, à ce jour, le chronométrage joue l'un des rôles les plus importants dans les systèmes de navigation.

Il est important d'obtenir la position du téléphone mobile en mouvement et d'adapter les mesures à la vitesse du véhicule. La navigation moderne ne détermine plus sa position en observant les étoiles, mais aussi en collectant des signaux radio émis par des satellites à des endroits connus. La position est ensuite restituée en réalisant un Triplicat des différentes distances obtenues par traitement du signal reçu.

I.2.1. Position, vitesse et temps

L'objectif de tout système de navigation GNSS est de permettre à l'utilisateur de calculer sa solution de la position, la vitesse et le temps (PVT). Pour obtenir ces paramètres, nous devons d'abord définir un système de référence pour la position et le temps. Les GNSS utilisent le concept de temps d'arrivée (TOA) pour récupérer la solution. Ce concept mesure l'intervalle de temps entre la transmission du signal du satellite et le moment où le signal est reçu par le récepteur. Grâce à ce temps de propagation, le récepteur est capable de calculer les distances entre le récepteur et les satellites. Pour un système tridimensionnel, nous créons une sphère qui entoure chaque satellite avec un rayon égal à la distance obtenue et en observant l'intersection de trois de ces sphères, donc nous pouvons obtenir notre position. En supposant que le satellite envoie le signal à l'instant t_0 et qu'il est reçu à l'instant $t_0 + \tau$, le rayon de la sphère peut être exprimé comme montré dans la Figure I.1, où c est la vitesse de la lumière [11].

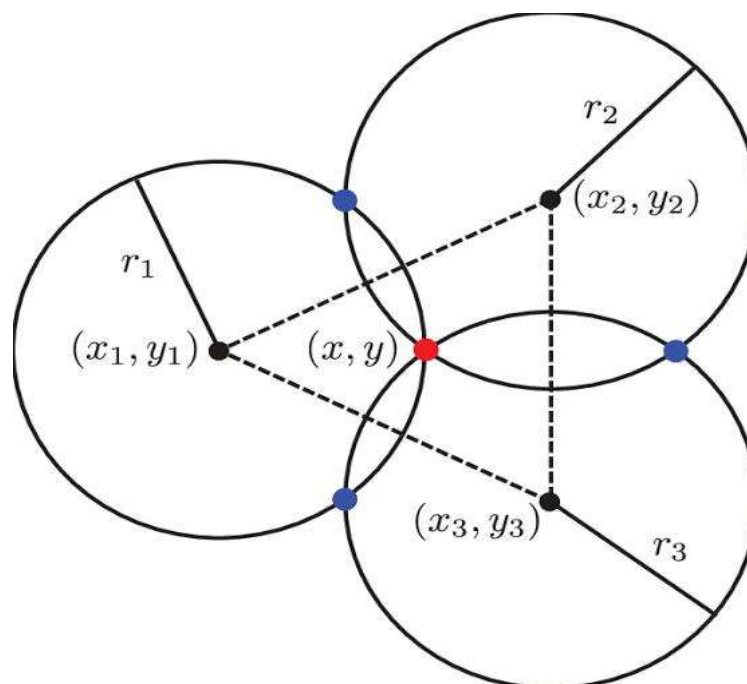


Figure I. 1. Principe de Trilatération.

$$P_i = c \tau_i \quad (\text{I.1})$$

$$P_i = \sqrt{(x_i - x_u)^2 + (y_i - y_u)^2 + (z_i - z_u)^2} + b_u \quad (\text{I.2})$$

À partir de la Figure I.1, nous pouvons observer un exemple de trilatération avec trois satellites différents. Si nous supposons que l'horloge du récepteur est parfaitement synchronisée avec l'heure système de navigation, le calcul TOA devient simple. Malheureusement, ce n'est pas le cas pour les récepteurs GNSS. Ensuite, les signaux obtenus du satellite auront un biais b_u , en raison de la différence entre l'horloge du système et l'horloge du récepteur. Les mesures effectuées par le récepteur sont appelées pseudo-distances. En générale, une pseudo-distance P_i est calculé en utilisant la formule (I.2), où x_j, y_j, z_j sont les coordonnées connues du i^{eme} satellite, x_u, y_u, z_u sont les coordonnées de l'utilisateur et b_u est le biais lié à l'erreur de synchronisation. Le biais d'horloge sera considéré comme une inconnue à estimer au même titre que les trois composantes de la position. Par conséquent, un récepteur aura besoin de la réception du signal d'un quatrième satellite pour pouvoir calculer sa position [12].

I.3. Segments des systèmes de navigation par satellites

Les systèmes de navigation par satellites GNSS se reposent sur trois segments comme nous présente la figure suivante :

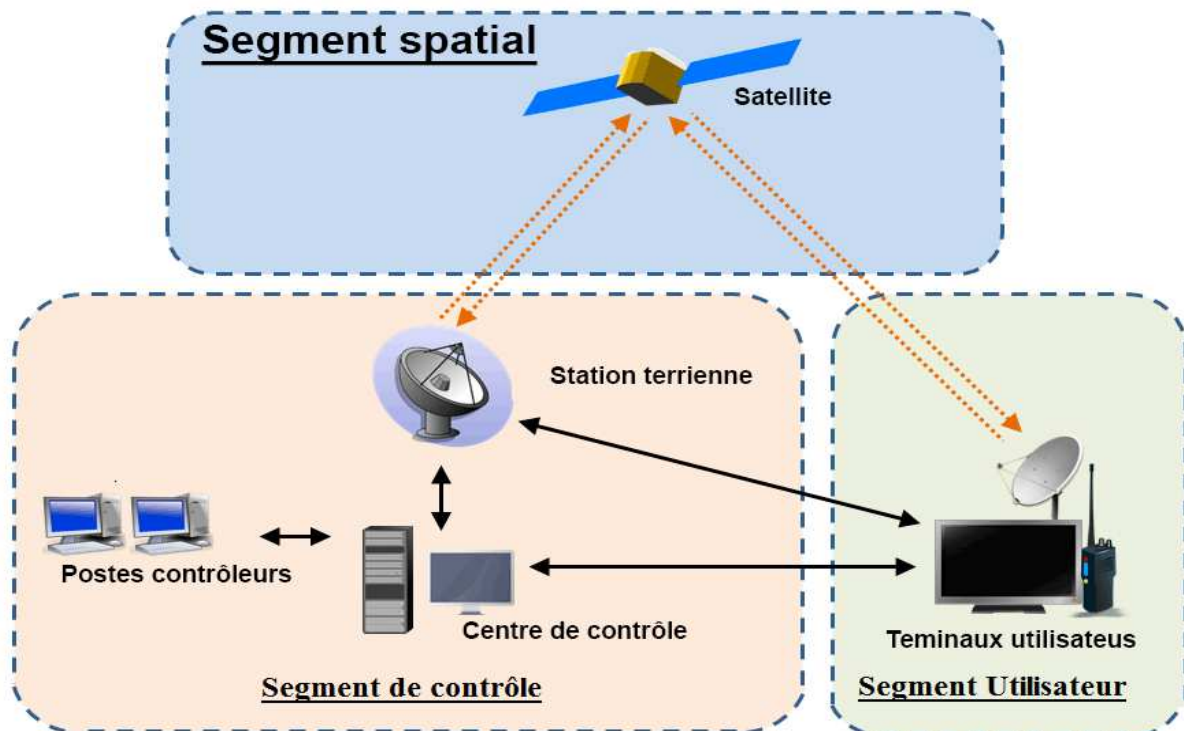


Figure I. 2. Segment du système GNSS.

I.3.1. Segment spatial

Il est constitué d'une constellation de satellites transmettant des signaux radios aux utilisateurs. L'Air Force gère la constellation de manière à assurer la disponibilité d'au moins 24 satellites 95 % du temps. Le premier satellite a été lancé en 1975 et le système n'est pleinement opérationnel que depuis 1995. A l'heure actuelle 32 satellites constituent la constellation appelée NAVSTAR plus quelques-uns de déclassés pouvant être réactivés si nécessaire. Les satellites naviguent dans l'Orbite terrestre moyenne (MOE) à une altitude d'environ 20 200 km et effectuent chaque jour 2 fois le tour de la terre. La configuration des trajectoires des satellites permet de capter les signaux d'au moins quatre satellites n'importe où à la surface du globe et à n'importe quel moment. Les satellites sont équipés d'une horloge atomique et émettent en permanence sur deux fréquences L1 (1 575,42 MHz) et L2 (1 227,60 MHz) modulées par des codes et par un message de navigation (éphémérides permettant le calcul de la position des satellites) (Figure 1). Parmi les codes « portés » par la fréquence L1 (on parle également de Porteuse L1) on retrouve le code C/A (Coarse Acquisition, acquisition brute en français) entièrement ouvert aux civils depuis l'année 2000 et le code P (Précis) réservé aux utilisateurs autorisés par le gouvernement américain. Les récepteurs (ou « géonavigateurs », un GPS pour désigner le récepteur est un abus de langage) commercialisés dans le domaine civil utilisent le code C/A. Quelques récepteurs pour des applications de haute précision, comme la géodésie, mettent en œuvre des techniques permettant d'utiliser le code P (malgré son cryptage) [13].

I.3.2. Segment de contrôle

Le segment de contrôle surveille et télécharge les données vers les satellites, via des stations au sol à différents endroits dans le monde. Il surveille des paramètres tels que le comportement, l'état et les orbites des satellites, et met également à jour les horloges des satellites, les éphémérides, les almanachs, etc. Il envoie également d'autres informations aux satellites. Le segment de contrôle maintient l'horloge du système et génère des messages de navigation [10].

I.3.3. Segment utilisateur

Ce segment est composé des récepteurs pouvant recevoir les signaux des satellites du système. Ces récepteurs fournissent la position, la vitesse et le temps précis à l'utilisateur à tout instant et à tout endroit du globe terrestre. Ils sont très variés dans leur conception et leur fonction selon leur utilisation, leur précision ou la durée d'acquisition des signaux au dé but

de chaque utilisation. Ils disposent dans leur mémoire des répliques de codes PRN des satellites du système GPS.

Trois types de récepteurs ont été développés selon les architectures des canaux de Réception suivantes (les 2 premières sont désormais obsolètes) :

- ✚ Les récepteurs séquentiels, possédant un ou deux canaux. Ils traitent les signaux un à un et calculent la position après quatre mesures différentes. Comme les mesures des pseudos distances ne sont pas calculées en même temps, ces récepteurs donnent une position moins bonne.
- ✚ Les récepteurs multiplex possèdent un seul canal partagé séquentiellement suivant une fréquence de 50Hz entre les différents signaux satellitaires en vue.
- ✚ Les récepteurs multicanaux offrent les meilleures performances. Ils utilisent au moins quatre canaux pouvant suivre donc simultanément quatre satellites. Actuellement, les récepteurs commercialisés disposent de 15 à 20 canaux parallèles qui utilisent tous les satellites disponibles [14].

I.4. Systèmes mondiaux de navigation par satellite

I.4.1. Système de positionnement global GPS

Anciennement appelé NAVSTAR (Navigation System by Timing And Ranging), il s'agit d'un système de positionnement radio par satellite militaire américain développé et mis en service par la NGA (National Geospatial Intelligence Agency). La constellation GPS est constituée de 31 satellites placés sur des orbites quasi-circulaires avec une altitude d'environ 20 200 km et une période de moins de 12 heures. Assure la visibilité de 4 à 8 satellites avec des angles d'élévation supérieurs à 15° en tout point de la Terre. Il existe plusieurs classes de satellites par génération (Bloc I en 1978 au Bloc 3 en 2014). Utilisé pour contrôler et surveiller la constellation GPS. Constitué d'environ 15 stations réparties dans le monde entier, il rassemble tous les utilisateurs civils et militaires du système GPS et diffuse sur deux fréquences, L1 (1575,43 MHz) et L2 (1227,60 MHz). Les principales applications sont le positionnement absolu ou relatif par code et/ou phase et la datation précise des événements [15].

Deux types de services basés sur la mesure de code sont proposés aux utilisateurs:

- ❖ **SPS (Standard Positioning Service)** : Il s'agit d'un service de positionnement accessible à tous les utilisateurs équipés de récepteurs GPS. C'est un service gratuit et anonyme.

❖ **PPS (Precise Positioning Service)** : Il s'agit d'un service de positionnement GPS précis. Vous devrez obtenir la clé de déchiffrement auprès du département américain de la défense. Il est réservé aux militaires américains et à leurs alliés [16].

I.4.2. Système de navigation Russe GLONASS

GLONASS «GLObalnaya NAVigatsionnaya Sputnikovaya Sistema» est la version Russe du système GPS, c'est un système de radio navigation qui assure le positionnement et le temps aux utilisateurs, il est développé initialement pour l'armée soviétique et il est géré par le ministère de la défense Russe. Ce système peut être considéré comme la deuxième génération du système de localisation soviétique Tsikada qui a été lancés entre 1976 et 1995 et qui donne une précision de l'ordre de 100 mètres [17].

I.4.3. Système de navigation Galileo

Le système Galileo GALILEO est un système de positionnement par satellites développée par l'union européenne. Il permettra à un utilisateur muni d'un terminal de réception d'obtenir sa position sur une carte géographique. GALILEO couvrira toute la surface de la terre, chaque point donnée étant balayée à tout moment par 06 à 08 satellites [18].

Il sera composé d'une cinquantaine de stations réparties sur toute la planète et il assure cinq niveaux de service sont:

- ✚ **Open Service (OS)**: Le service ouvert, C'est le service qui correspond à l'utilisation civile du GPS actuel. Le service ouvert fonctionne sur deux bandes de fréquences: 1164–1214 MHz et 1563–1591 MHz. Un récepteur qui utilise les 2 bandes de fréquences peut obtenir une précision horizontale inférieure à 4 m et une précision verticale inférieure à 8 m. Si le récepteur n'utilise qu'une des deux fréquences, il aura une précision horizontale inférieure à 5 m et une précision verticale inférieure à 35 m. Ce qui est comparable aux performances du GPS actuel.
- ✚ **Commercial Service (CS)**: D'une précision à l'échelle du centimètre, le service commercial est crypté et est destiné à des applications spécialisées. Ce service utilise les 2 bandes de fréquences du service ouvert, ainsi qu'une bande fréquence de supplémentaire de 1260-1300 MHz. Ce qui permet une précision inférieure à 1 m.
- ✚ **Safety of Life Service (SoL)**: Le service de sûreté de la vie informera automatiquement les utilisateurs, en quelques secondes, de toute panne de satellite ou de tout problème similaire affectant la performance. Ce service sera mis à disposition pour des applications critiques sur le plan de la sécurité, telles que l'exploitation des trains, le guidage des voitures, la navigation et l'aviation.

- ✚ **Public Regulated Service (PRS):** Le service public réglementé, qui s'adressera aux utilisateurs remplissant une mission de service public (services d'urgence, transport de matières dangereuses, etc).
- ✚ **Search and Rescue Service (SAR):** Le service de recherche et secours mondial de Galileo aidera au transfert des signaux de détresse vers un centre de coordination des sauvetages. Les utilisateurs recevront un signal les informant qu'une aide est en route

I.4.4. Système de navigation Beidou

Beidou est un système de navigation et de positionnement par satellites construit en Chine avec des droits de propriété intellectuelle indépendants. Par rapport à d'autres systèmes, l'une des caractéristiques mise en évidence du système Beidou est qu'il est capable de communiquer des messages courts. Chronologiquement, le système expérimental (système de 1^{ère} génération) a commencé son développement depuis 1994 et a atteint des capacités de positionnement actif en 2000. La deuxième étape, qui est la construction du Beidou (système de deuxième génération) a commencé en 2004, et la capacité de positionnement passif régional a été atteinte en 2012. Enfin, pour la troisième étape, a été élaboré en 2020. Notons que le système BeiDou de troisième génération (BDS-3) est achevé la constellation de 35 satellites pour fournir un service mondial dans le mois de Juin 2020. Le système global transmet des signaux aux fréquences B1 (E1/L1), B2 (E5/L5) et B3 (~E6). Partageant des bandes de fréquences et des formes d'ondes de signaux étroitement similaires avec le GPS et Galileo, le BDS-3 contribue d'une manière significative au monde multi-GNSS interopérable [18].

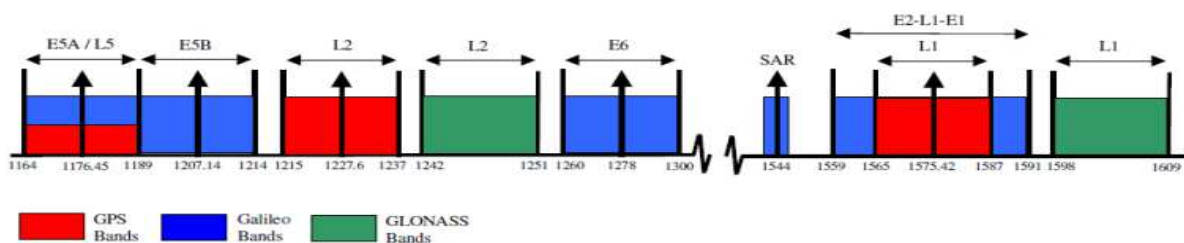


Figure I. 3. Bandes de fréquences allouées par les signaux GNSS.

I.5. Signaux GNSS

Le système GNSS est un système passif pour l'utilisateur; les satellites envoient des signaux et l'utilisateur reçoit passivement les recevoir et les traiter; le terme passif signifie que les récepteurs utilisateurs ne peuvent pas interagir avec le satellite. Cette caractéristique passive permet à un nombre illimité d'utilisateurs d'utiliser les systèmes GNSS en même temps. Le

système GNSS est composé de 3 segments: le segment utilisateur, le segment de contrôle et le segment spatial [16].

I.5.1. Principe de navigation par satellites

Certains pays ont déjà développé des systèmes de positionnement, et tentent d'améliorer les performances de ces systèmes, avec l'utilisation de nouveaux signaux et de nouvelles fréquences. Le GNSS comprend de nos jours les systèmes de positionnement par satellites tels que le GPS (États Unis), Galileo (Union Européenne), Glonass (Russie). En fait, le GPS est à l'origine du GNSS: au début des années 60, plusieurs départements du gouvernement américain tels que le département de la défense (DOD), la NASA (National Aeronautics and Space Administration), et le département des transports (DOT) s'intéressaient au développement d'un système de positionnement à 3 dimensions de l'espace par les satellites. Ce système devait être optimal tout en fournissant une couverture globale continue, pour toutes les conditions météorologiques: ce système est le GPS (Global Positioning Satellite System) [19].

I.5.2. Chaîne de réception GNSS

Les retards de propagation sont suivis dans le temps et les positions sont calculées via la chaîne de réception [20]. La Figure I.4 montre la chaîne du récepteur GNSS.

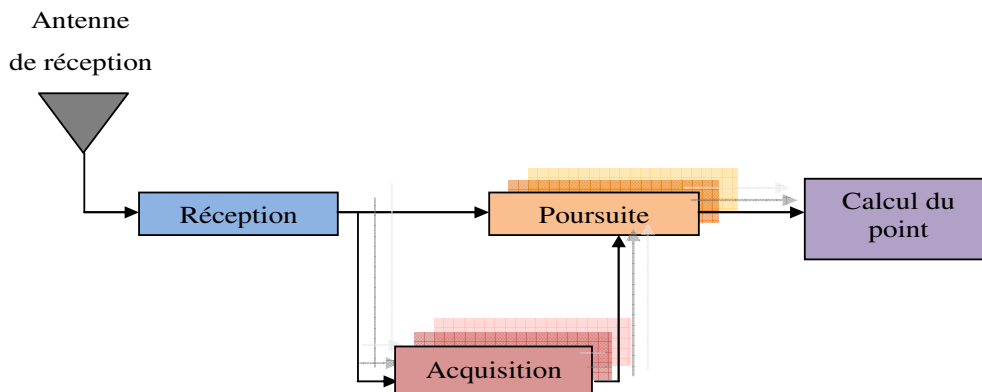


Figure I. 4. Chaîne de récepteur GNSS.

Entre la réception du signal et le calcul de la position de l'utilisateur, le récepteur doit effectuer plusieurs étapes pour extraire les informations nécessaires au "calcul du point". Le schéma bloc classique d'un récepteur est donné à la Figure I.4. Le bloc "Réception" se réfère au bloc de traitement classique du signal à la réception (i.e., numérisation du signal, séparation des voies, etc.). S'en suit alors deux étapes qui ont pour le but de synchroniser le récepteur avec les signaux de navigation: l'acquisition (qui est réalisée en parallèle sur chacun

des canaux de réception) et la poursuite (qui peut être ou non réalisée en parallèle selon le choix de l'architecture de poursuite).

Les catégories de récepteurs sont aussi variées que les domaines d'applications [21]:

- ❖ Les récepteurs grand public.
- ❖ Les récepteurs certifiés pour les transports.
- ❖ Les récepteurs de qualité géodésique.
- ❖ Les récepteurs militaires.

I.5.3. Architecture d'un récepteur GPS

Un récepteur GPS est constitué de différents blocs permettant d'obtenir les différentes mesures nécessaires au calcul de la position [22]:

- Une antenne, permettant de transformer l'onde électromagnétique en signal électrique.
- Un bloc frontal radiofréquence, composé de hardware permettant la mise en bande de base, l'amplification et la numérisation du signal.
- Un bloc d'acquisition, permettant d'identifier le satellite dont le signal provient et d'avoir une première estimation des paramètres d'intérêt (temps de propagation, fréquence, phase du signal).
- Un bloc de poursuite, permettant de suivre au cours du temps l'évolution des différents paramètres d'intérêt.
- Un bloc de démodulation des données, permettant de récupérer les éphémérides nécessaires au calcul de la position, de la vitesse du satellite, et des paramètres permettant la correction des erreurs atmosphériques.
- Un bloc navigateur de Position Vitesse Temps, ou Position Velocity Time (PVT), permettant l'estimation des différents paramètres d'intérêt de l'utilisateur.

Cette configuration est représentée dans la Figure I.5.

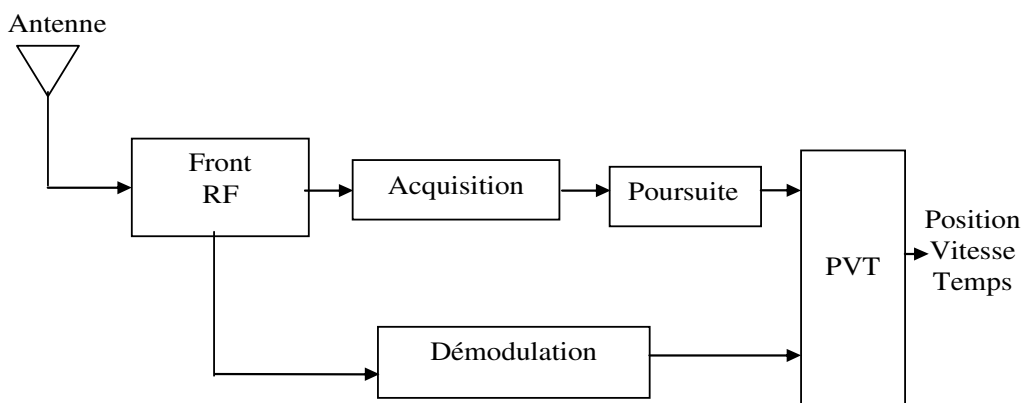


Figure I. 5. Schéma global d'un récepteur GPS.

I.5.4. Acquisition du signal GNSS

Cette opération obligatoire a pour le but de détecter la présence du signal d'un satellite dans l'ensemble des signaux reçus par l'antenne du récepteur. L'acquisition permet également de déterminer le décalage du code et de fréquence provoqués par la diffusion du signal et l'effet Doppler. De manière générale, l'acquisition est une recherche temps-fréquence qui consiste à calculer, sur un ensemble de délais Temps/Fréquence, les différentes corrélations entre le signal reçu et une réplique locale affectée par un délai et un Doppler fixé [3, 22].

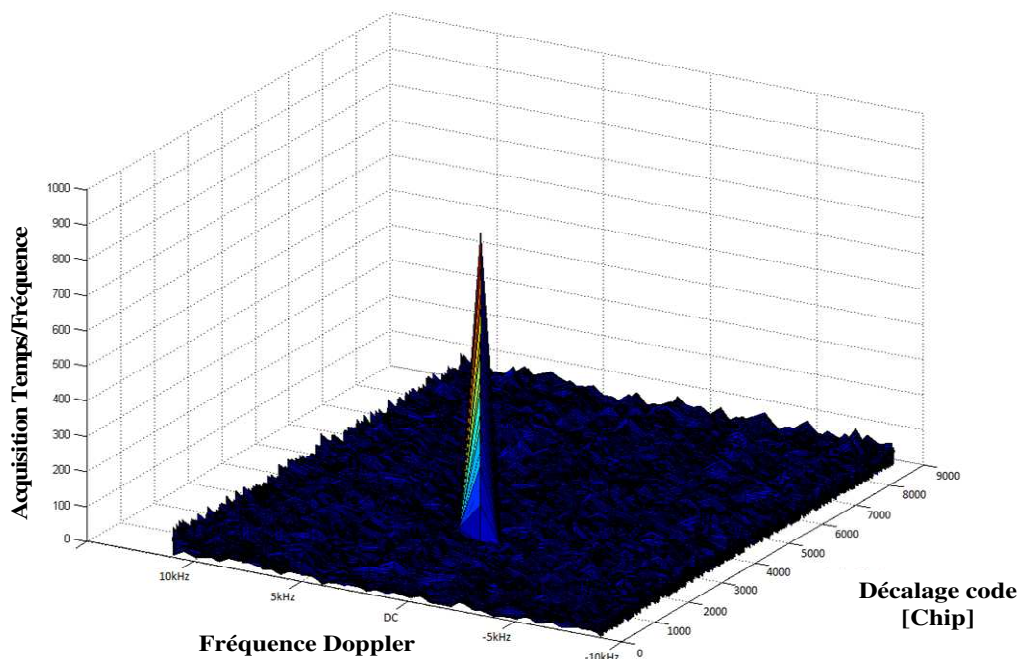


Figure I. 6. Exemples de résultat d'acquisition.

Les trois techniques d'acquisition des signaux GNSS sont les suivantes :

- ✓ Acquisition par recherche en série.
- ✓ Acquisition par recherche parallèle sur le retard de code par (FFT).
- ✓ Acquisition par recherche parallèle sur le décalage de fréquences par (FFT).

I.5.4.1. Acquisition par recherche en série

Cette méthode est souvent utilisée pour les systèmes CDMA (division de code à accès multiple). Cet algorithme est basé sur la multiplication du signal reçu par un code PRN généré localement, correspondant à un satellite donné et ayant une marge de déphasage de code de 0 à 1022 chips ; et une porteuse générée elle aussi localement (Figure I.7).

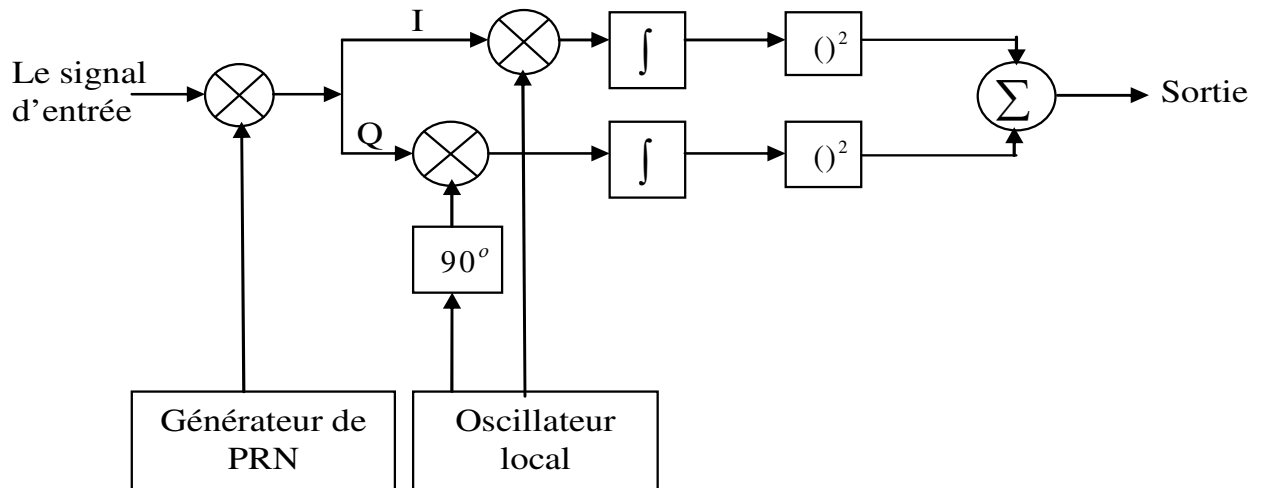


Figure I. 7. Principe d'acquisition par recherche série.

I.5.4.2. Acquisition par recherche parallèle sur le retard de code par (FFT)

L'objectif de la méthode d'acquisition par recherche parallèle sur le retard de code est d'effectuer une corrélation avec le signal entrant et un code PRN dans le domaine fréquentiel, pour modérer les inconvénients de la méthode de recherche en série. Le signal d'entrée est converti en deux composantes «en phase et en quadrature» qui sont utilisées en entrée comme partie réelle et complexe. Le résultat est multiplié par le conjugué de la Transformée de Fourier (FFT) du code d'étalement à générer du satellite recherché. Enfin le résultat sera obtenu en prenant l'amplitude de la Transformée de Fourier Inverse (IFFT) du produit (pour le retour au domaine de temporaire) [23]. On présente dans la Figure I.8 le principe de fonctionnement de cette approche.

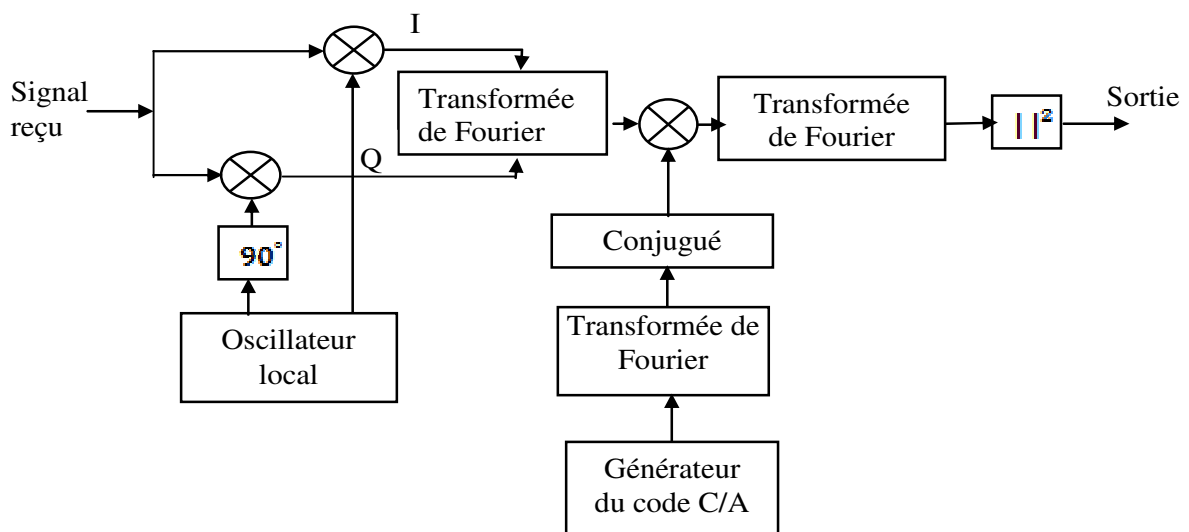


Figure I. 8. Acquisition par recherche parallèle sur le retard de code par (FFT).

I.5.4.3. Acquisition par recherche parallèle sur le décalage de fréquences par (FFT)

Le signal d'entrée est multiplié par une séquence PRN localement générée avec un code correspondant à un satellite spécifique et une phase de code entre 0 et 1022 chips. Le signal résultant est transformé au domaine fréquentiel par une Transformée de Fourier. La Transformée de Fourier peut être implémentée comme une Transformée de Fourier discrète (DFT). Si le signal entrant contient des composantes du signal provenant d'autres satellites, ces composantes seront réduites au minimum en raison des propriétés de corrélation croisée des séquences PRN:

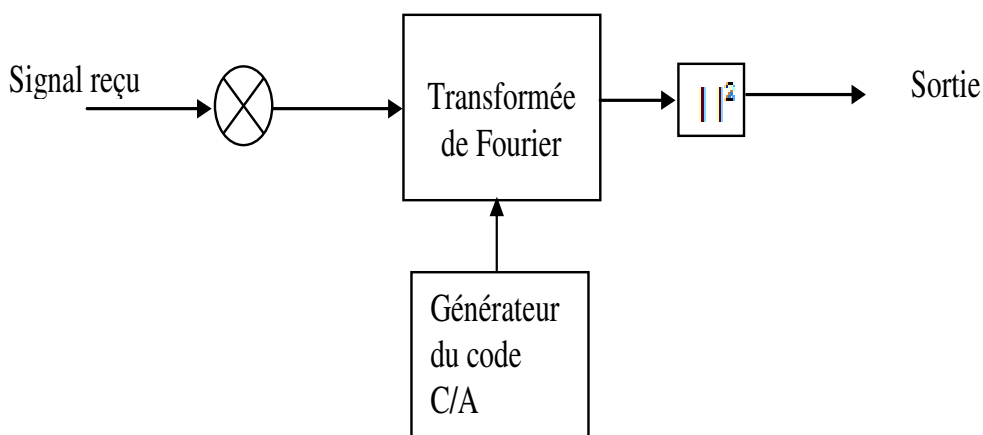


Figure I. 9. Schéma bloc de la recherche parallèle sur l'espace de fréquences par FFT.

Après la transformation du signal au domaine fréquentiel par l'algorithme FFT, il devient ainsi un signal complexe. Si le code généré localement est bien aligné avec le code dans le signal entrant, la sortie de la FFT aura un pic f_{IF} plus la fréquence de décalage Doppler.

I.5.5. Poursuite du signal GNSS

Une fois que l'acquisition a fourni des estimations de la fréquence Doppler et du décalage de code l'étape de la poursuite prend la relève pour suivre leurs évolutions. Pour cela les services de trois boucles d'asservissement doivent coopérer : (i) La boucle à verrouillage de fréquence FLL (Frequency Lock Loop) poursuit la fréquence porteuse. (ii) La boucle à verrouillage de phase PLL (Phase Lock Loop) poursuit la phase, (iii) et la boucle à verrouillage de délai de code DLL (Delay Lock Loop) poursuit les décalages du code.

Dans le système de poursuite (voir la Figure I.10), ces trois boucles sont imbriquées. En effet, la sortie de la boucle de poursuite de la porteuse sera utilisée lors de la démodulation de la boucle de poursuite du code. Les boucles de poursuite PLL et FLL utilisent le code synchronisé, fourni par la boucle de code pour le dés-étalement du signal [24].

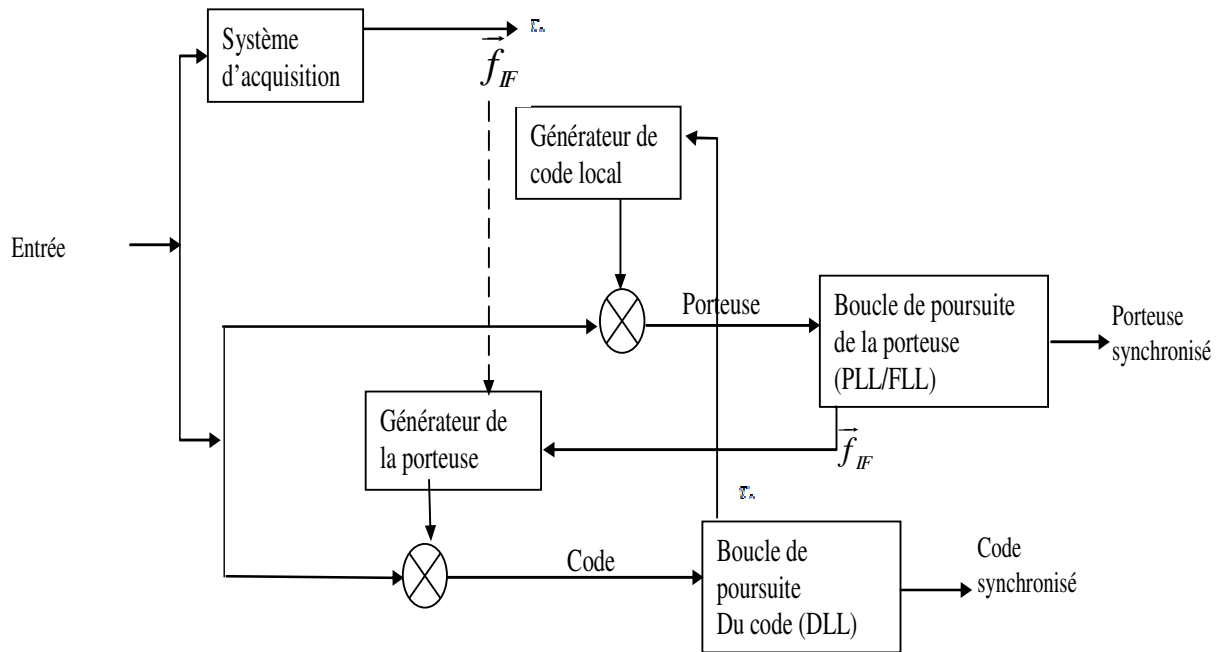


Figure I. 10. Schéma fonctionnel d'une boucle de poursuite du code et de la porteuse.

Le schéma bloc de l'étape de poursuite est donné à la Figure I.10. Différentes mesures peuvent être réalisées par le récepteur lors de la poursuite pour estimer les différentes distances utilisateur/satellite.

I.6. Classification de la source interférente

La classification des principales perturbations pour les récepteurs GNSS prend en compte des aspects hétérogènes. Les types d'émission peuvent être définis comme étant intentionnels (brouillage) ou non intentionnels. Les premiers sont courants pour les scénarios militaires même si le brouillage des applications civiles commence à être courant en raison de la disponibilité des dispositifs de brouillage sur le marché. De plus, un grand nombre de systèmes de communication présents dans notre vie quotidienne émettent un signal qui pourrait interférer avec la bande L du GNSS, en raison d'émissions hors bande par ces systèmes électroniques [10].

Nous passons maintenant à une discussion sur la classification des sources interférentes, basées sur leurs caractéristiques spectrales et temporelles.

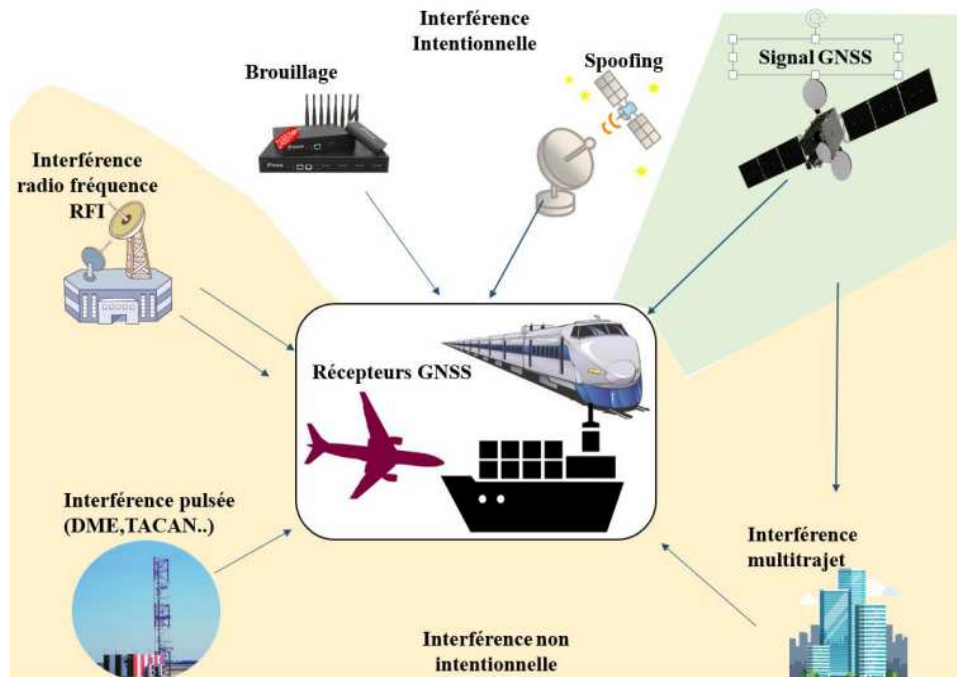


Figure I. 11. Différents types d'interférences sur les récepteurs GNSS.

La Figure I.11 montre les différentes sources d'interférences. Les interférences GNSS se présentent sous de nombreuses formes et classifications telles que la bande passante du récepteur Intérieur (dans la bande) ou la bande passante du récepteur extérieur (hors bande).

I.6.1. Caractéristiques spectrales d'interférence

Une classification générale des signaux d'interférence est basée sur leurs caractéristiques spectrales telles que la fréquence porteuse f_{int} et la bande passante B_{int} . Les interférences peuvent être classées selon leurs caractéristiques dans le domaine fréquentiel comme suit [3]:

- **Interférence à bande étroite (NBI):** L'occupation spectrale est plus petite comparée à la largeur de bande du signal GNSS ($B_{int} \ll B_{GNSS}$).

Les méthodes de génération d'interférences à bande étroite sont également différentes, peuvent utiliser des interférences à bande étroite autorégressives et peuvent également être équivalentes à la superposition de plusieurs signaux d'interférence à fréquence unique; aussi, le signal d'interférence est considéré comme un signal modulé par un filtre à bande étroite. L'interférence à fréquence unique est un signal spécial à bande étroite, qui est essentiellement une onde continue à fréquence unique. Ce signal d'interférence peut interférer avec un ou plusieurs points de fréquence dans le système de navigation par satellite. L'expression est donnée par la relation suivante:

$$J(t) = \sqrt{2p_j} \cos(\omega_j t + \phi_j) \quad (I.3)$$

où ω_j est la fréquence angulaire de signal interférence, p_j est la puissance, et ϕ_j est la phase aléatoire répartie sur $[0, 2\pi]$.

- **Interférence large bande (WBI):** L'occupation spectrale est comparable à la largeur de bande du signal GNSS ($B_{int} \approx B_{GNSS}$).
- **Interférence a ondes continues (CWI):** Représente la limite ultime en NBI et apparaît comme une seule tonalité dans le domaine de fréquence ($B_{int} \rightarrow 0$). Ce type d'interférence est modélisé par l'équation suivante:

$$J_{SCWI}(t) = J \times \cos(2\pi(f_{IF} \pm \Delta f)t + \Phi) \quad (I.4)$$

où J est l'amplitude du signal d'interférence, f_{IF} est la fréquence de l'interférence reçue, Δf est la différence par rapport à la fréquence centrale du signal GNSS, et Φ est une phase initiale aléatoire uniformément distribuée dans l'intervalle $[-\pi, \pi]$.

Il peut également s'agir de formes d'onde modulées ou de bruit blanc ainsi que pulsé ou continu dans le temps comme suit :

a) Interférences à ondes continues (CW) et à bruit blanc Gaussien additif (AWGN) :

Les signaux CW et AWGN sont parmi les formes d'interférences les plus connues rencontrées par un récepteur. Les interférences CW peuvent provenir de plusieurs sources. Une interférence indésirable est souvent causée par les harmoniques externes à la bande des amplificateurs ou des émetteurs. Les brouilleurs GNSS, qui sont devenus de plus en plus répandus, utilisent également des signaux CW pour interrompre la réception du signal GNSS. Le module AWGN peut également être utilisé pour simuler les bruits d'antenne et d'amplificateur ou pour évaluer la performance en présence de brouilleurs de bruit.

b) Interférence à onde continue multiple (MCWI): Il s'agit d'un phénomène où plusieurs signaux d'interférence continus sont émis dans différentes fréquences ou canaux, ce qui peut perturber la réception des signaux de communication normaux. Ce type d'interférence est modélisé par la formule suivante:

$$J_{MCWI}(t) = \sum_{i=1}^N J_i \times \cos(2\pi(f_{IF} \pm \Delta f_i)t + \Phi_i) \quad (I.5)$$

où: J_i , Δf_i et Φ_i sont respectivement l'amplitude, la différence de fréquence et la phase aléatoire de la $i^{\text{ème}}$ tonalité.

De plus, en général, les interférences peuvent avoir des fréquences caractéristiques variables, par exemple, les signaux modulés en fréquence «Chirp Signal» caractérisés par une variation linéaire en temps de la fréquence instantanée, apparaissent ainsi comme WBI. Ce type de

signal d'interférence est généralement généré par les brouilleurs «Jamming». De tels dispositifs qui sont capables de transmettre des signaux de forte puissance, le signal Chirp balaye plusieurs mégahertz en quelques microsecondes, obscurcit ainsi la bonne réception du signal de chaque canal GNSS. Ce type de signal de brouillage intentionnel gagne de plus en plus d'attention dans les applications civiles [10].

Les CWI pourraient avoir un impact grave sur un récepteur GNSS, soit sur l'acquisition soit sur le processus de poursuite, car la puissance de l'interférence est dispersée sur tout l'espace de recherche par la corrélation avec le code local, ce qui affaiblit la précision de l'acquisition et affecte les autres blocs fonctionnels. L'impact CWI et NBI dépend fortement de la valeur de la fréquence centrale de l'interférence dans la bande de fréquence.

1.6.1.1. Les interférences pulsées

Les signaux d'interférences pulsées sont caractérisés par un état d'activation/désactivation de courte durée (ordre de microsecondes), qui alternent dans le domaine temporel. Ce type de signal d'interférence est typique des scénarios aéronautiques, dans lesquels plusieurs services de radionavigation diffusent des signaux à fortes puissances dans une bande passante qui est partagée avec certains systèmes de navigation par satellites [10].

Les paramètres utilisés pour décrire l'interférence pulsée sont les suivants :

- **Largeur d'impulsion (PW):** Durée d'une impulsion;
- **Fréquence de répétition des impulsions (PRF):** Nombre d'impulsions par seconde;
- **Cycle de service (DC=PRF*PW):** Pourcentage de temps occupé par les impulsions.

Les signaux parasites pulsés sont caractérisés par un état marche-arrêt de courte durée (ordre des microsecondes), qui alternent dans le domaine temporel. Ce type de signal d'interférence est typique des scénarios aéronautiques, les services aéronautiques de radionavigation (ARNS) diffusent des signaux pulsés puissants dans une bande passante partagée avec certains des systèmes de navigation par satellite.

La modalisation de l'interférence pulsée (DME) est donnée par l'équation suivante :

$$J_{DME}(t) = \sqrt{P} \sum_{i=1}^M \left(e^{-\frac{\alpha(t-t_k)^2}{2}} - e^{-\frac{\alpha(t-\Delta t-t_k)^2}{2}} \right) \times \cos(2\pi(f_{IF} \pm \Delta f)t + \Phi) \quad (I.6)$$

où : P est la puissance du pic du DME/TACAN à l'antenne, t_k est l'ensemble des temps d'arrivée des paires d'impulsions, f_{IF} est la fréquence de l'interférence reçue, Φ est la phase de la porteuse du signal d'interférence.

1.7. Sources potentielles d'interférence

L'interférence potentielle peut partager les fréquences GNSS (RFI dans la bande) ou être loin de la porteuse GNSS (RFI hors bande). Il n'y a presque pas d'émissions autorisées dans les largeurs de bande GNSS. Cependant, les interférences viennent principalement des émissions non essentielles de systèmes hors bande, qui génèrent l'harmonique qui entre en collision avec les largeurs de bande GNSS [25].

I.7.1. Signaux hors bande

Dans cette sous-section, quelques-unes des principales interférences potentielles hors bande sont analysées [3, 25]:

- ✚ **Chaînes TV analogiques:** Les émissions de télévision sont de véritables sources d'interférences pour un récepteur GNSS. Elles peuvent être de type NBI ou WM : les ondes vidéo sont considérées comme des signaux à bande moyenne/large alors que les ondes sonores sont considérées comme des ondes CWI. Dans le signal TV diffusé, des antennes à très haute fréquence (VHF) et UHF sont utilisées. Ainsi, les deux extrémités des mains, dans leur sous-canal, peuvent représenter des sources d'interférence pour un récepteur GNSS.
- ✚ **Signaux DVB-T:** La norme DVB a été constituée en 1993 par le projet européen DVB. L'objectif principal de ce projet, qui comprend aujourd'hui plus de 300 membres européens et extra-européens, est d'harmoniser les stratégies d'introduction de la télévision numérique et des nouveaux services interactifs multimédia sur les réseaux de transmission et de définir les spécifications techniques correspondantes. Certains résultats sont obtenus en définissant les spécifications du système pour la norme DVB-S, développée pour la diffusion directe de programmes multiples de télévision par satellite et pour la norme DVB-C pour la distribution de signaux de télévision à travers les réseaux centraux. La famille DVB comprend également la norme DVB-T, qui fournit la télévision numérique terrestre. Sa définition a été réalisée en novembre 2005 avec la norme approuvée en février 1997 par l'ETSI.
- ✚ **❖VHF COM:** D'autres systèmes de communication VHF peuvent être dangereux pour un récepteur GNSS. La bande VHF (118–137 MHz) contient 760 canaux espacés de 25 kHz, et il est couramment utilisé par les communications du contrôle de la circulation aérienne (ATC). Les harmoniques sont considérées comme du NBI avec une largeur de bande d'environ 25 kHz. Les canaux VHF, centrés à 121,150, 121,175

et 121,200 MHz, avoir la 13^{ème} harmonique dans la bande passante GPS, alors que les canaux centrés à 131,200, 131,250 et 131,300 MHz sont dangereux pour la santé.

- ✚ **Harmoniques FM:** Les petites bandes de fréquences à l'intérieur de la bande FM (87,5–108 MHz) ont également des harmoniques qui tombent dans les bandes GNSS. Les canaux à 104,9 et 105,1 MHz avoir leurs 15 harmoniques à proximité des largeurs de bande GPS et Galileo.
- ✚ **Appareils électroniques personnels:** Les appareils électroniques personnels (PED) à proximité d'un récepteur GNSS peuvent la perturbation de la réception du signal GNSS. Les PED comprennent les téléphones portables, les téléavertisseurs, radios bidirectionnelles, jouets télécommandés, ordinateurs portables et bien d'autres. Un plus grand nombre On s'attend à ce que plusieurs PED incluent, à l'avenir, l'ultra-large bande (UWB) transmission qui permet le développement d'appareils personnels à haut débit.
- ✚ **SATCOM:** Les communications par satellite (SATCOM) fonctionnent dans les bandes de fréquences 1626–1660,5 MHz avec canaux espacés de 0,75 MHz et bande passante de 20 kHz. La transmission multi-porteuse dans un service SATCOM produit de l'intermodulation produits qui peuvent tomber dans la bande GNSS.
- ✚ ❖ **Service mobile par satellite (MSS):** Le système de service mobile par satellite (SMS) peut générer deux brouillages distincts menacés par un récepteur GNSS. Les stations terriennes mobiles du SMS utilisent la Bande 1610-1660,5 MHz, introduisant potentiellement une puissance large bande dans la Bande GNSS.

I.7.2. Signaux en bande

On distingue plusieurs types des signaux en bande, comme ceux représentés dans la suite :

- **Systèmes de communication aéronautiques Militaires/Civils:** Les systèmes de communications militaires peuvent être considérés comme des brouilleurs dans la bande en raison de la bande de signal utilisée par les systèmes. Les bandes Galileo E5a et E5b situées entre 1164 et 1214 MHz, occupent des fréquences déjà utilisées pour des applications aéronautiques, services de radionavigation (ARNS) tels que la navigation aérienne tactique (TACAN), équipement de mesure de distance (DME) et surveillance secondaire radar (SSR), ainsi que par le département commun de diffusion de l'information tactique du DoD Système (JTIDS) et le système de diffusion d'informations multifonctions(MIDS), d'autres systèmes aéronautiques fonctionnent dans ces fréquences telles que le Système de prévention des collisions et

des accidents de la circulation (TCAS), identifiant un ami ou un ennemi (IFF), et surveillance dépendante automatique - diffusion prévue (ADS-B).

- **Signaux ultra-large bande:** La définition d'UWB inclut tout signal occupant plus de 500 MHz entre 3,1 et 10,6 GHz. Les signaux UWB sont apparus en tant qu'une solution potentielle pour une consommation à faible complexité, à faible coût et à faible consommation d'énergie, et connectivité sans fil à haut débit de données. Les technologies basées sur UWB offrent une communication simultanée à haut débit, avec des débits de transmission de données de 100 à 500 *Mbps* à des distances de 2 à 10 m en même temps une puissance rayonnée moyenne de quelques centaines de microwatts Les systèmes UWB utilisent souvent les modulations : modulation en position d'impulsions (PPM) et la modulation d'impulsions en amplitude (PAM). Le signal UWB est généré en utilisant des impulsions sous-nanosecondes qui répartissent l'énergie du signal sur une large bande de fréquences. Ainsi, ces systèmes utilisent des signaux de faible puissance mais avec une bande passante extrêmement large. Cet aspect est critique pour les systèmes tels que les GNSS dont la puissance du signal est très inférieure au bruit de fond [23].
- **Interférences entre systèmes et intra-systèmes:** L'interférence inter-système et intra-système peut être considérée comme des perturbations de la bande. Le système Galileo a été développé pour assurer la compatibilité avec le GPS [26]. Les deux systèmes doivent fonctionner indépendamment l'un de l'autre ou avec un niveau minimum d'interférence entre eux. Le terme "inter-système" désigne la perturbation que l'émetteur GPS produit sur le signal Galileo et vice versa.

I.8. Brouilleurs

Les brouilleurs sont également appelés dispositifs de confidentialité personnels (PPD) est capable de brouiller ou de bloquer les signaux GNSS, ils sont destinés à de faire perdre la poursuite et d'empêcher la ré-acquisition du signal dans une zone limitée. Le brouillage fait référence à la transmission intentionnelle d'énergie RF pour empêcher le service de navigation en masquant les signaux GNSS avec du bruit. Les brouilleurs peuvent causer de graves dommages si leurs signaux ne sont pas correctement détectés et les effets atténués dans les récepteurs des utilisateurs [3].



Figure I. 12. Les brouilleurs.

I.9. Impact des Interférences radiofréquence sur les récepteurs GNSS

La distance entre le satellite et le récepteur est très importante et la puissance d'émission du satellite est assez faible, généralement le signal émis est immergée dans le bruit thermique du récepteur. Lorsque le récepteur est soumis aux impacts des divers interférences radiofréquence RF intentionnelles «Inconscientes», le signal GNSS ne peut pas être extrait du bruit, ce qui entraîne finalement une réduction de la précision de positionnement et une perte de sur la boucle de verrouillage de la poursuite.

Les interférences radiofréquence (RFI) sont généralement définies comme le phénomène qui se produit lorsque l'énergie RF rayonnée par une source externe provoque un dysfonctionnement d'un appareil électronique. D'une manière simple, les interférences peuvent être définies comme «quel que soit le signal, quel que soit le service, fonctionnant à la même fréquence que le récepteur satellite et pouvant l'ennuyer». Les interférences sur les récepteurs GNSS peuvent être classées en deux types : les interférences intentionnelles et non intentionnelles (voir la Figure I.11).

Les interférences intentionnelles comprennent le brouillage et les attaques d'imitation «Spoofing»; tandis que les interférences non intentionnelles incluent les signaux transmis indésirables (avec la même fréquence porteuse que les signaux GNSS), les interférences de trajets multiples et les interférences d'impulsions dans la bande [12, 27].

I.10. Conclusion

Ce premier chapitre couvre les concepts de base de la navigation par satellite, en particulier les systèmes de navigation, et offre une brève introduction aux principes fondamentaux des

signaux GNSS utilisés dans ce travail. Ensuite, nous présentons l'architecture des récepteurs GNSS en mettant l'accent sur la description précise des signaux, les performances d'acquisition et le suivi des signaux de navigation GNSS. Enfin, nous discutons les différents types d'interférences, de leurs caractéristiques et de leur effet sur les récepteurs GNSS.

Dans le chapitre suivant, nous décrivons les méthodes d'intelligence artificielle traditionnelles, telles que les réseaux de neurones multicouches, les réseaux de neurones à fonction radiale de base et les machines à vecteurs de support, qui sont utilisées pour les problèmes de classification. Dans le contexte de la classification des signaux GNSS, ces différentes méthodes d'intelligence artificielle peuvent être utilisées pour distinguer entre le signal utile et les interférences, ce qui permet de filtrer et de supprimer efficacement les interférences pour améliorer la qualité et la fiabilité des signaux GNSS utilisés dans la navigation par satellite.

Chapitre II

Apprentissage automatique

II.1. Introduction

L'apprentissage automatique (ML : Machine Learning) est un ensemble de techniques permettant de détecter les motifs (pattern recognition) dans les données et de décrire leurs caractéristiques, ce qui permet de prédire le comportement de nouvelles données. Des méthodes d'analyse des données sont utilisées pour traiter les données et en extraire des informations utiles [28].

L'apprentissage automatique basé sur les statistiques utilise des données d'entraînement pour déterminer les paramètres d'un modèle statistique, puis utilise ce modèle pour faire des prédictions sur de nouvelles données [29]. L'apprentissage automatique basé sur des modèles numériques utilise des données d'entraînement pour déterminer les paramètres d'un modèle numérique générique d'un certain type (par exemple, un réseau neuronal profond utilisant un très grand nombre de paramètres) [29].

L'apprentissage automatique est l'étude systématique d'algorithmes intelligents et de systèmes qui améliorent leurs connaissances ou leur performance grâce à l'expérience. Dans son concept général, le processus d'apprentissage automatique se réfère à la capacité de résoudre une tâche en traitant les bonnes caractéristiques décrivant le domaine d'intérêt, selon un modèle [30].

La procédure d'un problème d'apprentissage automatique qui doit être résolu, ainsi que les principaux éléments de l'algorithme d'apprentissage automatique, sont classés comme suit [30]:

- **Domaine** : Le problème à résoudre.
- **Caractéristiques** : La description des objets du domaine.
- **Tâche** : La représentation abstraite du problème qui se reflète dans la correspondance entre l'entrée et la sortie (la classification automatique des échantillons de collection de données).
- **Modèle** : La sortie de l'apprentissage automatique lorsque l'ensemble d'entraînement est alimenté dans les algorithmes.

Dans notre travail, le domaine d'application est celui des systèmes de navigation par satellite, et la tâche consiste à détecter et supprimer les interférences des signaux GNSS. Pour cela, nous essayons d'utiliser plusieurs types ou modèles d'apprentissage automatique pour distinguer entre le signal utile et les interférences. Ces modèles seront capables de traiter les données GNSS, afin d'améliorer la précision et la fiabilité des informations de navigation fournies par les systèmes de navigation par satellite [27, 31].

Dans ce chapitre, nous donnons un bref aperçu des algorithmes d'apprentissage automatique et de leurs différents types. Ensuite, nous abordons quelques méthodes de classification classiques, notamment le perceptron multicouche MLP, réseaux de neurones à fonction radiale de base RBF et la machine à vecteurs de support SVM. Nous discutons également de la détection des valeurs aberrantes à l'aide de la méthode des vecteurs de support à une seule classe (OCSVM). Ensuite, nous comparons les performances des méthodes SVM et OCSVM pour la détection des valeurs aberrantes. Enfin, nous avons présenté la méthode d'optimisation des paramètres des différents modèles d'apprentissage automatique utilisés dans notre travail.

II.2. Algorithmes d'apprentissage

L'algorithme d'apprentissage consiste à présenter au modèle des données d'entrée et à lui demander de modifier ses poids de manière à obtenir la sortie correspondante. Tout d'abord, les entrées sont propagées vers l'avant à travers le modèle jusqu'à ce que la sortie calculée soit obtenue. Ensuite, cette sortie est comparée avec la sortie désirée, et les poids sont modifiés de manière à minimiser l'erreur commise entre les sorties réelles et les sorties désirées lors de la prochaine itération. Ce processus est répété jusqu'à ce qu'une erreur de sortie négligeable soit atteinte, permettant ainsi au modèle de découvrir les dépendances entre les entrées et les sorties par lui-même [32].

Il existe deux principales méthodes d'apprentissage, qui sont détaillées dans les sous-sections suivantes :

II.2.1. Apprentissage supervisé

Il s'agit d'une méthode d'apprentissage qui utilise un ensemble de classes d'apprentissage connues afin d'ajuster un modèle statistique qui pourra être utilisé ultérieurement pour le déploiement.

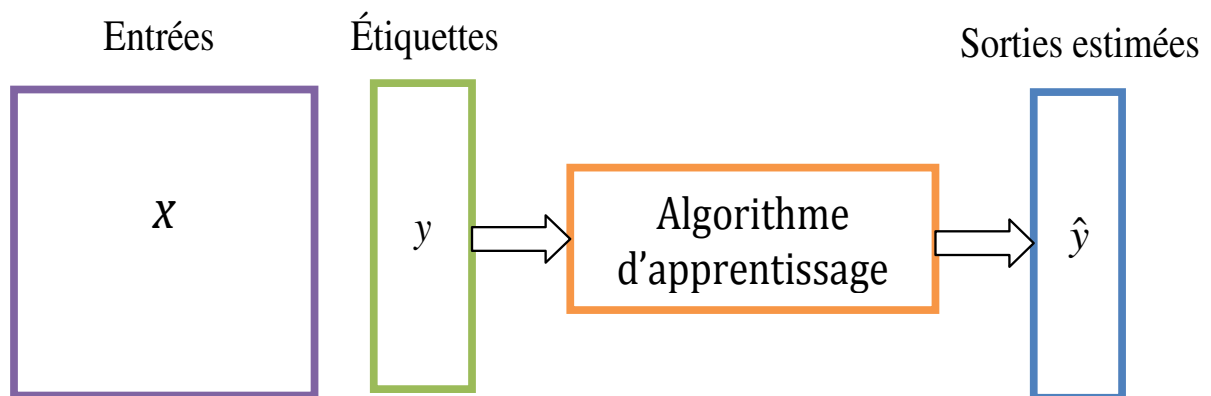


Figure II. 1. Schéma d'un modèle supervisé.

Cette méthode s'oppose aux méthodes non supervisées, dans lesquelles les classes ne sont pas connues (pas étiquetées) [33].

L'objectif est de déterminer à quel groupe l'individu a le plus de chances d'appartenir. Un expert doit préalablement étiqueter des exemples. Le processus se passe en deux phases. Lors de la première phase (hors ligne, dite d'apprentissage), il s'agit de déterminer un modèle des données étiquetées. La seconde phase (en ligne, dite de test) consiste à prédire l'étiquette d'une nouvelle donnée, connaissant le modèle préalablement appris. Parfois il est préférable d'associer une donnée non pas à une classe unique, mais une probabilité d'appartenance à chacune des classes prédéterminées (on parle alors d'apprentissage supervisé probabiliste), comme illustré dans la Figure II.1.

II.2.2. Apprentissage non supervisé

Dans le cas de l'apprentissage non supervisé, un ensemble de données d'entrées non étiquetées est utilisé, comme illustré dans la Figure II.2, afin de permettre à l'algorithme d'apprentissage de découvrir par lui-même les patterns ou les points communs présents dans cet ensemble de données. Les méthodes d'apprentissage automatique qui facilitent l'apprentissage non supervisé sont particulièrement utiles, car les données non étiquetées sont généralement plus considérables que les données étiquetées.

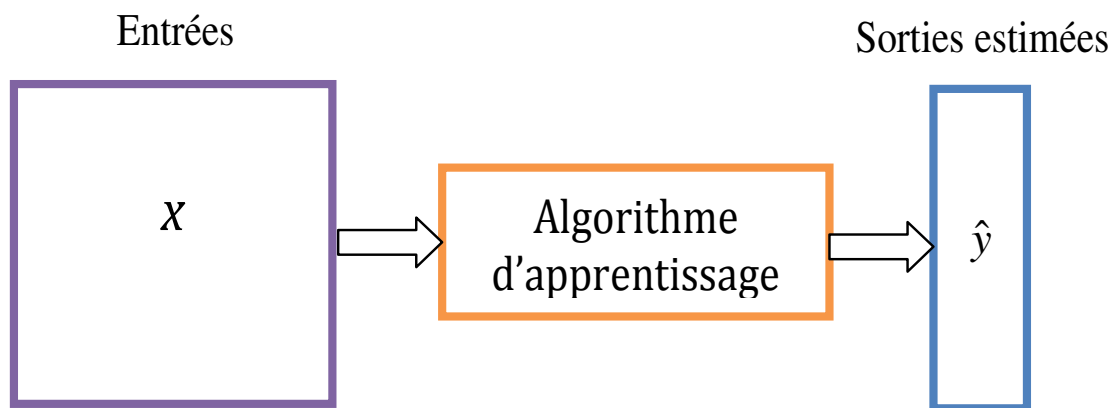


Figure II. 2. Schéma d'un modèle non-supervisé.

On peut considérer que l'objectif initial de l'apprentissage non supervisé est aussi simple que de détecter les modèles cachés dans un ensemble de données, mais il peut aussi avoir un objectif d'apprentissage des caractéristiques, ce qui va rendre la machine intelligente et capable de découvrir automatiquement les représentations nécessaires pour classer des données brutes.

II.2.3. Différence entre apprentissage supervisé et non supervisé

La principale différence entre ces deux types d'apprentissage est la disposition des données d'entrée, d'autres différences peuvent être trouvées dans le tableau ci-dessous [34].

Tableau I. 1. La différence entre l'apprentissage supervisé et non supervisé.

	L'apprentissage supervisé	L'apprentissage non supervisé
Les données d'entrée	Utilise des données connues et étiquetées en tant qu'entrée	Utilise des données inconnues qu'entrée non étiquetées
Nombre des classes	Connues	Inconnues
Précision des résultats	Résultats précis et fiables	Précision et fiabilité modère

II.3. Méthodes de classification traditionnelle

Les méthodes de classification sont des approches didactiques utilisées pour déterminer les classes auxquelles appartiennent les échantillons en se basant sur les paramètres descriptifs donnés. Elles permettent de trouver le chemin le plus approprié pour comprendre les motifs et les différences entre les différentes classes, et de déterminer quels modèles sont les plus adaptés pour gérer une étude de cas spécifique, tels que les réseaux de neurones MLP, réseaux de neurones à fonction radiale de base RBF, machines à vecteurs de support (SVM), et autres. Ces méthodes d'apprentissage sont utilisées pour modérer et attribuer les échantillons de paramètres descriptifs spécifiques à des catégories particulières [35].

II.3.1. Réseaux de neurones multicouches MLP

Chaque neurone artificiel est un processeur élémentaire. Il reçoit un nombre variable d'entrées en provenance de neurones en amont ou des capteurs composant la machine dont il fait partie. A chacune de ses entrées est associé un poids représentatif de la force de la connexion. Chaque processeur élémentaire est doté d'une sortie unique, qui se ramifie ensuite pour alimenter un nombre variable de neurones en aval. A chaque connexion est associé un poids. Il est commode de représenter graphiquement un neurone comme indiqué sur la Figure II.3.

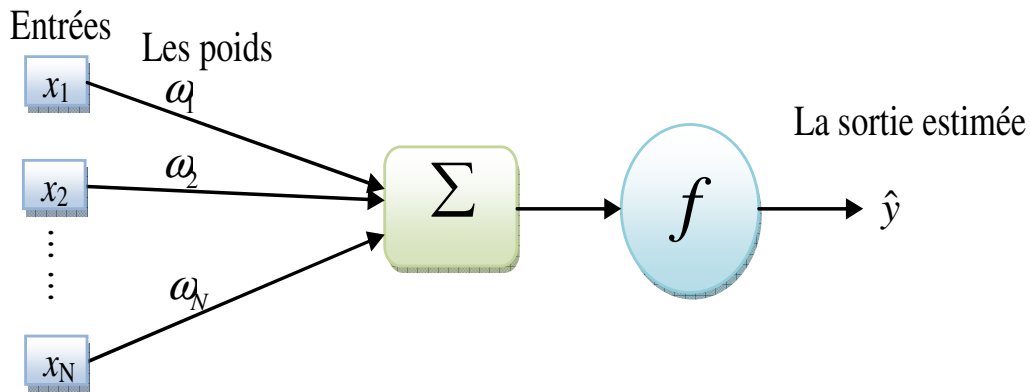


Figure II. 3. Neurone artificiel.

La généralisation d'un neurone artificiel est en effet réalisée à travers l'utilisation d'un réseau de neurones multicouche MLP.

Les réseaux multicouches sont construits selon une même logique et avec les mêmes briques élémentaires que les perceptrons monocouches (Figure II.4). L'unique différence réside dans le nombre de couches cachées, mais le fonctionnement reste identique : chaque neurone d'une couche reçoit une somme pondérée de toutes les sorties des neurones de la couche précédente (couche "dense"). Chaque neurone possède également un biais et une fonction d'activation : sa sortie, non linéaire par rapport à ses entrées, est envoyée à chaque neurone de la couche suivante, ou à la couche de sortie. Le réseau est "à propagation avant", c'est-à-dire que le signal ne se propage que de la gauche vers la droite, il ne revient jamais à contre-courant : les structures particulières.

Cet empilement de couches et de non-linéarités n'est pas nécessairement un avantage. Certes, les perceptrons multicouches satisfont aussi le théorème d'approximation universelle, mais c'était déjà le cas avec un perceptron monocouche : un perceptron monocouche, avec un nombre suffisant de neurones cachés, suffit pour pouvoir approximer toute fonction. L'ajout de couches supplémentaires n'apporte rien au cas de la régression standard.

Cet ajout de couches s'avère, par contre, utile lorsque les données d'entrée possèdent une structure, connue ou à découvrir. C'est cette caractéristique qui fera la force de ces réseaux multicouches qui évolueront vers les réseaux profonds exploités en Deep Learning (voir chapitre III). La structure complexe de ces réseaux combine des fonctions de classification, de réduction de dimensionnalité et de régression selon des descripteurs ou des variables latentes.

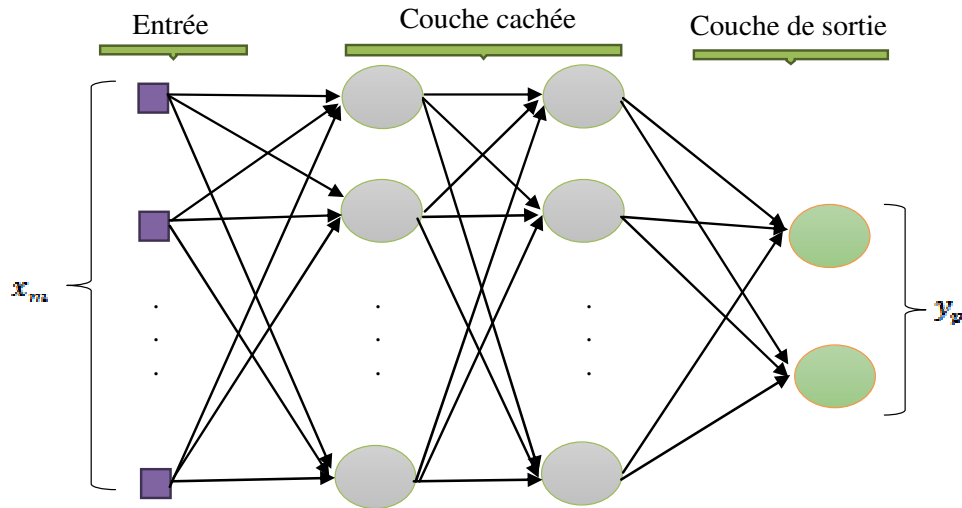


Figure II. 4. L'architecture d'un réseau de neurones de type MLP.

L'idée principale est de grouper des neurones dans une couche. La première couche est reliée aux entrées, puis ensuite chaque couche est reliée à la couche précédente. Les neurones de la première couche sont reliés au monde extérieur et reçoivent tous le même vecteur d'entrée. Ils calculent alors leurs sorties qui sont transmises aux neurones de la deuxième couche, etc.

Voici les caractéristiques du modèle, telles que représentées dans la Figure II.5:

- ✚ Il comporte une seule couche d'entrée et une seule couche de sortie.
- ✚ Il peut comporter une ou plusieurs couches cachées.
- ✚ Chaque neurone est relié uniquement à tous les neurones de la couche suivante.

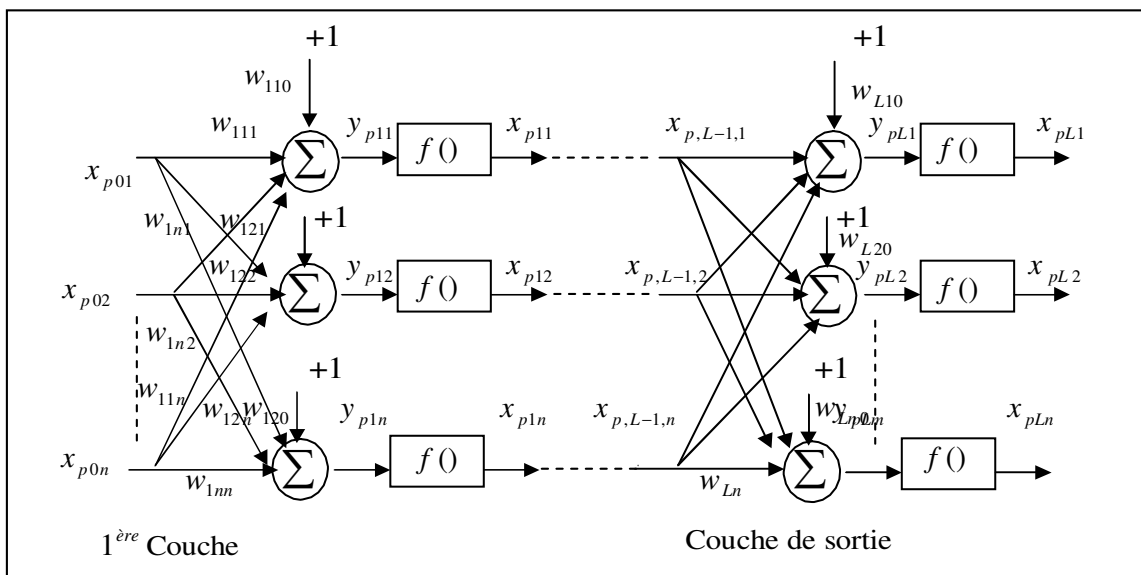


Figure II. 5. Structure d'un réseau de neurone multicouche.

L'MLP utilise généralement une technique de descente de gradient connu sous le nom de rétropropagation (RP) des erreurs comme un algorithme d'apprentissage.

L'algorithme de RP du gradient de l'erreur, est une technique de calcul des dérivées d'une entité par rapport à l'ensemble de ses entrées, utilisé pour résoudre le problème de calcul des erreurs de sorties des neurones des couches cachées. Cet algorithme change les poids d'un réseau dont l'architecture est fixée par l'utilisateur, repose sur la minimisation de l'erreur quadratique entre les sorties calculées et celles souhaitées.

L'erreur commise sur le $k^{\text{ème}}$ nœud de sortie est :

$$\delta_{pk} = O_{PK} - x_{plk} \quad (\text{II.1})$$

Par conséquent l'erreur totale (pour tous les nœuds) est :

$$E_p = \frac{1}{2} \sum_{k=1}^m \delta_{p,k}^2 = \frac{1}{2} \sum_{k=1}^m (O_{p,k} - x_{p,l,k})^2 \quad (\text{II.2})$$

Pour minimiser E_p (erreur quadratique), on calcule son gradient par rapport à chaque poids w , puis on modifie les poids dans le sens inverse du gradient [35].

II.3.1.1. Fonctions de transfert

Jusqu'à présent, nous n'avons pas spécifié la nature de la fonction d'activation $a = f(x)$ de notre modèle. Il se trouve que plusieurs possibilités existent et celles-ci sont quasiment empiriques et à adapter en fonction des situations. Les plus courantes et les plus citées dans la littérature scientifique sont énumérées dans la figure ci-dessous :

Les trois fonctions les plus utilisées sont : (a) "seuil" (en anglais "hard limit"), (b) "linéaire" et (c) "sigmoïde" comme représentées ci-dessous:

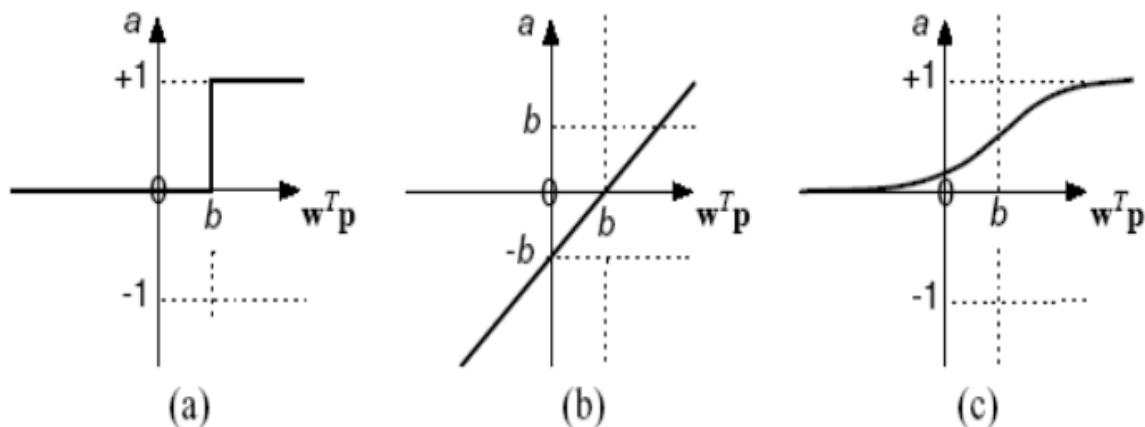


Figure II. 6. Les trois types de fonctions de transfert.

- **La fonction seuil:** Comme son nom l'indique, la fonction seuil applique un seuil sur son entrée. Plus précisément, une entrée négative ne passe pas le seuil, la fonction retourne la valeur 0 (faux), alors qu'une entrée positive ou nulle dépasse le seuil, et la fonction retourne 1 (vrai). Il est évident que ce genre de fonction permet de prendre des décisions binaires (cette fonction peut aussi être assimilée à la fonction de Heaviside pour ceux qui connaissent...).

La fonction seuil est donnée par la relation suivante:

$$f(x) = \begin{cases} 1 & \text{si } x > b \\ 0 & \text{si } x \leq b \end{cases} \quad (\text{II.3})$$

- **La fonction linéaire :** La fonction linéaire est quant à elle très simple, elle affecte directement son entrée à sa sortie selon la relation :

$$f(x) = x - b \quad (\text{II.4})$$

Il est évident que la sortie du neurone correspond alors à son niveau d'activation dont le passage à zéro.

- **La fonction sigmoïde :** La fonction de transfert sigmoïde est quant à définie par la relation mathématique :

$$f(x) = \frac{1}{1 + e^{-x}} \quad (\text{II.5})$$

II.3.1.2. Mise en œuvre des réseaux neuronaux

La mise en œuvre des réseaux neuronaux implique plusieurs étapes, notamment :

Étape 01) Fixer le nombre de couches cachées : Mis à part les couches d'entrée et de sortie, il faut décider du nombre de couches intermédiaires ou cachées. Sans couche cachée, le réseau n'offre que de faibles possibilités d'adaptation. Néanmoins, il a été démontré qu'un Perceptron Multicouches avec une seule couche cachée pourvue d'un nombre suffisant de neurones, peut approximer n'importe quelle fonction avec la précision souhaitée.

Étape 02) Déterminer le nombre de neurones par couches cachées : Chaque neurone peut prendre en compte des profils spécifiques de neurones d'entrée. Un nombre plus important permet donc de mieux "coller" aux données présentées mais diminue la capacité de généralisation du réseau. Il faut alors trouver le nombre adéquat de neurones cachés nécessaire pour obtenir une approximation satisfaisante.

Étape 03) Choisir la fonction d'activation : Le passage de cette dernière à la couche de sortie se fera selon les types de nos variables : Soit sigmoïde, soit linéaire, soit seuil.

Étape 04) Choisir l'apprentissage : A partir d'une architecture de réseau de neurones donnée et des exemples disponibles (la base d'apprentissage) on détermine les poids optimaux, par l'algorithme de la rétropropagation (RP) des erreurs, pour que la sortie du modèle s'approche le plus possible au fonctionnement désiré. L'apprentissage par RP nécessite la détermination du paramètre d'ajustement des poids synaptiques à chaque itération. La détermination du critère d'arrêt est aussi cruciale dans la mesure où la convergence peut passer par des minima locaux.

II.3.2. Réseaux de neurones à fonction radiales de base RBF

Le réseau RBF utilise le plus souvent la règle de l'apprentissage de correction d'erreur et la règle par apprentissage compétitif. Il peut avoir un apprentissage qui combine en même temps l'apprentissage supervisé et l'apprentissage non supervisé [36].

Chaque neurone RBF contient une Gaussienne qui est centrée sur un point de l'espace d'entrée. Pour une entrée donnée, la sortie du neurone RBF est la hauteur de la gaussienne en ce point. La fonction Gaussienne permet aux neurones de ne répondre qu'à une petite région de l'espace d'entrée, région sur laquelle la gaussienne est centrée.

Donc il y a quatre paramètres principaux à régler dans un réseau RBF, telles que représentées dans la Figure II.7:

- ✚ Le nombre de neurones RBF (nombre de neurones dans l'unique couche cachée).
- ✚ La position des centres des Gaussiennes de chacun des neurones.
- ✚ La largeur de ces gaussiennes.
- ✚ Le poids des connexions entre les neurones RBF et le(s) neurone(s) de sortie.

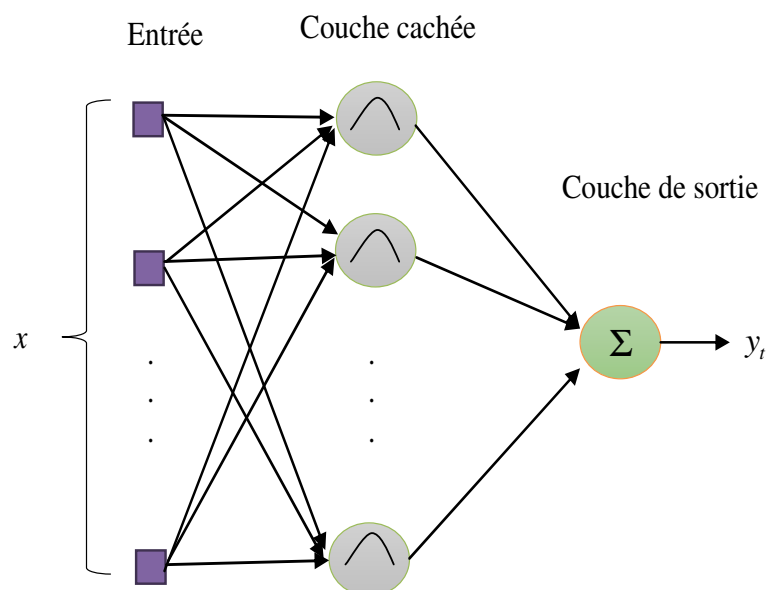


Figure II. 7. Architecture d'un réseau de neurones à base de fonctions radiales.

Toute modification d'un de ces paramètres entraîne directement un changement du comportement du réseau RBF.

Nous disposons d'un ensemble d'entrées x et un ensemble de sorties y . L'approximation par un RBF sera notée. Cette approximation sera la somme pondérée de m fonctions gaussiennes Φ [22].

$$\hat{y}_t = \sum_{i=1}^m \lambda_i \Phi(x_i, c_i, \sigma_i) \quad (\text{II.6})$$

où

$$\Phi(x_i, c_i, \sigma_i) = e^{-\left(\frac{\|x_i - c_i\|}{\sqrt{2}\sigma_i}\right)^2} \quad (\text{II.7})$$

avec σ_i est la largeur de la fonction d'activation, et c_i sont les centres des Gaussiennes de chacun des neurones.

II.3.3. Machines à Vecteurs de Support (SVM)

Les machines à support de vecteurs ont été appliquées avec succès dans divers domaines comme la vérification et la reconnaissance, telle que l'identification de visages, la reconnaissance de caractères manuscrits et des chiffres, la vérification et reconnaissance du discours et du parlant et la prédiction. Les SVM constituent une classe spécifique d'algorithmes qui est caractérisée par, l'utilisation de *kernels* (fonctions noyaux), la capacité d'arriver à un résultat optimal global, la faible densité de la solution et sa capacité de contrôle en agissant sur la marge ou sur une autre quantité indépendante de la dimension comme le nombre de vecteurs de support.

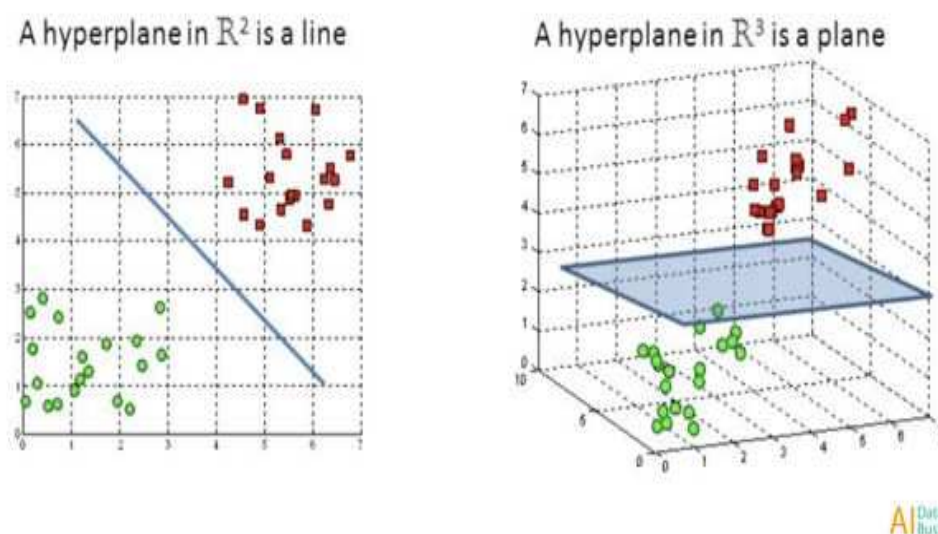


Figure II. 8. Transformer l'espace de départ vers un espace de dimension supérieure.

Ces caractéristiques font en sorte qu'elle soit une des techniques dont l'application peut être très adéquate pour la classification des données, comme montré dans la Figure II.8.

II.3.3.1. Principe des SVMs

Les SVMs sont conçus pour mettre en œuvre le principe du risque structurel. Grâce à leur base théorique solide, ces classifieurs ont requis un très grand succès dans les différents domaines de traitement de données. En outre, ils répondent à deux problèmes majeurs de la théorie de l'apprentissage statistique qui sont:

- ✚ Le contrôle de la capacité du classifieur.
- ✚ Le sur-apprentissage de données.

Dans la Figure suivante, on détermine un hyperplan qui sépare deux ensembles de points.

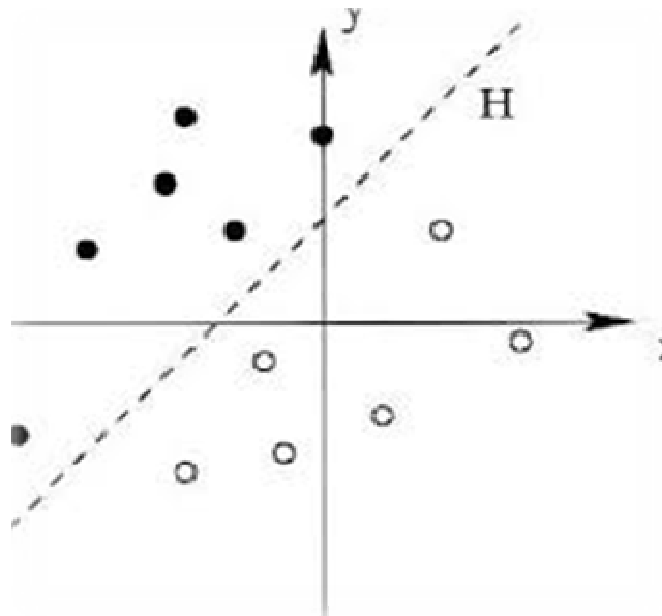


Figure II. 9. Séparation de deux ensembles de points par un hyperplan H.

Les points les plus proches, qui seuls sont utilisés pour la détermination de l'hyperplan, appelés vecteurs de support. Il existe une infinité d'hyperplans capable de séparer parfaitement les deux classes d'exemples. Le principe des SVM est de choisir celui qui va maximiser la distance minimale entre l'hyperplan et les exemples d'apprentissage (la distance entre l'hyperplan et les vecteurs supports), appelée la marge le fait d'avoir une marge plus large procure plus de sécurité lorsque l'on classe un nouvel exemple (voir la Figure II.10).

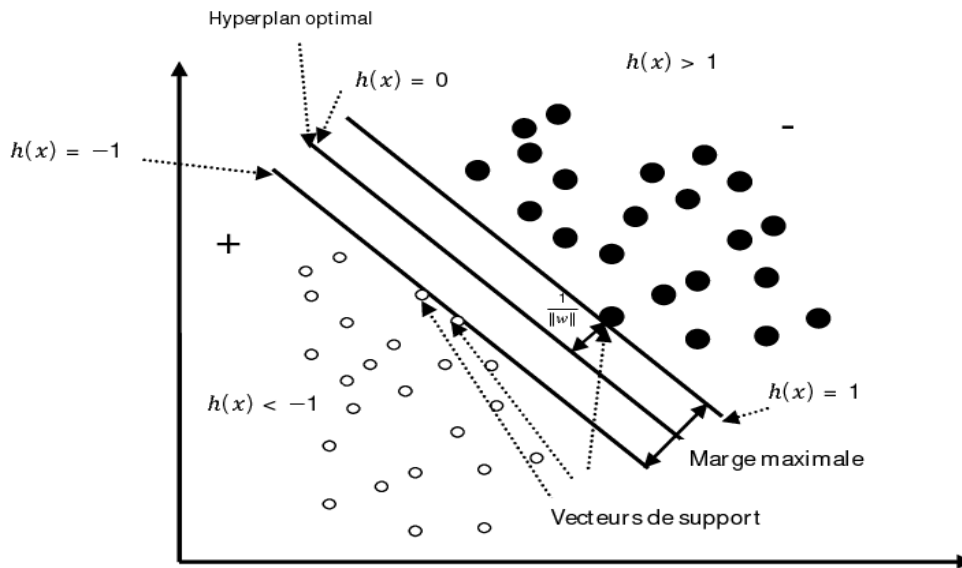


Figure II. 10. Cas linéairement séparable.

S’il existe un hyperplan qui sépare les deux classes, les points de l’hyperplan sont décrits par l’équation suivante [32]:

$$x_i w + b = 0 \tag{II.8}$$

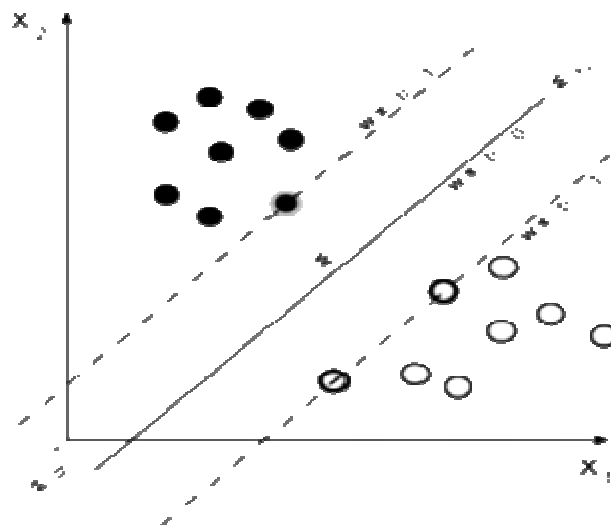


Figure II. 11. Illustration d’un classificateur SVM dans le cadre d’une classification binaire.

où : w est la normale au plan et $b/\|w\|$ la distance entre l’hyperplan et l’origine et $1/2\|\omega\|$ est la marge maximal, les points en rouge sont les vecteurs de support [37]. Pour estimer la classe d’un nouveau point x , on calcule [37] :

$$f(x) = \text{sign}(x\omega^\circ + b^\circ) = \text{sign}\left(\sum \alpha_i^\circ y_i (x_i - x) + b^\circ\right) \tag{II.9}$$

où : ω^o , b^o sont des constantes qui vérifient l'hyperplan optimal et les α^o sont les multiplicateurs de Lagrange.

Revenons à notre problème qui est la classification entre le signal utile et l'interférence dans les récepteurs GNSS. Une autre variante de la méthode traditionnelle des SVM, plus efficace pour la détection des anomalies, est l'OCSVM (One-Class Support Vector Machine).

L'avantage de l'OCSVM dans la détection d'interférences réside dans le fait qu'il peut être plus robuste à la présence de données d'interférence peu représentées ou inconnues. Il peut également être utilisé pour détecter des anomalies de manière non linéaire en utilisant des fonctions noyau, ce qui lui confère une flexibilité supplémentaire dans la détection d'interférences dans des données complexes.

Dans la section suivante, on donne un aperçu sur la méthode OCSVM, son principe et une comparaison avec la méthode traditionnelle SVM.

II.3.4. Machine à vecteur de support à une classe OCSVM

La machine à vecteurs de support à une classe (OCSVM) est une approche largement utilisée pour la classification à une classe, le problème de la distinction d'une classe de données du reste de l'espace des caractéristiques [38]. Comme illustre la Figure II.12, l'OCSVM n'a besoin que de données d'une seule classe pour l'entraînement, généralement la classe normale ou non-anormale. L'objectif de l'OCSVM est de trouver un hyperplan qui encapsule la région normale des données dans un espace de caractéristiques tout en maximisant la marge autour de ces données normales.

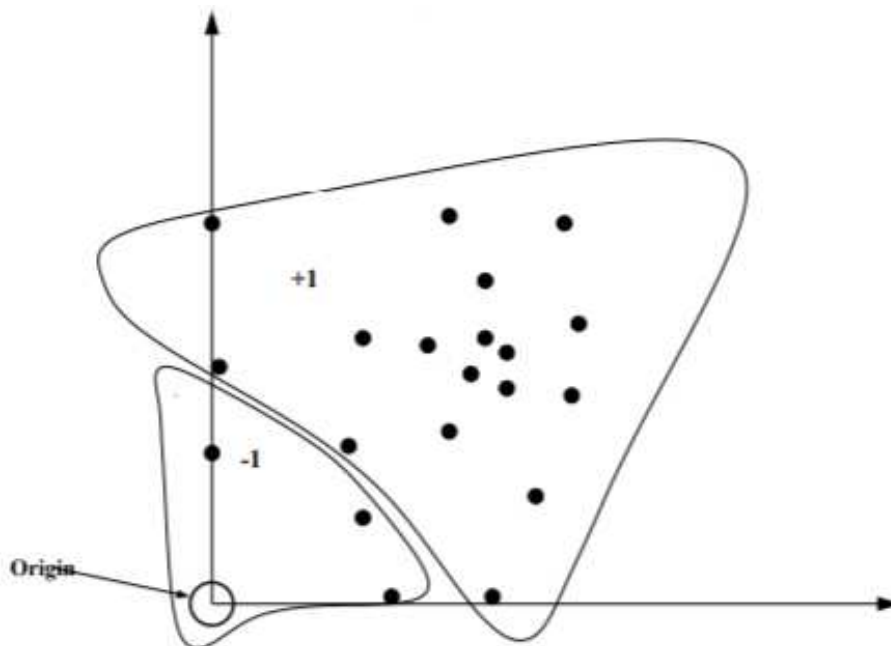


Figure II. 12. One-class SVM.

II.3.4.1. Le principe d'OCSVM

Son principe de base consiste à séparer à l'aide d'un hyperplan avec une marge maximale les données (Target) du point origine, unique représentant des données de l'out-of-class. Cette approche est très intéressante car elle permet de simuler une approche de classification binaire standard et donc de résoudre la tâche de classification one-class à l'aide d'une formulation identique à celle du SVM standard.

Les données sont donc projetées sur une hypersphère de rayon unitaire centrée à l'origine de H (hyperplan). La phase d'apprentissage d'un OCSVM consiste à définir l'hyperplan séparateur $w = \{h \in H : \langle h, w \rangle - b = 0\}$ dans H tel que marge $b/\|w\|$ soit maximum. La Figure II.13 explique le principe d'OCSVM.

Les paramètres w et b sont les solutions du problème d'optimisation suivant :

$$\min_{w, \xi, b} \frac{1}{2} \|w\|^2 - b + \frac{1}{vl} \sum_{i=1}^l \xi_i \quad (\text{II.10})$$

$$\begin{cases} \langle w, \phi(x_i) \rangle - b \geq -\xi_i \\ \xi_i \geq 0, i = 1 \dots l \end{cases} \quad (\text{II.11})$$

où : les ξ_i sont les termes de pénalisation associés à chaque x_i .

Les multiplicateurs de Lagrange $\alpha_i, i=1, \dots, l$, associés à ce problème dans sa formulation duale, déterminent complètement w et b [38].

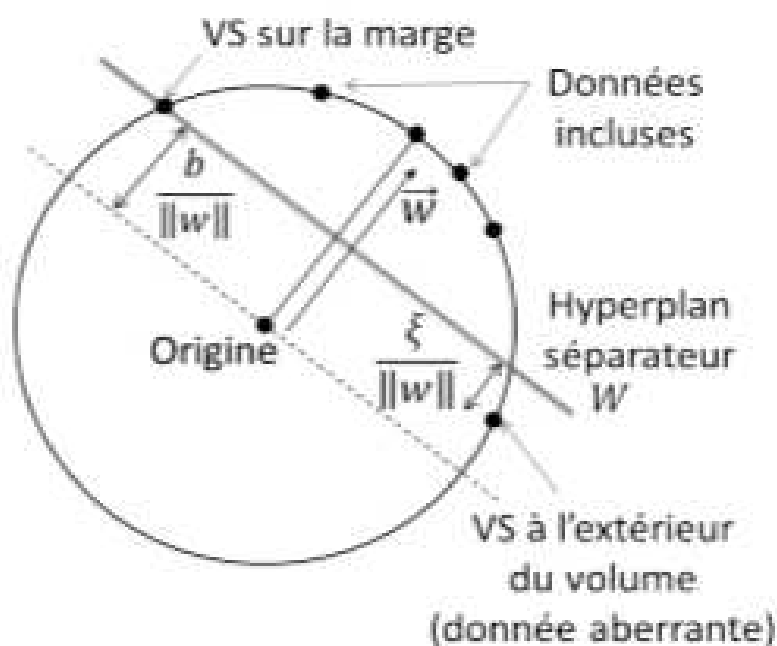


Figure II. 13. Principe des OCSVM.

Les OCSVM créent ainsi un hyperplan caractérisé par w , b qui a une distance maximale de l'origine dans l'espace des caractéristiques, séparant ainsi tous les points de données de l'origine.

II.3.5. Comparaison SVM avec OCSVM

La principale différence entre le SVM à classe unique et le SVM d'origine est que les seules informations fournies par le SVM à classe unique sont les mêmes échantillons normaux à classe unique (également appelés échantillons positifs ou signaux sans interférence), des informations sur les échantillons normaux et anormaux (également appelés échantillons négatifs ou interférences). Essentiellement, un SVM à une classe estime une région englobante contenant la plupart des échantillons d'apprentissage. Si un nouveau candidat est dans cette limite, il est classé dans la classe normale (sans interférence), sinon il est reconnu comme aberrant (avec interférence) [39].

II.4. Optimisation des paramètres des algorithmes d'apprentissage automatique

Il existe plusieurs techniques dans la littérature scientifique pour optimiser les paramètres des algorithmes de classification (MLP, RBF, SVM, OCSVM). Validation simple, Monte-Carlo, validation Croisé, *k-fold* cross-validation. Dans notre travail nous intéressent à la méthode de validation simple et une classification binaire:

II.4.1. Validation simple

La technique de validation simple consiste tout simplement à diviser les données disponibles en deux ensembles (les ensembles d'apprentissage et de validation), sans qu'une donnée ne soit commune. Un nombre non négligeable de données est bien entendu nécessaire dans l'ensemble de validation pour estimer correctement le critère de généralisation, donc pour optimiser les paramètres de classifieur (exemple : SVM). Souvent on garde 2/3 des données dans l'ensemble d'apprentissage et en réserver 1/3 pour la validation. Les ensembles d'apprentissage et de validation sont tirés aléatoirement parmi les N données disponibles [40].

II.4.2. Classification binaire (Bi-classe)

La classification binaire est le processus de classification des éléments d'un groupe donné en deux groupes ou classes en prédisant à quel groupe appartient chaque élément en fonction de la règle de classification. Voici des exemples de ce contexte qui nécessitent une décision quant à savoir si un élément a des propriétés qualitatives, des propriétés spécifiques ou des classifications binaires typiques [40]:

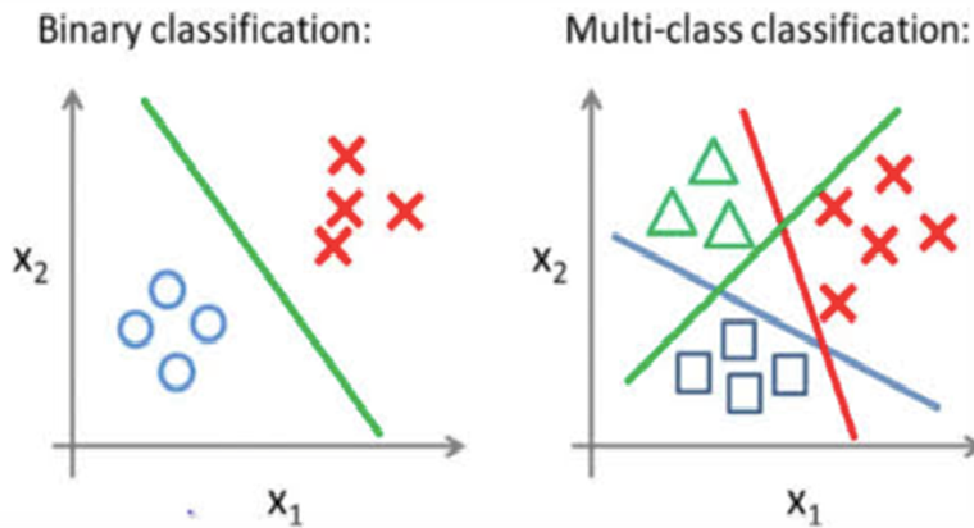


Figure II. 14. Deux cas de la classification (bi-classe et multi-classe).

II.5. Conclusion

Ce chapitre a introduit les techniques d'apprentissage automatique connues sous le nom de méthodes de classification, notamment le MLP (Multi-Layer Perceptron), le RBF (Radial Basis Function), le SVM (Support Vector Machine) et l'OCSVM (One-Class Support Vector Machine). Nous avons présenté ces différentes méthodes de classification dans le contexte de notre problème d'étude, qui est la distinction entre le signal utile et les interférences dans les récepteurs GNSS.

Nous avons discuté des principes de fonctionnement de chaque méthode, ainsi que de leurs avantages et inconvénients potentiels pour notre application spécifique. En particulier, nous avons souligné que l'OCSVM est une variante du SVM traditionnel qui est plus efficace pour détecter les anomalies ou les interférences dans les données des récepteurs GNSS, car il est capable de modéliser une classe unique en utilisant uniquement les échantillons positifs (signal utile), sans nécessiter d'échantillons négatifs (avec interférence).

Dans le prochain chapitre, nous nous intéresserons aux techniques de classification plus avancées, notamment les méthodes d'apprentissage profond, également appelées "*deep learning*". Ces dernières techniques sont utilisées pour améliorer et filtrer les signaux GNSS en exploitant les avantages des réseaux de neurones convolutionnels (CNN) pour apprendre des représentations complexes des données.

Dans le chapitre IV, nous avons testé la méthode OCSVM pour la détection d'anomalies dans les données GNSS, ainsi que les méthodes d'apprentissage profond pour améliorer et filtrer

les signaux. Nous analyserons les résultats obtenus et comparerons les performances de ces différentes approches pour notre problème d'étude de distinction entre le signal utile et les interférences dans les récepteurs GNSS.

Chapitre III

Apprentissage profond

III.1. Introduction

Le Deep Learning, qui associe les connaissances en neurosciences, les avancées technologiques et les mathématiques, est aujourd'hui considéré comme une véritable révolution dans le domaine de l'intelligence artificielle. Cette technologie a déjà permis d'énormes progrès et de multiples applications dans des domaines tels que la reconnaissance faciale et vocale, l'étiquetage d'images, le traitement automatisé du langage et la vision par ordinateur.

L'apprentissage profond est basé sur ce qui a été appelé, par analogie, des «réseaux de neurones artificiels», composés de milliers d'unités (les « neurones ») qui effectuent chacune de petites opérations simples. Les résultats d'une première couche de «neurones» servent d'entrée aux calculs d'une deuxième couche et ainsi de suite. Les progrès de l'apprentissage profond ont été possibles notamment grâce à l'augmentation de la puissance des ordinateurs et au développement de grandes bases de données [41].

Dans ce mémoire, notre objectif est de rechercher des algorithmes d'apprentissage adaptés à ces "architectures profondes" pour distinguer entre le signal utile et le signal avec interférence dans les signaux GNSS. Pour cela, nous avons consacré ce chapitre à présenter un aperçu général des méthodes d'apprentissage profond, de la construction d'une architecture profonde, de leurs caractéristiques et des différentes architectures pré-entraînées telles que LeNet-5, AlexNet, ZFNet, MobileNet, GoogleNet et ResNet. Cela nous permettra d'acquérir une compréhension solide des concepts de base nécessaires à la mise en place de notre propre architecture de deep learning pour notre application de détection d'interférences dans les récepteurs GNSS.

III.2. Définition de l'apprentissage profond (deep learning)

Le Deep Learning est un nouveau domaine de recherche du ML (machine learning), qui a été introduit dans le but de rapprocher le ML (machine learning) de son objectif principal: l'intelligence artificielle (Figure III.1). Il concerne les algorithmes inspirés par la structure et le fonctionnement du cerveau. Ils peuvent apprendre plusieurs niveaux de représentation dans le but de modéliser des relations complexes entre les données.

L'apprentissage profond «deep learning» est un ensemble de techniques d'apprentissage automatique qui a permis des avancées importantes en intelligence artificielle dans les dernières années. Dans l'apprentissage automatique, un programme analyse un ensemble de données afin de tirer des règles qui permettront de tirer des conclusions sur de nouvelles données.

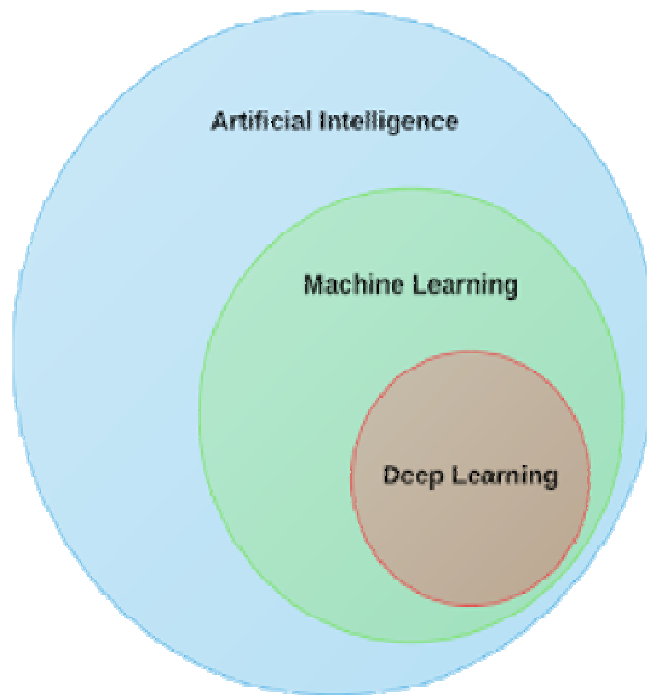


Figure III. 1. La relation entre l'intelligence artificielle, le ML et le deep learning.

La première partie d'une architecture de deep learning est la partie convolutive à proprement parler. Elle fonctionne comme un extracteur de caractéristiques des images (Figure III.2). Une des grandes différences entre le Deep Learning et les algorithmes de ML classique c'est qu'il s'adapte bien, plus la quantité de données fournies est grande plus les performances d'un algorithme de Deep Learning sont meilleurs.

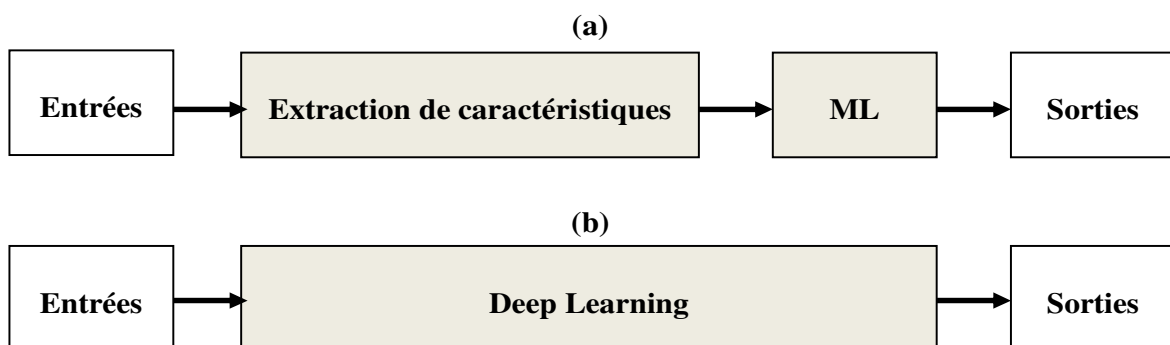


Figure III. 2. Les réseaux de neurones convolutifs (CNN).

Contrairement à plusieurs algorithmes de ML classiques qui possèdent une borne supérieure à la quantité de données qu'ils peuvent recevoir des fois appelée "plateau de performance", il

faut faire des méthodes de sélection des échantillons [42], par contre dans les modèles de Deep Learning n'ont pas de telles limitations.

Autre différence entre les algorithmes de ML traditionnelles et les algorithmes de Deep Learning c'est l'étape de l'extraction de caractéristiques. Dans les algorithmes de ML classique l'extraction de caractéristiques est faite manuellement, c'est une étape difficile et coûteuse en temps et requiert un spécialiste en la matière alors qu'en Deep Learning cette étape est exécutée automatiquement par l'algorithme [43].

III.3. Architectures de réseaux de neurones profonds

Le Deep Learning est un domaine à croissance rapide, et de nouvelles architectures variantes aux algorithmes apparaissent tous les semaines [44]. Dans cette section, nous présenterons un bref aperçu des structures communes que l'on retrouve dans de nombreux réseaux profonds. La figure suivante représente un exemple d'architecture d'un réseau de neurones convolutionnel.

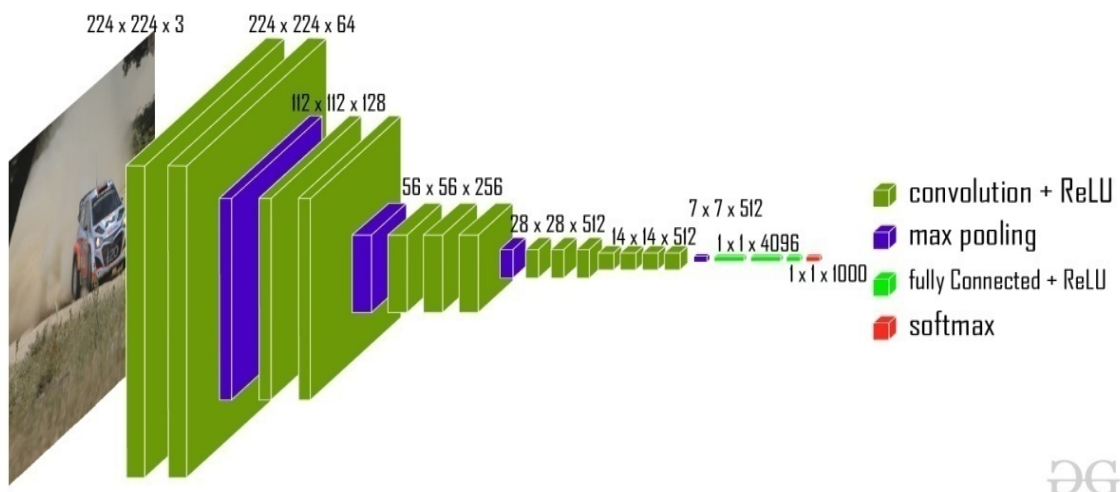


Figure III. 3. Exemple d'architecture CNN.

III.3.1. Principe d'architecture d'un réseau de neurones convolutionnel CNN

Les réseaux de neurones convolutifs sont à ce jour les modèles les plus performants pour classer des images. Désignés par l'acronyme CNN, de l'anglais Convolutional Neural Network, ils comportent deux parties bien distinctes. En entrée, une image est fournie sous la forme d'une matrice de pixels. Elle a deux dimensions pour une image aux niveaux de gris. La couleur est représentée par une troisième dimension, de profondeur 3 pour représenter les couleurs fondamentales [Rouge, Vert, Bleu].

La première partie d'un CNN est la partie convolutive à proprement parler. Elle fonctionne comme un extracteur de caractéristiques des images. Une image est passée à travers d'une

succession de filtres, ou noyaux de convolution, créant de nouvelles images appelées cartes de convolutions (Figure III.4). certains filtres intermédiaires réduisent la résolution de l'image par une opération de maximum local. En fin, les cartes de convolutions sont mises à plat et concaténées en un vecteur de caractéristiques, appelé code CNN [45].

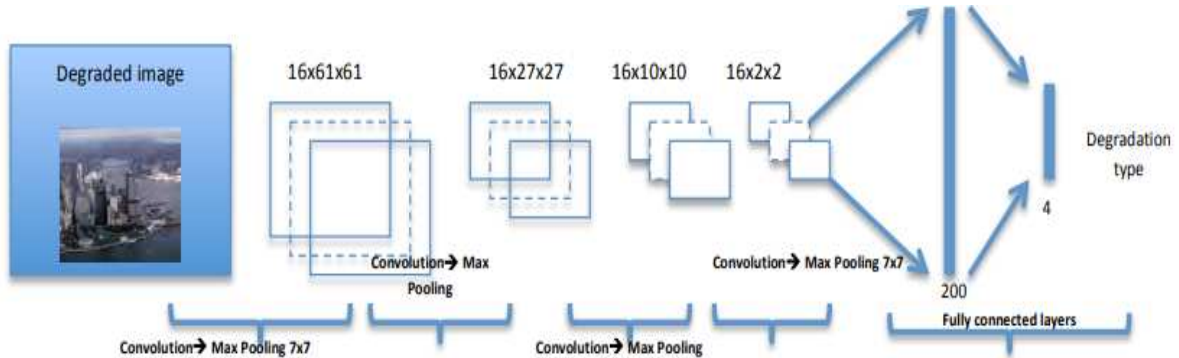


Figure III. 4. Exemple de schéma de principe du CNN.

Ce code CNN en sortie de la partie convolutive est ensuite branché en entrée d'une deuxième partie, constituée de couches entièrement connectées. Le rôle de cette partie est de combiner les caractéristiques du code CNN pour classer l'image. La sortie est une dernière couche comportant un neurone par catégorie. Les valeurs numériques obtenues sont généralement normalisées entre 0 et 1, de somme 1, pour produire une distribution de probabilité sur les catégories.

III.3.2. Blocs de construction

Un CNN est composé de plusieurs couches d'opérations, chacune avec sa propre fonction. Le CNN prend une image en entrée et la transmet à la première couche. Les informations sur les entités sont propagées à travers les couches cachées. Pour chaque couche, les fonctions d'activation effectuent une activation élément par élément à la sortie produite par la couche précédente. La sortie de la couche finale est ensuite comparée à la sortie cible. Ce processus est appelé passage direct (forward pass) [46].

III.3.3. Couche de convolution (CONV)

La couche de convolution est parfois appelée couche d'extraction de caractéristiques, car les caractéristiques de l'image sont extraites dans cette couche. Tout d'abord, une partie de l'image est connectée à la couche Conv pour effectuer une opération de convolution et calculer le produit scalaire entre le champ récepteur (c'est une région locale de l'image d'entrée ayant la même taille que celle du filtre) et le filtre comme le montre la Figure III.5. Le résultat de l'opération est un entier unique du volume de sortie. Ensuite, nous faisons glisser le filtre sur le champ récepteur suivant de la même image d'entrée par une foulée et

refaisons la même opération. Cette opération est répétée par le même processus encore et encore jusqu'à ce que toute l'image soit parcourue [47].

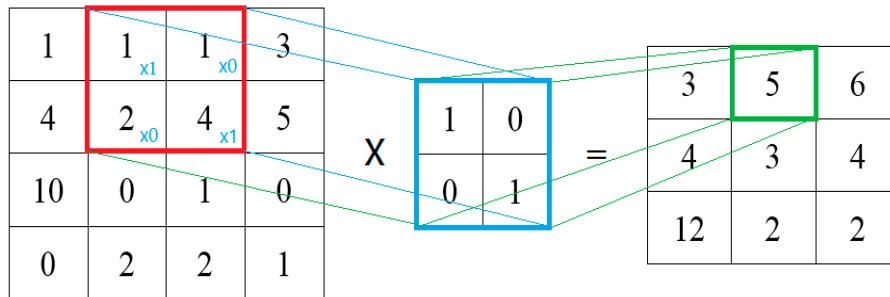


Figure III. 5. Représentation de convolution sur une image de taille «4x4» pixels avec une filtre de convolution.

III.3.4. Couche de pooling (POOL)

Le pooling est le processus de sous-échantillonnage et de réduction de la taille de la matrice. Un filtre est passé sur les résultats de la couche précédente et sélectionne un nombre dans chaque groupe de valeurs (le maximum, c'est ce qu'on appelle le max pooling, et le moyen pour le average pooling).

Tableau III. 1. Max et Moyenne Pooling.

Max Pooling	Moyenne Pooling
Chaque opération de Pooling sélectionne la valeur maximale de la surface.	Chaque opération de Pooling sélectionne la valeur moyenne de la surface.
<ul style="list-style-type: none"> ➤ Garde les caractéristiques détectées ➤ Plus communément utilisé 	<ul style="list-style-type: none"> ➤ Sous échantillonne les caractéristiques

Cela permet au réseau de s'entraîner beaucoup plus rapidement, en se concentrant sur les informations les plus importantes dans chaque caractéristique de l'image et de contrôler également le sur-apprentissage [48].

Elle permet la réduction de la taille de l'image en tenir en compte ses caractéristiques importantes. D'après la Figure III.6, le principe est de couper l'image en cellules régulières, puis nous conservons la valeur maximale dans chaque cellule par rapport au filtre utilisé (2x2 pixels, 3x3...). Donc des petites cellules carrées sont souvent utilisées afin de ne pas perdre trop d'informations.

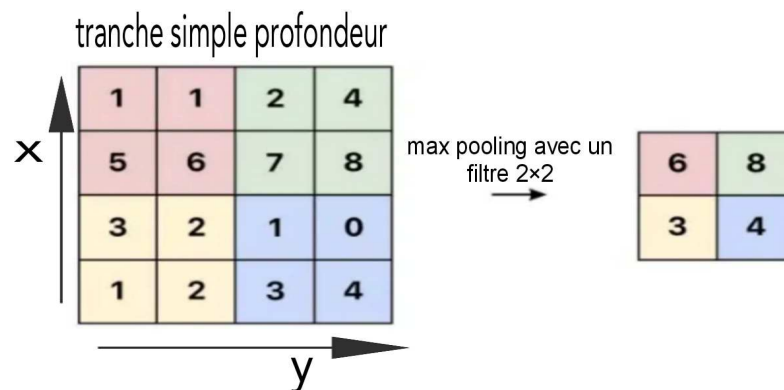


Figure III. 6. Max Pooling avec un filtre 2x2.

III.3.5. Fonctions d'activation

La fonction d'activation est une formule mathématique activée dans certaines circonstances. Comme illustré dans la Figure III.7, lorsque les neurones calculent la somme pondérée des valeurs d'entrée + le biais, elles sont transmises à la fonction d'activation, qui vérifie si la valeur calculée est supérieure au seuil requis. La fonction d'activation est activée et une valeur de sortie est calculée. Cette valeur de sortie est ensuite transmise aux couches suivantes ou précédentes (en fonction de la complexité du réseau), ce qui peut aider les réseaux de neurones à modifier le poids de leurs neurones.

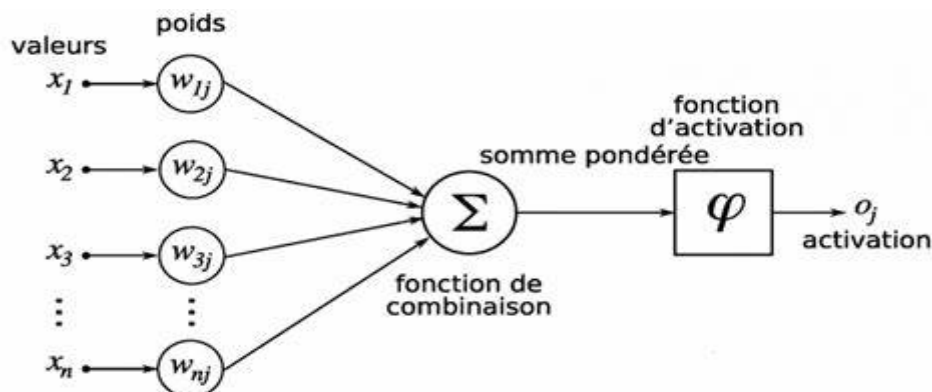


Figure III. 7. Illustration de la fonction d'activation.

Les fonctions d'activation introduisent la non-linéarité dans les réseaux de neurones, nécessaire pour résoudre des problèmes complexes.

Si nous traçons les sorties non linéaires produites par les fonctions d'activation, nous obtiendrons une courbure. La pente de la courbe est utilisée pour calculer le gradient. Et le gradient nous aide à comprendre le taux de changement et les relations entre les variables. À partir des relations, les algorithmes sont optimisés et les poids sont mis à jour [49].

a) Fonction Sigmoidé

La fonction Sigmoidé est une fonction continue. Comme le montre la Figure III.8, elle est utilisée lorsque les valeurs à prédire par le RN sont comprises dans l'intervalle $[0, 1]$. Son but est d'exprimer sa valeur de sortie sous forme d'une probabilité, si la valeur en entrée est un très grand nombre positif, la fonction convertira cette valeur en une probabilité de 1. À l'inverse, si la valeur en entrée est un très petit nombre négatif, la fonction convertira cette valeur en une probabilité de 0.

La fonction Sigmoidé est définie par :

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

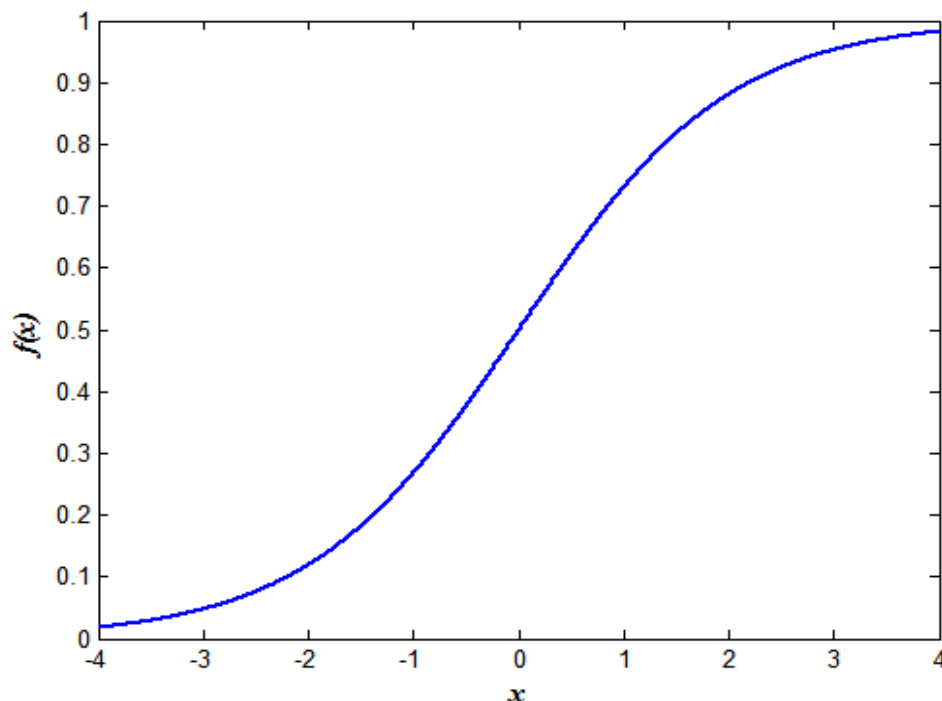


Figure III. 8. Fonction sigmoïde.

b) Fonction Relu / Leaky Relu

La fonction Relu (Unité de Rectification Linéaire), montrée sur la Figure III.9, est créée pour palier au problème de saturation (est la situation où le gradient reste presque nul après chaque période de temps durant le processus d'apprentissage) des deux fonctions (Sigmoides et Tanh).

Relu désigne la fonction réelle non-linéaire interprétée par la formule :

$$f(x) = \max(0, x) \quad (\text{III.2})$$

Si l'entrée est négative alors la sortie est 0, si l'entrée est positive alors la sortie est x. Cette fonction d'activation augmente considérablement la convergence du réseau et ne sature pas.

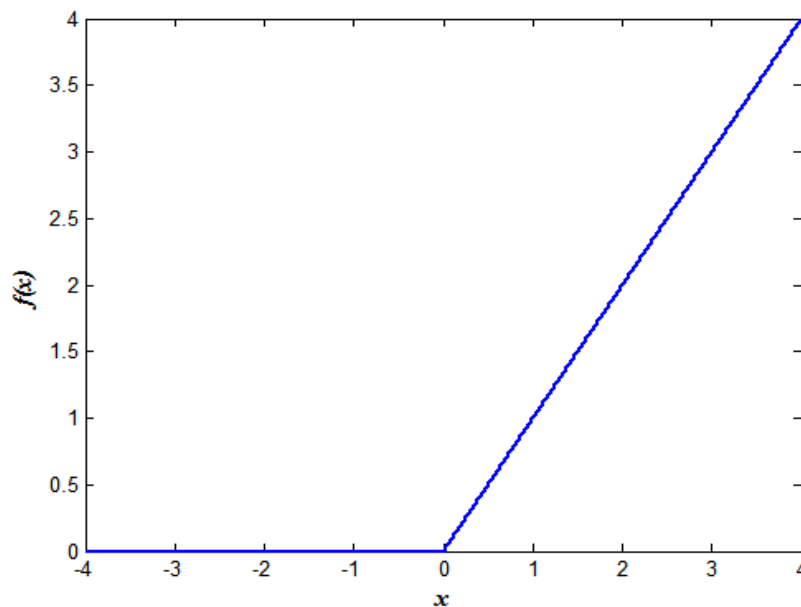


Figure III. 9. La fonction RELU.

L'inconvénient de la fonction Relu est que si la valeur d'entrée est négative, le neurone reste inactif, ainsi les poids ne sont pas mis à jour et le réseau n'apprend pas en raison des gradients de 0 dans la partie négative. Comme montrée la Figure III.10, la fonction Leaky Relu essaye de résoudre le problème de la fonction Relu lorsque l'entrée est négative.

Le concept est lorsque l'entrée est négative, il aura une petite pente négative de $a=0,01$ ou plus. Cette fonction élimine le problème d'inactivité de la fonction Relu pour les valeurs négatives.

La fonction Leaky RELU est donnée par la formule :

$$f(x) = \begin{cases} ax & \text{si } x < 0 \\ x & \text{si } x > 0 \end{cases} \quad (\text{III.3})$$

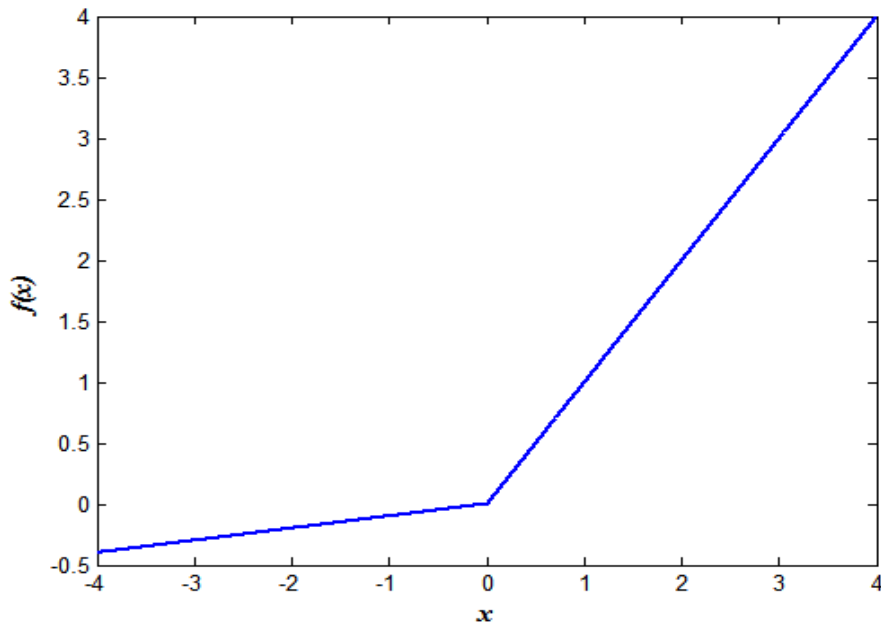


Figure III. 10. La fonction Leaky RELU.

Souvent, la fonction d'activation Relu est préférable, montrée sur la Figure III.11, car il en résulte la formation de réseau neuronal plusieurs fois plus rapide, sans faire une différence significative à la généralisation de précision [45].

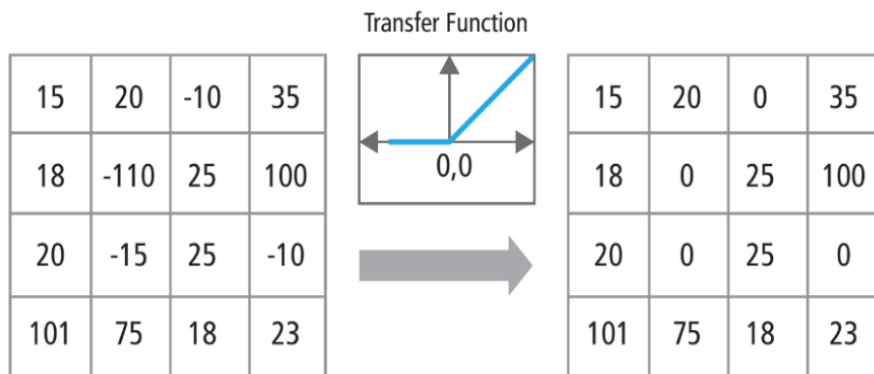


Figure III. 11. Représentation graphique de la fonctionnalité ReLU.

III.3.6. Couche entièrement connectée (FC)

Après plusieurs couches de convolution et de pooling, le raisonnement de haut niveau dans le réseau neuronal se fait via des couches entièrement connectées. Ces dernières sont souvent utilisées en tant que couches finales dans les CNNs. Les couches entièrement connectées combinent mathématiquement une pondération des entités de la couche précédente pour

produire une sortie cible spécifique. Dans une couche entièrement connectée, tous les éléments de toutes les entités de la couche précédente sont connectés à chaque sortie.

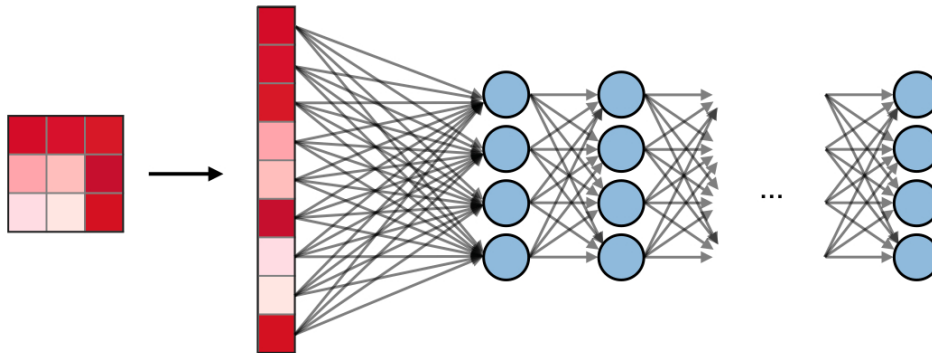


Figure III. 12. Couche entièrement connectée (Fully connected).

III.3.7. Couche de perte (LOSS)

La couche de perte spécifie comment le réseau pénalise l'écart entre la prédiction et la vraie valeur de la cible lors de l'entraînement, elle est la dernière couche du réseau. Différentes fonctions de perte adaptées à différentes tâches peuvent y être utilisées. Par exemple, la fonction de perte «Softmax» est utilisée pour prédire une seule classe parmi K classes mutuellement exclusives, tandis que la fonction de perte par entropie croisée sigmoïde est utilisée pour prédire K valeurs de probabilité indépendantes dans l'intervalle $[0,1]$. La fonction de perte euclidienne est utilisée pour la régression vers des valeurs réelles [50].

La perte Softmax est l'une des fonctions de perte les plus couramment utilisées dans les CNN. Il prend un vecteur N -dimensionnel de valeurs réelles et le transforme en vecteur de valeurs réelles dans un intervalle $[0,1]$ de sorte que la somme soit égale à 1. Étant donné une matrice d'entrée X_n , la perte Softmax illustré dans la Figure III.13, est définie comme suit [46] :

$$f(x_j) = \frac{e^{x_j}}{\sum_{k=1}^k e^{x_j}} \quad (\text{III.4})$$

Et la perte est calculée comme suite:

$$L = \frac{1}{N} \sum_{n=1}^N \log \left(\frac{e^{x_n}}{\sum_{k=1}^k e^{x_j}} \right) \quad (\text{III.5})$$

Où N est la taille du batch et k est le nombre d'identités (c'est-à-dire les étiquettes de classe).

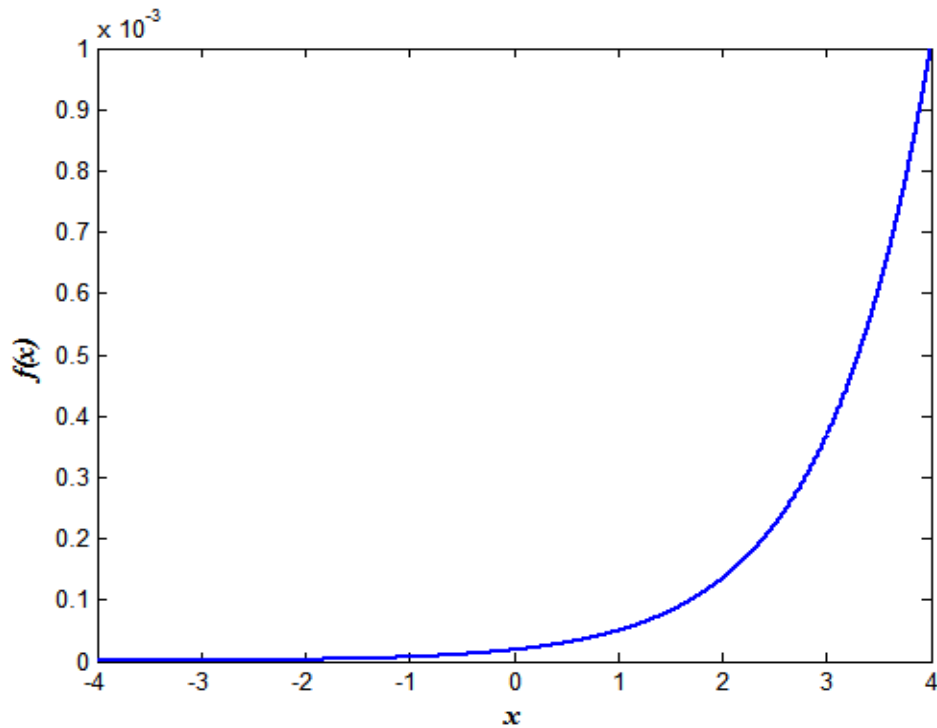


Figure III. 13. La fonction Softmax.

III.3.8. Couche de sortie (output layer)

La couche de sortie contient l'étiquette qui est sous forme codée comme le montre la Figure III.14.

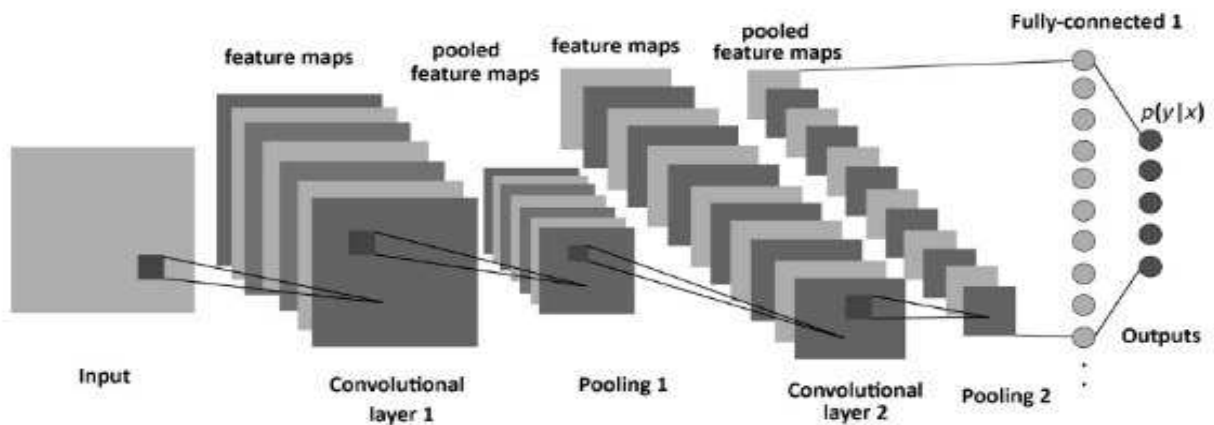


Figure III. 14. Exemple montrant l'étiquette codée de la couche de sortie CNN.

III.3.9. Choix des hyperparamètres

Les hyperparamètres sont les variables qui déterminent la structure du réseau (par exemple : nombre couches cachées, nombre de filtre, forme de filtre, etc...) et les variables qui

déterminent la façon dont le réseau est formé (par exemple : taux d'apprentissage, optimiseur Adam, Batch normalisation, etc....).

- ✚ **Nombre de filtres :** Comme la taille des images intermédiaires diminue avec la profondeur du traitement, les couches proches de l'entrée ont tendance à avoir moins de filtres tandis que les couches plus proches de la sortie peuvent en avoir davantage. Pour égaliser le calcul à chaque couche, le produit du nombre de caractéristiques et le nombre de pixels traités est généralement choisi pour être à peu près constant à travers les couches. Pour préserver l'information en entrée, il faudrait maintenir le nombre de sorties intermédiaires (nombre d'images intermédiaire multiplié par le nombre de positions de pixel) pour être croissante (au sens large) d'une couche à l'autre [51].
- ✚ **Forme de filtres :** Les filtres sont des matrices de poids qui sont appliquées sur les entrées de chaque couche pour extraire des caractéristiques pertinentes. Le nombre d'images intermédiaires contrôle directement la puissance du système, dépend du nombre d'exemples disponibles et la complexité du traitement [45].

III.3.10. Avantages de CNNs

Un avantage majeur des réseaux convolutifs est l'utilisation d'un poids unique associé aux signaux entrant dans tous les neurones d'un même noyau de convolution. Cette méthode réduit l'empreinte mémoire, améliore les performances et permet une invariance du traitement par translation. C'est le principal avantage du CNN par rapport au MLP, qui lui considère chaque neurone indépendant et donc affecte un poids différent à chaque signal entrant.

Lorsque le volume d'entrée varie dans le temps (vidéo ou son), il devient intéressant de rajouter un paramètre de temporisation (delay) dans le paramétrage des neurones.

On parlera Comparés à d'autres algorithmes de classification de l'image, les réseaux de neurones convolutifs utilisent relativement peu de prétraitement. Cela signifie que le réseau est responsable de faire évoluer tout seul ses propres filtres (apprentissage sans supervision), ce qui n'est pas le cas d'autres algorithmes plus traditionnels. L'absence de paramétrage initial et d'intervention humaine est un atout majeur des CNN.

III.4. Classification de l'apprentissage profond

III.4.1. LeNet-5

LeNet-5 est l'une des architectures les plus simples proposées par LeCun et al. [46] en 1998. Comme le montre la Figure III.15, l'architecture LeNet-5 comprend 2 couches convolutionnelles (CONV) et 3 couches fully connected (FC). Cette architecture compte environ 60000 paramètres. Elle a été largement utilisée par plusieurs banques pour la

reconnaissance des nombres manuscrits sur les chèques. Cependant, la capacité de traiter des images à haute résolution nécessite des couches plus grandes et plus convolutives, ce qui peut être limité par la disponibilité des ressources informatiques [46].

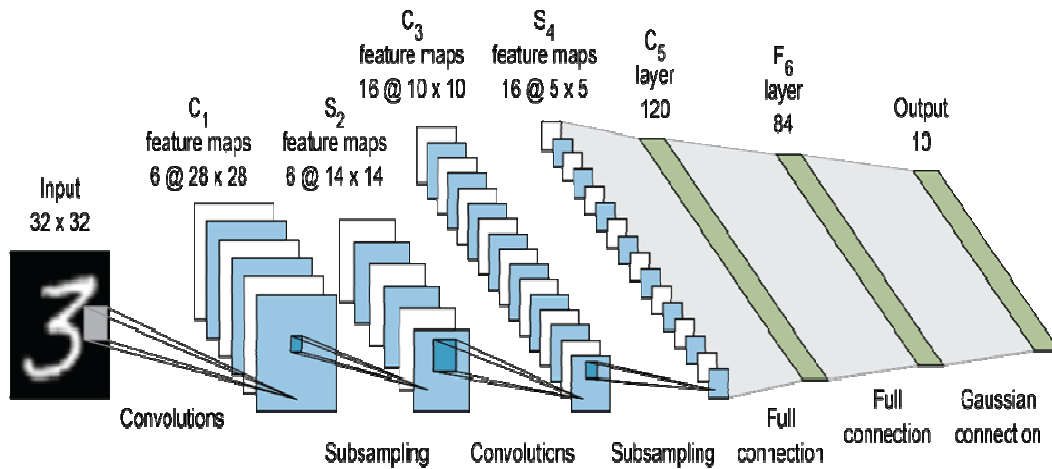


Figure III. 15. Architecture LeNet-5.

III.4.2. AlexNet

AlexNet est le nom d'un réseau neuronal convolutif, initialement écrit avec CUDA pour fonctionner avec le support du GPU, qui a participé au défi de reconnaissance visuelle à grande échelle ImageNet en 2012. AlexNet a largement surperformé tous ses concurrents précédents et a remporté le défi en réduisant l'erreur de top 5 de 26% à 15,3%, il a été conçu par le groupe supervision, composé Alex krizhevsky. AlexNet a eu un grand impact dans le domaine de l'apprentissage automatique, en particulier dans l'application de l'apprentissage en profondeur à la vision artificielle. AlexNet contenait huit couches; les cinq première étaient des couches convolutionnelles et les trois dernières étaient des couches entièrement connectées il a parallèlement deux lignes CNN formées sur deux GPU avec des connexions croisées et Il a utilisé la fonction d'activation non saturante de Relu, qui a montré une amélioration des performances d'entraînement par rapport au Tanh et au sigmoïde. Il a été formé pendant 6 jours simultanément sur deux GPU Nvidia Geforce GTX 580, ce qui explique pour quoi le réseau est divisé en deux pipelines [52]. Comme il est représenté dans les Figures III.16 et III.17.

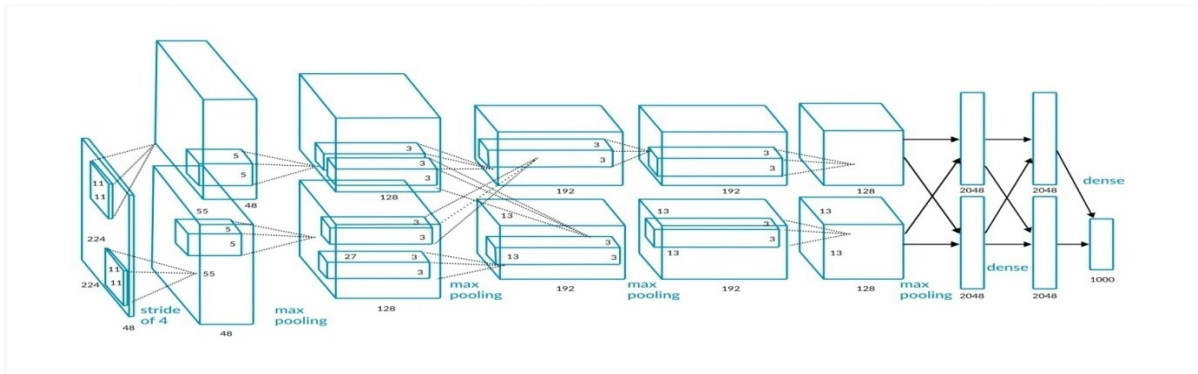


Figure III. 16. Architecture AlexNet.

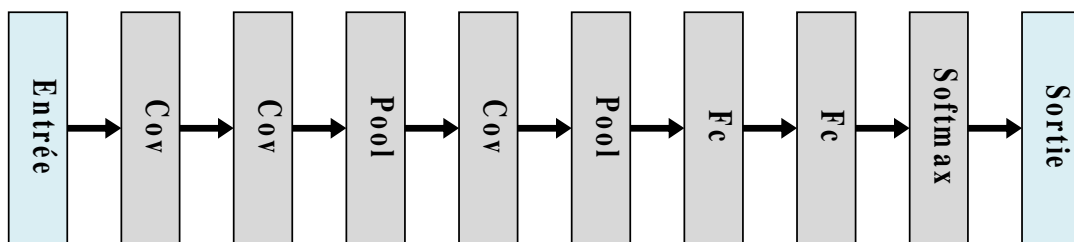


Figure III. 17. Représentation de l'architecture de AlexNet.

III.4.3. ZFNet

Le ZFNet [46] a été proposé par Zeiler et Fergus en 2013. Il n'est pas surprenant qu'il a remporté l'ILSVRC en 2013 avec le top 5 des taux d'erreur de 11,2%. Ce modèle n'a été entraîné que sur 1,3 million d'images, contre 15 millions d'images pour AlexNet. Ils ont également changé la taille du filtre de 11x11 dans AlexNet en 7x7 dans ZFNet. Ce modèle a été entraîné sur le GPU Nvidia Geforce GTX 580 mais pendant 12 jours [46]. La Figure III.18 ci-dessous montre l'architecture de ZFNet.

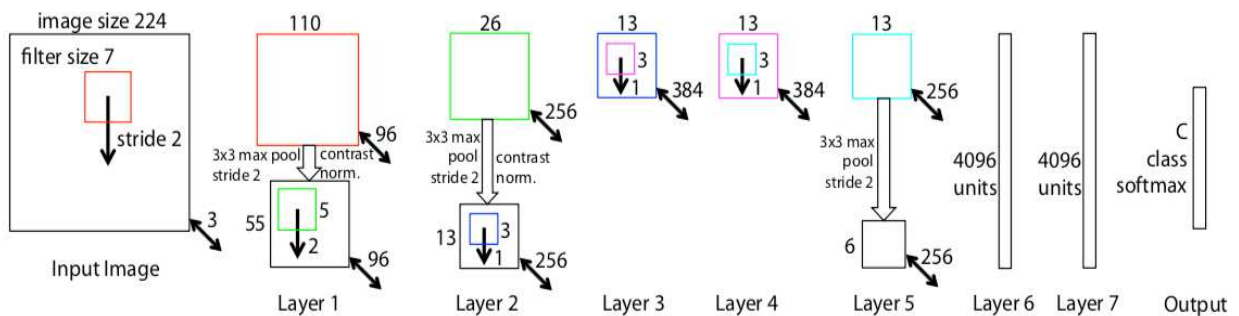


Figure III. 18. Architecture ZF Net.

III.4.4. VGGNet

VGGNet a été proposé en 2014 par Simonyan et al. L'équipe de Visual Geometry Group (VGG) ont inventés le VGG-16 qui a 13 couches convolutionnelles (CONV) et 3 couches Fully Connected (FC). Ils ont remplacés le filtre 11×11 AlexNet et le filtre 7×7 ZFNet par un filtre 3×3 dans leur modèle. Il se compose de 138 millions de paramètres et occupe environ 500 Mo d'espace de stockage. Ils ont également conçu un autre modèle plus profond, VGG-19. Cela représentait 7,3% du taux d'erreur dans le top 5 lors la compétition ILSVRC.

III.4.4.1. VGG16

VGG16 prend en entrée une image en couleurs de taille 224×224 pixels et la classifie dans une des 1000 classes. Il renvoie donc un vecteur de taille 1000, qui contient les probabilités d'appartenance à chacune des classes [53].

Les deux Figures III.19 et III.20 ci-dessous montrent l'architecture de VGG-16 et sa représentation en 3D.

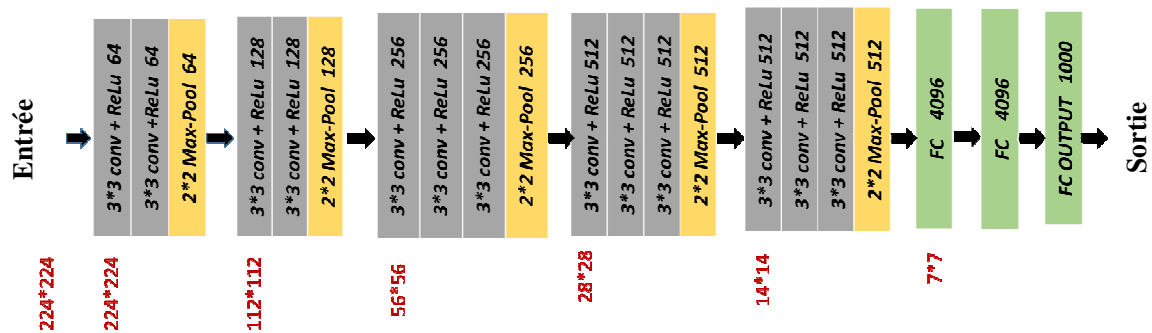


Figure III. 19. Architecture de VGG-16.

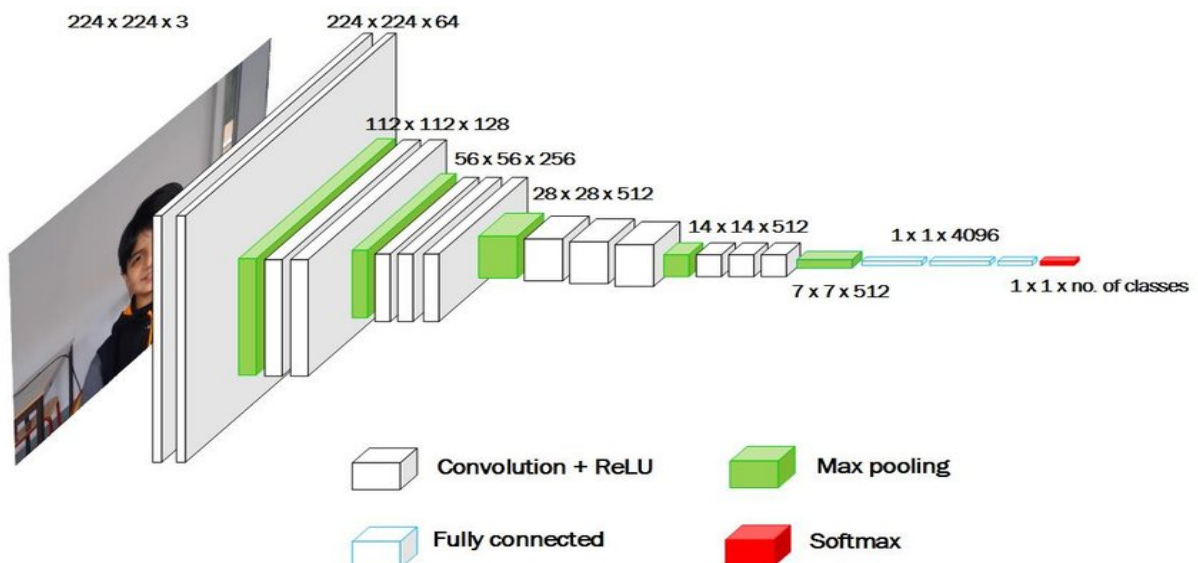


Figure III. 20. Représentation 3D de l'architecture de VGG-16.

Chaque couche de convolution utilise des filtres en couleurs de taille 3×3 pixels, déplacés avec un pas de 1 pixel. Le zero-padding vaut 1 pixel afin que les volumes en entrée aient les mêmes dimensions en sortie. Le nombre de filtres varie selon le "bloc" dans lequel la couche se trouve. De plus, un paramètre de biais est introduit dans le produit de convolution pour chaque filtre.

Chaque couche de convolution a pour fonction d'activation une ReLU. Autrement dit, il y a toujours une couche de correction ReLU après une couche de convolution. L'opération de Pooling est réalisée avec des cellules de taille 2×2 pixels et un pas de 2 pixels, les cellules ne se chevauchent donc pas.

Les deux premières couches fully-connected calculent chacune un vecteur de taille 4096, et sont chacune suivies d'une couche ReLU. La dernière renvoie le vecteur de probabilités de taille 1000 (le nombre de classes) en appliquant la fonction softmax. De plus, ces trois couches utilisent un paramètre de biais pour chaque élément du vecteur en sortie.

III.4.4.2. VGG19

VGG-19 est un réseau de neurones convolutifs de 19 couches de profondeur. Il est possible de charger une version pré-entraînée du réseau entraîné sur plus d'un million d'images de la base de données ImageNet.

Tableau III. 2. Configuration réelle de réseau.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
Input (224 × 224 RGB image)					
Conv 3-64	Conv 3-64 LRN	Conv 3-64 Conv 3-64	Conv 3-64 Conv 3-64	Conv 3-64 Conv 3-64	Conv 3-64 Conv 3-64
Maxpool					
Conv 3-128	Conv 3-128	Conv 3-128 Conv 3-128	Conv 3-128 Conv 3-128	Conv 3-128 Conv 3-128	Conv 3-128 Conv 3-128
Maxpool					
Conv 3-256	Conv 3-256	Conv 3-256	Conv 3-256	Conv 3-256	Conv 3-256

Conv 3-256	Conv 3-256	Conv 3-256	Conv 3-256 Conv 1-256	Conv 3-256 Conv 3-256	Conv 3-256 Conv 3-256 Conv 3-256
Maxpool					
Conv 3-512 Conv 3-512	Conv 3-512 Conv 3-512	Conv 3-512 Conv 3-512	Conv 3-512 Conv 3-512 Conv 1-512	Conv 3-512 Conv 3-512 Conv 3-512	Conv 3-512 Conv 3-512 Conv 3-512 Conv 3-512
Maxpool					
Conv 3-512 Conv 3-512	Conv 3-512 Conv 3-512	Conv 3-512 Conv 3-512	Conv 3-512 Conv 3-512 Conv 1-512	Conv 3-512 Conv 3-512 Conv 3-512	Conv 3-512 Conv 3-512 Conv 3-512 Conv 3-512
Maxpool					
FC-4096					
FC-4096					
FC-1000					
Soft-max					

Le réseau pré-entraîné peut classer les images en 1000 catégories d'objets, telles que le clavier, la souris, le crayon et de nombreux animaux. En conséquence, le réseau a appris de riches représentations de caractéristiques pour un large éventail d'images. Le réseau a une taille d'entrée d'image de 224 par 224 [54].

III.4.4.3. MobileNet-V2

Une architecture qui cherche à bien fonctionner sur les appareils mobiles. Il est basé sur une structure résiduelle inversée où les connexions résiduelles sont entre les couches de bottleneck. La couche d'expansion intermédiaire utilise des convolutions légères en profondeur pour filtrer les entités en tant que source de non-linéarité. Dans son ensemble, l'architecture de MobileNetV2 contient la couche initiale entièrement à convolution avec 32 filtres, suivie de 19 couches de bottleneck résiduelles [46]. Ci-après la Figure III.21 montre l'architecture complète du modèle MobileNet-V2.

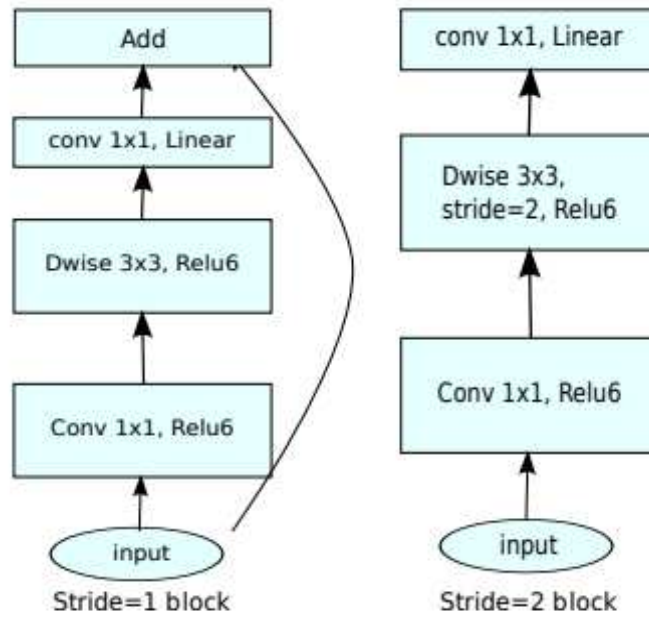


Figure III. 21. Architecture MobileNet-V2.

III.4.4.4. GoogLeNet/Inception

GoogLeNet est la première architecture qui utilise des multi-connexions entre les couches dans un module appelé «Inception» et élimine les couches entièrement connectées et les remplacer une couche de Softmax. GoogLeNet contient 9 modules «Inception» et propose plusieurs sortie de softmax [55].

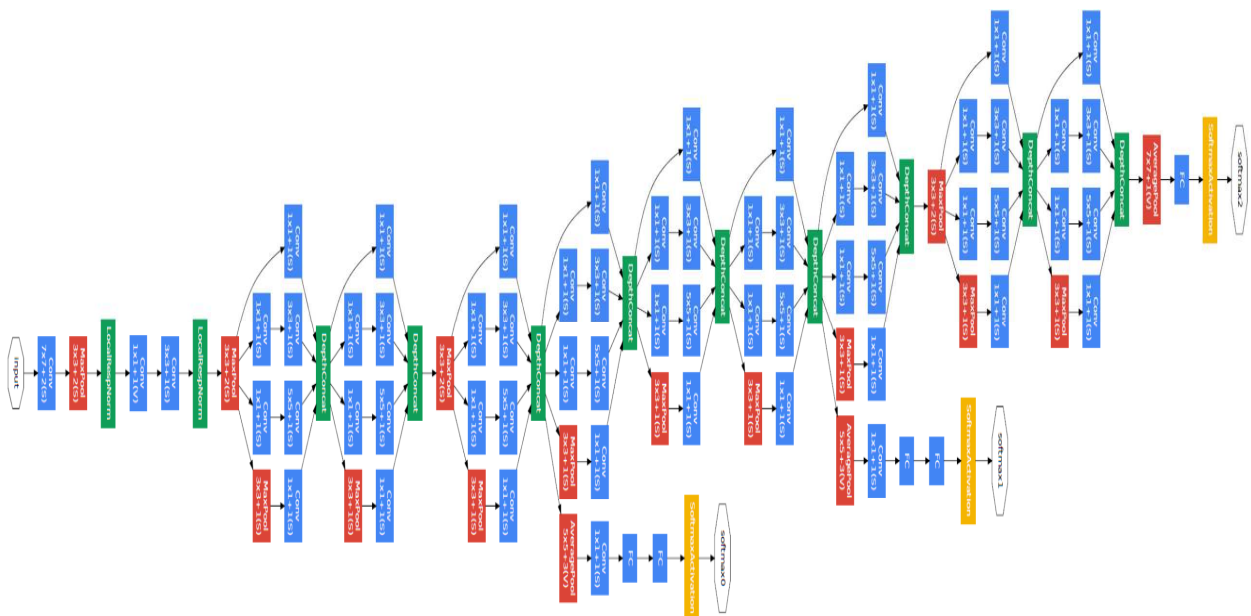


Figure III. 22. Architecture GoogLeNet/Inception-v1.

Autre modèle proposé en 2015 «Inception-V2» est un successeur d'Inception-V1, c'était le prototype d'Inception-V3, avec 24M paramètres.

III.4.4.5. ResNet

Le réseau Residual Neural Network (ResNet) a introduit une architecture innovante avec des «sauts de connexions» et une lourde normalisation par lots. Ces connexions sont également appelées unités gated ou unités récurrentes gated et ont une forte similitude avec les éléments récents réussis appliqués dans les RNN. Grâce à cette technique, ils ont pu former un NN avec 152 couches tout en ayant une complexité inférieure à celle de VGGNet. Il atteint un taux d'erreur de 3,57% parmi les cinq premiers, ce qui est supérieur à la performance au niveau humain sur cet ensemble de données (voir Figure III.23).

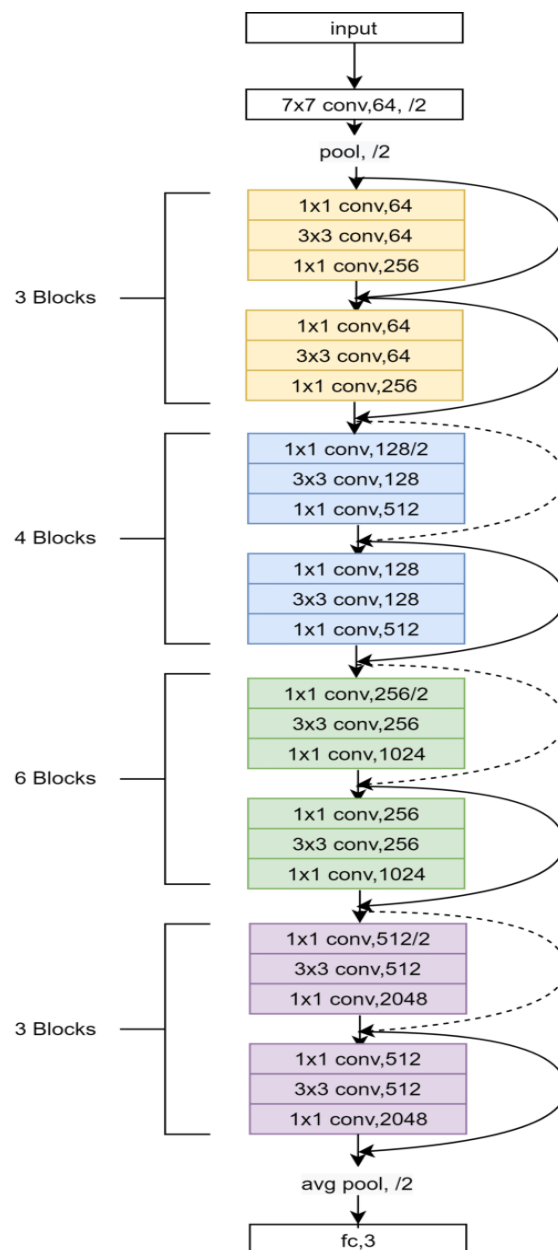


Figure III. 23. Architecture de ResNet.

III.5. Domaines d'application de l'apprentissage profond

Ces techniques se développent dans le domaine de l'informatique appliquée (reconnaissance visuelle, par exemple d'un panneau de signalisation par un robot ou une voiture autonome, et vocale) à la robotique, à la bioinformatique, la reconnaissance ou comparaison de formes, la sécurité, la santé, etc.

L'apprentissage profond peut par exemple permettre à un ordinateur de mieux reconnaître des objets hautement déformables et/ou analyser par exemple les émotions révélées par un visage photographié ou filmé, ou analyser les mouvements et position des doigts d'une main, ce qui peut être utile pour traduire le langage des signes, améliorer le positionnement automatique d'une caméra, etc.

Elles sont utilisées pour certaines formes d'aide au diagnostic médical (ex. : reconnaissance automatique d'un cancer en imagerie médicale), ou de prospective ou de prédiction (ex. : prédiction des propriétés d'un sol filmé par un robot [45]).

Une autre application très intéressante est l'utilisation de méthodes d'apprentissage profond dans le domaine des signaux GNSS.

En effet, les signaux GNSS sont souvent affectés par des interférences, telles que celles émises par des sources artificielles ou naturelles, qui peuvent perturber la qualité du signal et altérer sa précision. L'utilisation de méthodes traditionnelles de filtrage ou de correction de signaux peut être insuffisante pour lutter contre ces interférences. C'est là que les méthodes d'apprentissage profond peuvent offrir une alternative intéressante en permettant une distinction plus précise entre le signal utile et le signal avec interférence. Les réseaux de neurones convolutionnels CNN peuvent être utilisés pour extraire des caractéristiques importantes des signaux GNSS, tandis que les réseaux de neurones entièrement connectés peuvent être utilisés pour effectuer la classification finale des signaux en signal utile ou signal interférence. L'application de ces méthodes d'apprentissage profond pourrait améliorer considérablement la qualité et la précision des signaux GNSS, et ainsi contribuer à des avancées significatives dans de nombreux domaines tels que la navigation, la cartographie, la surveillance environnementale, etc.

III.6. Conclusion

Dans ce chapitre, nous avons présenté des notions importantes en relation avec l'apprentissage profond telles que la définition, les architectures, les avantages des CNN, etc. Nous avons également donné une vision générale de l'apprentissage profond, en détaillant la méthode que nous avons choisie pour notre travail qui est le CNN.

L'application de techniques d'apprentissage profond dans le domaine des signaux GNSS est une avancée récente et prometteuse. Les réseaux de neurones convolutionnels ont été utilisés pour résoudre des problèmes de classification de signaux GNSS, tels que la détection des interférences. Dans la littérature scientifique, ces réseaux ont démontré leur efficacité et leur précision par rapport aux méthodes de classification traditionnelles. En outre, la possibilité d'entraîner des réseaux de neurones convolutionnels à partir de grandes quantités de données permet d'améliorer continuellement la précision de ces méthodes de classification. Cela ouvre la voie à de nouvelles applications potentielles pour les signaux GNSS, telles que la détection de brouillage ou l'identification de signaux de systèmes de navigation alternatifs.

Dans le chapitre suivant, nous avons utilisé l'architecture CNN pour identifier les interférences dans les récepteurs GNSS, et nous avons comparé ces architectures par rapport à une méthode traditionnelle, l'OCSVM (One-Class Support Vector Machine).

Chapitre IV

Résultats et discussion

IV.1. Introduction

Ces dernières années, grâce aux applications réussies de l'apprentissage profond (DL) dans de nombreux domaines tels que la reconnaissance d'image [56], l'analyse de la parole [57] et le traitement du langage naturel (NLP : Natural Language Processing) [58], la méthode de détection d'interférences dans les récepteurs GNSS basée sur le DL est également devenue un sujet de recherche important.

Notre objectif est d'identifier et de supprimer les interférences dans les récepteurs des systèmes de navigation par satellite GNSS, d'où la nécessité d'un système de détection ou de classification automatique conçu pour détecter et analyser les signaux GNSS. Pour cela, nous avons proposé d'utiliser des architectures neuronales CNN.

Dans ce dernier chapitre, nous avons proposé trois architectures de classification CNN adaptées aux signaux GNSS. Ensuite, nous avons présenté les résultats de la simulation après avoir sélectionné la méthode optimale pour le modèle CNN et nous avons démontré l'efficacité de la méthode proposée à l'aide de plusieurs exemples. Enfin, pour valider notre méthode, nous avons comparé notre approche proposée à un classificateur robuste contre les valeurs aberrantes OCSVM.

IV.2. Description de la base de données GNSS

Les performances des techniques d'atténuation des interférences ont été obtenues à l'aide d'un simulateur 'GE5-TUT' open-source dans le MATLAB. Ce simulateur du système Galileo est dédié à la bande E5. Le simulateur permet la simulation de la transmission de données, qui est composée de trois blocs essentiels qui sont: émetteur, le canal de propagation et le bloc du récepteur. Par conséquent, l'algorithme d'atténuation est simulé en tant qu'un bloc défini et intégré dans le simulateur [59].

Les signaux Galileo E5 font partie de la bande E5 ([1164-1215 MHz] qui est le plus grand système de radionavigation par satellite (RNSS) bande. C'est aussi une bande ARNS, donc protégée par UIT, mais sans exclusivité pour le RNSS. Cela signifie que tout Le système de radiodiffusion à l'intérieur de cette bande devra faire face à la les services existants non RNSS déjà présents dans cette bande. Dans en particulier, les systèmes utilisant des signaux pulsés forts, tels que Appareils de mesure de distance (DME), TACTical Air Navigation (TACAN), Distribution d'information tactique interarmées Systèmes (JTIDS)/ Distribution d'informations multifonctionnelles, les systèmes (MIDS) sont déployés dans cette bande [60]. Les détails concernant les paramètres du signal Galileo sont illustré dans le tableau IV.1

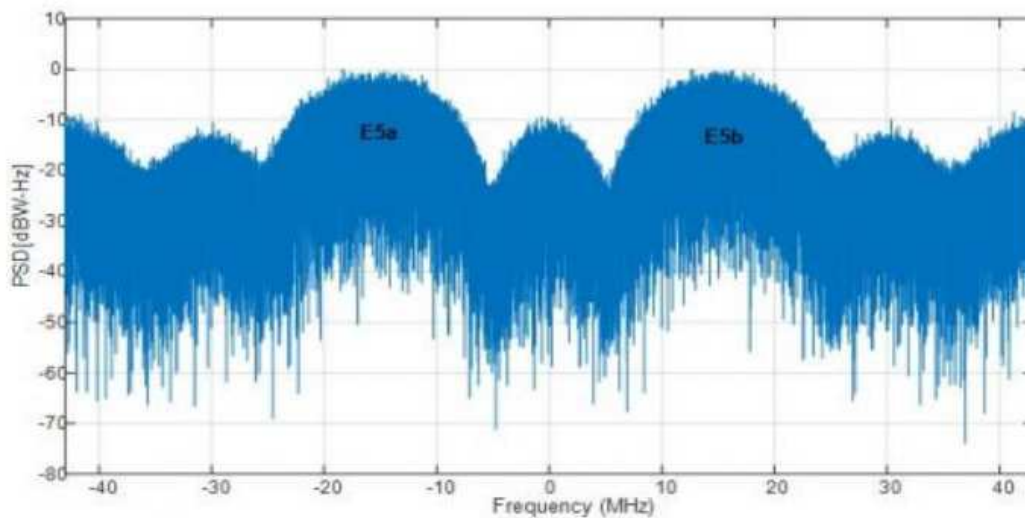
Tableau IV. 1. Paramètres du signal E5aI.

Paramètres	valeur
Signal désiré	Galileo E5a-I
Fréquence d'échantillonnage (Fe)	31.500 MHz
Fréquence intermédiaire (Fi)	4.655 MHz
Intégration cohérente	1 ms
CNR 49 dB-Hz	CNR 49 dB-Hz

Le test de simulation est composé du signal Galileo E5aI et du signal d'interférence à ondes continues ajouté. Le rapport d'interférence au signal ISR du SCWI (l'interférence à onde continue unique) et du MCWI (l'interférence à onde continue multiple) et DME (l'interférence pulsé) est compris entre 10 à 60 dB. Il est a noté que le rapport est décrit par l'équation suivante:

$$ISR = 10 \times \log\left(\frac{I}{S}\right) \quad (IV.1)$$

où : I est la puissance d'interférence et S est la puissance du signal GNSS.

**Figure IV. 1.** Présentation d'un signal E5 avec les deux composantes E5a et E5b.

Notez que le CWI utilisé est un signal sinusoïdal pur dans le cas de SCWI, situé au centre du lobe principal à la fréquence intermédiaire FI=4,655 MHz de E5a bande qui correspond à une fréquence porteuse de 1176,45 Mhz. Ce scénario présente l'attaque la plus dangereuse sur le système, et l'interférence MCWI simulée est la somme de plusieurs signaux sinusoïdes avec

des fréquences différentes répartis sur la bande E5aI, avec celle de la fréquence intermédiaire $FI=4,655\text{MHz}$ [3].

Pour plus d'illustration, des exemples de l'amplitude du signal E5aI dans le domaine temporel sans interférence et avec interférence, sont montrés dans les Figures IV.2 et IV.3. Dans le cas de signal contaminé par une interférence DME, le rapport interférence sur signal ISR est varié entre 10dB à 60dB avec un pas de 10dB. La représentation des signaux contaminé et non contaminé dans le domaine fréquentiel sont illustrés dans les Figure IV.4 et IV.5.

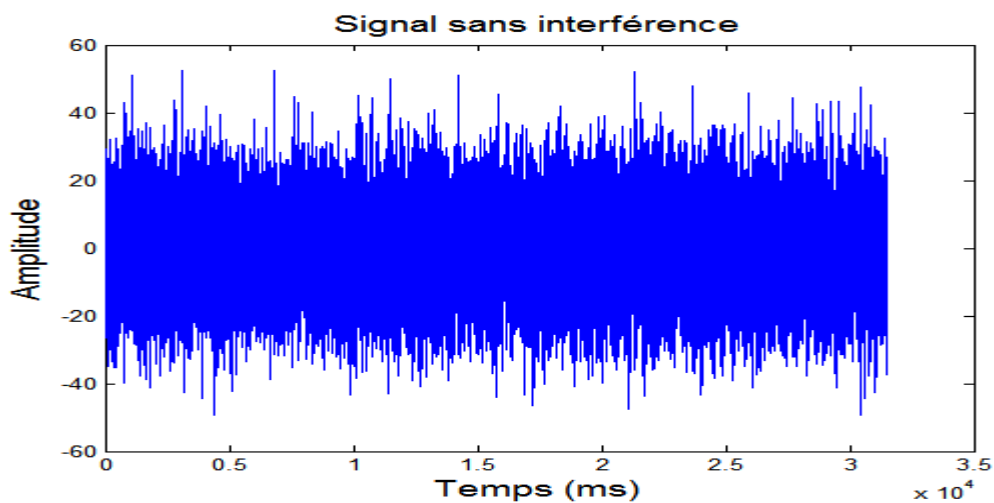
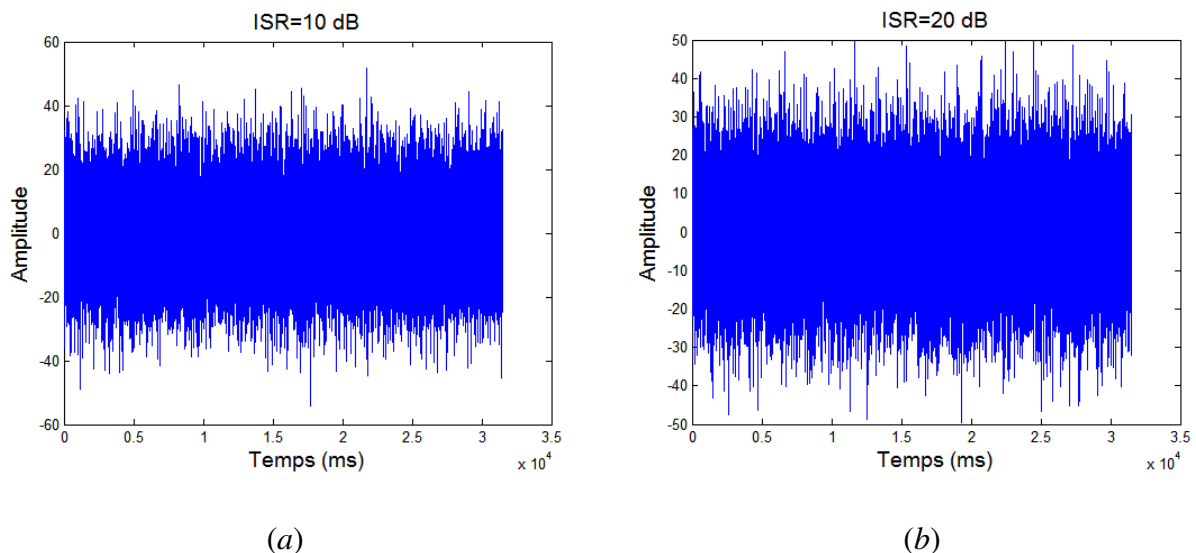


Figure IV. 2. Amplitude du signal E5aI dans le domaine temporel sans interférence.



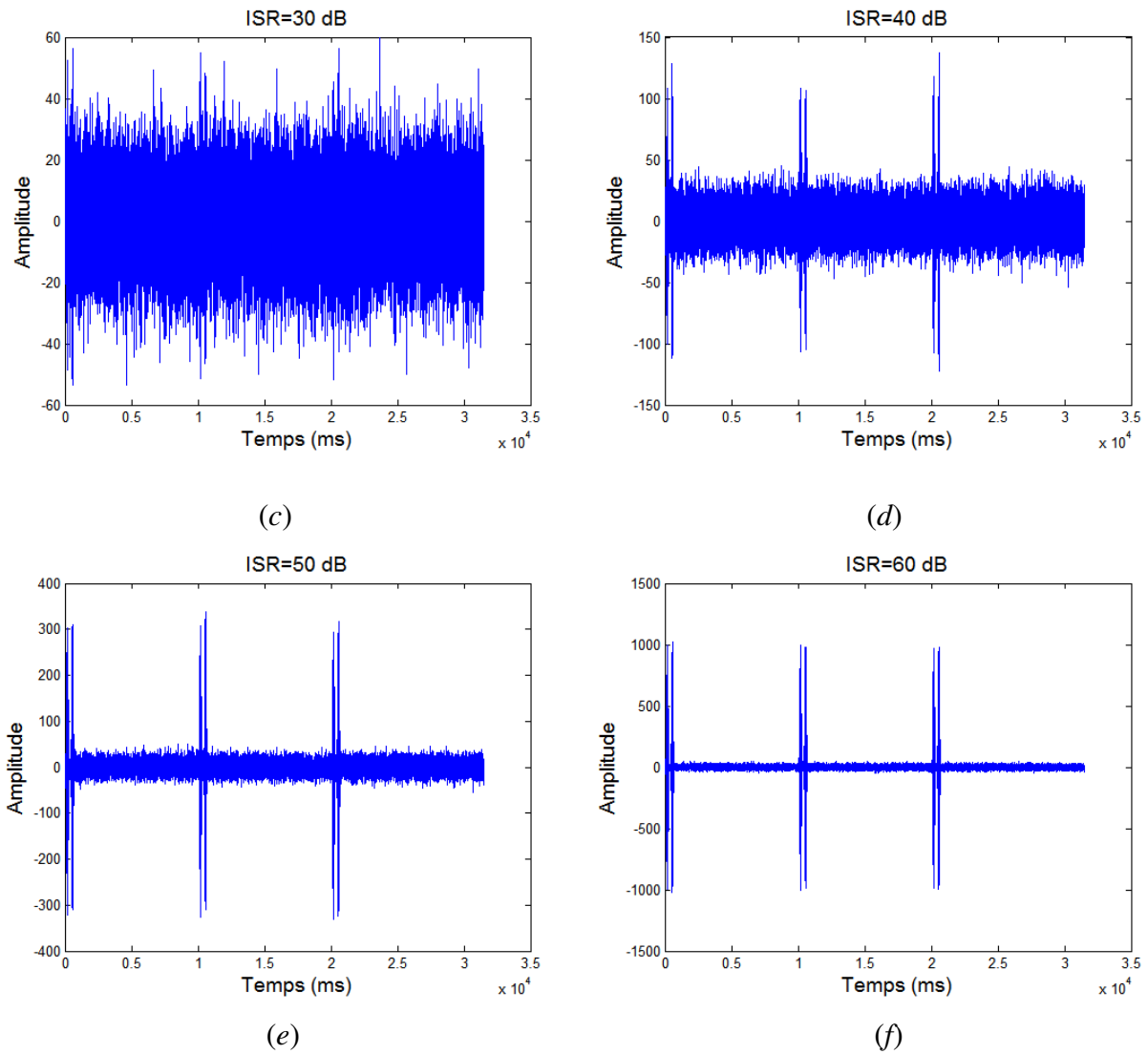


Figure IV. 3. Amplitude du signal E5aI dans le domaine temporel contaminé par une interférence DMA, ISR varie entre 10dB à 60dB.

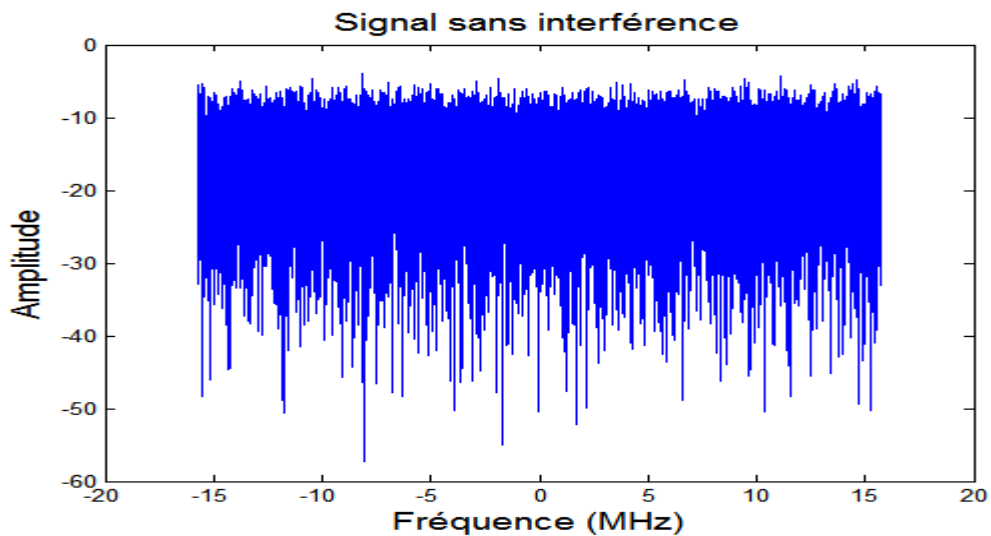


Figure IV. 4. Amplitude du signal E5aI dans le domaine fréquentiel sans interférence.

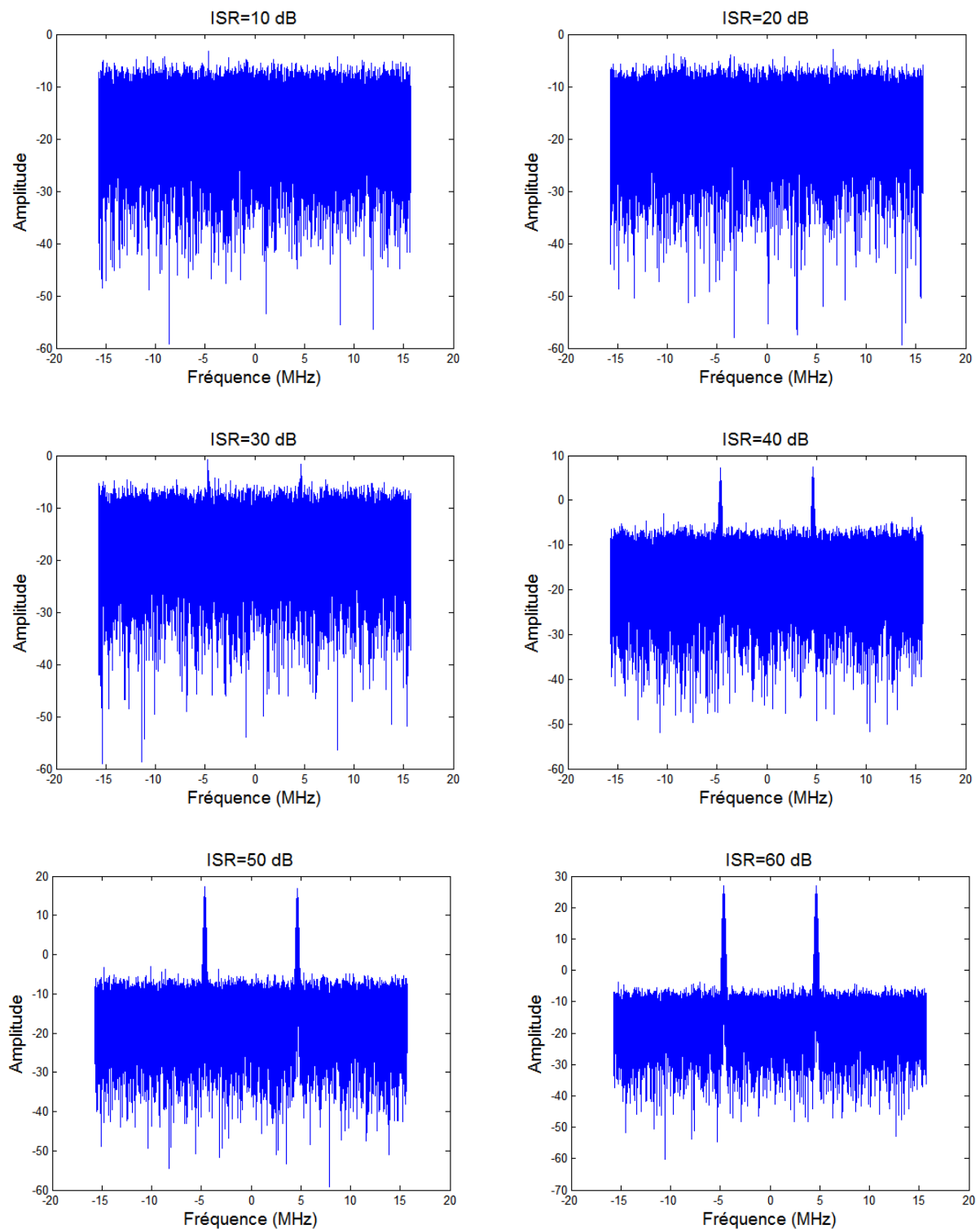


Figure IV. 5. Amplitude du signal E5aI dans le domaine fréquentiel contaminé par des interférences DMA, ISR varie entre 10dB à 60dB.

IV.3. Méthode proposée

Ce travail s'inspire des différentes recherches menées dans la littérature scientifique pour proposer une méthode basée sur l'architecture de réseaux de neurones convolutifs (CNN) permettant l'identification et la suppression intelligentes des interférences dans les récepteurs GNSS. Cette méthode se compose principalement de deux étapes comme le montre la Figure IV.6:

- 1) La création de bases de données d'interférences extraites directement à partir des données brutes des signaux GNSS à l'aide d'une méthode simple de transformation signal-image;
- 2) Le développement d'un nouveau modèle d'identification (ou de suppression) intelligent avec l'architecture CNN permettant de reconnaître automatiquement les interférences.

Dans les sous-sections suivantes, nous détaillerons les différentes étapes de notre méthode proposée.

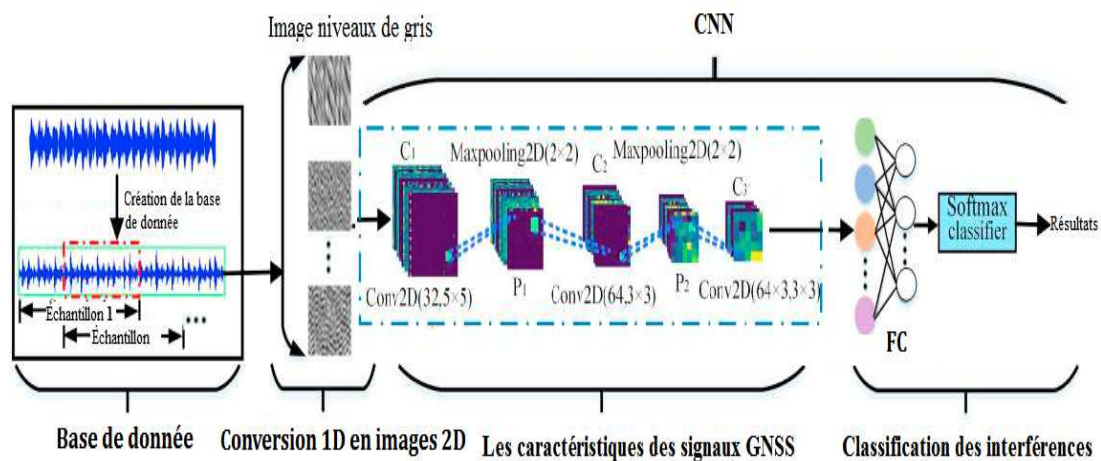


Figure IV. 6. Classification des interférences des signaux GNSS en utilisant l'architecture CNN.

IV.3.1. Conversion des signaux 1D en images 2D

Afin d'entraîner le modèle CNN sur les caractéristiques abstraites des différents signaux GNSS dans le domaine temporel, nous utilisons une méthode de prétraitement des données qui permet de convertir directement les signaux 1D GNSS en images 2D [61]. Cette méthode est ensuite utilisée pour la détection des interférences dans les récepteurs GNSS.

Le principe de la conversion d'un signal 1D en une image 2D en niveaux de gris est illustré dans la Figure IV.7. Pour un signal GNSS dans le domaine temporel, d'une longueur de N^2 , il peut être directement transformé en une image 2D en niveaux de gris de taille $N \times N$ grâce à la

méthode présentée dans la Figure IV.7. Les points d'échantillonnage du signal 1D dans le domaine temporel sont mappés sur la valeur de pixel correspondante dans l'image 2D en niveaux de gris, et chaque intensité de pixel de l'image est variée de 0 à 255 (nombre de bites égale à 8), c'est-à-dire l'échelle de gris. Par conséquent, la relation de transformation entre le signal 1D et l'image 2D en niveau de gris est exprimée mathématiquement comme suit :

$$p(i, j) = \text{round}\left(\frac{L(i + j \times N) - \text{Min}(L)}{\text{Max}(L) - \text{Min}(L)} \times 255\right) \quad (\text{IV.2})$$

où $L(k)$, $k= 0,1,2,\dots,N^2-1$ représente l'amplitude des d'interférences; $\text{Max}(L)$ et $\text{Min}(L)$ désignent respectivement la valeur maximale et minimale du signal GNSS L ; et $P(i, j)$, $i, j=0,1,2,\dots,N-1$ désigne l'intensité de pixel correspondante de l'image en niveaux de gris 2D.

Cette méthode de transformation de données présente une complexité temporelle réduite par rapport à certaines méthodes d'analyse temps-fréquence, telles que la transformée en ondelettes (WT), la STFT et la transformée S.

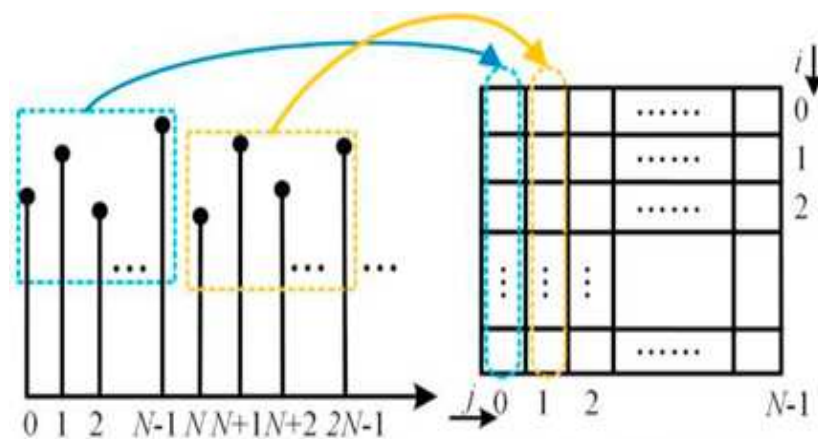


Figure IV. 7. Principe de conversion des signaux en une image au niveau de gris.

VI.3.2. Réseaux de neurones convolutifs

CNN est un type de réseaux de neurones qui utilise des opérations de convolution pour extraire des caractéristiques de haut niveau à partir des séquences ou d'images, et qui peut donner de bonnes performances de généralisation. Depuis que Lecun [62] a proposé la structure du réseau LeNet-5, les CNN ont fait de grands progrès dans le domaine de traitement de signal et d'image.

La Figure IV.8 montre la structure typique d'un modèle CNN, qui comprend une couche d'entrée, une couche de convolution, une couche de *Pooling*, une couche entièrement connectée FC et une couche de sortie.

Dans ce mémoire, le signal GNSS du domaine temporel transformé en image est traité par des opérations de convolution et de *Pooling*, pour obtenir la sortie correspondante du signal utile ou interférence. Le processus d'apprentissage du CNN comprend principalement la rétropropagation pour la mise à jour des paramètres.

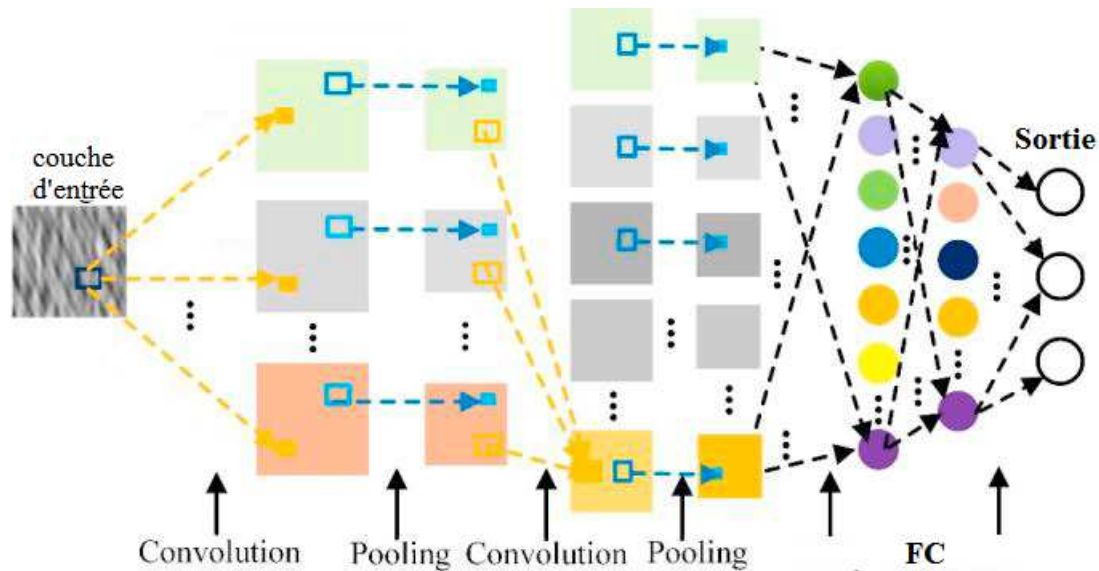


Figure IV. 8. Structure du modèle CNN.

a) Processus de propagation vers l'avant : Le processus de propagation vers l'avant comprend principalement la convolution, sous-échantillonnage (*Pooling*) et la couche entièrement connectée FC. Dans ce processus, la convolution et *Pooling* sont généralement alternées pour réaliser l'apprentissage des fonctionnalités et des représentations.

La couche convolutif (C) est une technique de base du modèle CNN. Les noyaux de convolution de tailles différentes et de formes différentes peuvent être utilisés pour extraire différentes caractéristiques locales abstraites de l'image d'entrée. Si x représente les entrées, l'opération de convolution est exprimée comme suit :

$$x = f\left(\sum x * w_{ij} + b\right) \quad (\text{IV.3})$$

où "*" représente l'opération de convolution, w_{ij} représente le noyau convolutionnel, b désigne le biais additif et $f(\bullet)$ est une fonction d'activation qui est généralement ReLU. La couche de *Pooling* (P) est souvent utilisée après la convolution. Elle est exprimée comme suit :

$$x = f(\text{down}(x) * w + b) \quad (\text{IV.4})$$

où w désigne la matrice de poids et b le biais multiplicatif, $down(\bullet)$ désigne la fonction de sous-échantillonnage. La fonction Pooling réduit la taille de l'image (par exemple, la fonction maximale ou la fonction moyenne) et en calculant une seule valeur pour chaque groupe de pixels. Cette opération peut aider à prévenir le surajustement et à augmenter la robustesse du modèle.

Après avoir appliqué les opérations de convolution et de Pooling pour extraire les caractéristiques de l'image d'entrée, la couche entièrement connectée (FC) est utilisée pour cartographier ces caractéristiques distribuées dans l'espace des étiquettes d'échantillons, permettant ainsi la classification des fonctionnalités. FC est exprimé par la relation suivante:

$$h(x) = f(wx + b) \quad (IV.5)$$

où x et $h(x)$ sont respectivement l'entrée et la sortie de la couche entièrement connectée, w est la matrice de poids et b est le biais additif. La fonction $f(\bullet)$ *softmax* est utilisée pour normaliser les scores de sortie en une distribution de probabilité sur les classes de sortie (signal utile, signal avec interférence).

b) Rétropropagation pour la mise à jour des paramètres: dans le processus d'apprentissage des CNN, l'algorithme de rétropropagation (BP) est utilisé pour mettre à jour les poids et les biais de chaque couche du réseau avec la fonction de perte minimale ERR (ou la précision maximale Acc) pour minimiser l'erreur entre la prédiction du modèle et la valeur cible réelle. Dans notre cas nous avons choisi de calculer le taux de classification ou d'identification (Acc), ce critère il est donné par la formule suivante:

$$Acc(\%) = \frac{P}{N} \times 100 \quad (IV.6)$$

où : P est les échantillons bien classés et N est le nombre d'échantillons des classes cibles. Un "bon" classifieur doit présenter un accuracy ou précision élevé (proche de 100%).

L'erreur de classification à minimiser est donnée par la relation suivante :

$$ERR(\%) = 100 - Acc(\%) \quad (IV.7)$$

VI.3.3. Génération des images aux niveaux de gris

Les signaux GNSS traités sont divisés en données d'apprentissage, validation et données de test, chacun ayant une longueur de données de 1000 échantillons. Ils sont ensuite convertis en images aux niveaux de gris 2D de plusieurs tailles telles que 28×28, 48×48, et 64×64 à l'aide de l'équation (V.2).

Les Figures IV.9 et IV.10 montrent un exemple de la transformation des signaux GNSS en images aux niveaux de gris 2D. À partir de ces figures, nous pouvons clairement distinguer le

comportement des images aux niveaux de gris 2D pour les deux cas de classes avec et sans interférence, ce qui facilite la classification des signaux.

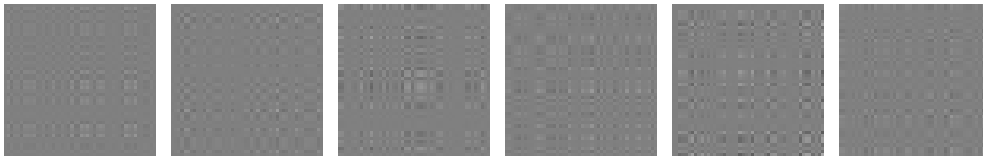


Figure IV. 9. Image en niveau de gris 2D des signaux GNSS sans interférences.

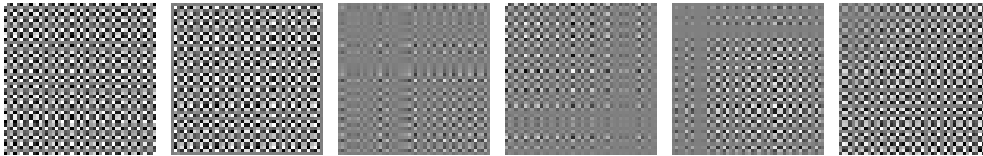


Figure IV. 10. Image en niveau de gris 2D des interférences des signaux GNSS.

IV.4. Résultats des différents modèles CNN

La méthode de suppression d'impulsions dans le domaine temporel dans la partie numérique de la chaîne radiofréquence du récepteur GNSS qui a illustré dans la Figure IV.11, est la méthode la plus courante pour la contre-mesure de l'interférence pulsée.

Ce type de circuit numérique fournit une excision d'interférence pulsée par le moyen d'une opération de seuillage sur les échantillons à la sortie du convertisseur ADC.

Chaque échantillon est comparé à un niveau de seuil numérique, qui est établi en fonction d'une estimation de la puissance de bruit thermique uniquement «les impulsions sont courtes et ont une très grande amplitude par rapport au niveau de bruit» [7]. Dans notre travail, la stratégie de suppression et d'identification des interférences est différente. Afin d'éviter le problème de seuillage, nous avons proposée une méthode de suppression automatique des interférences basée sur trois architectures CNN.

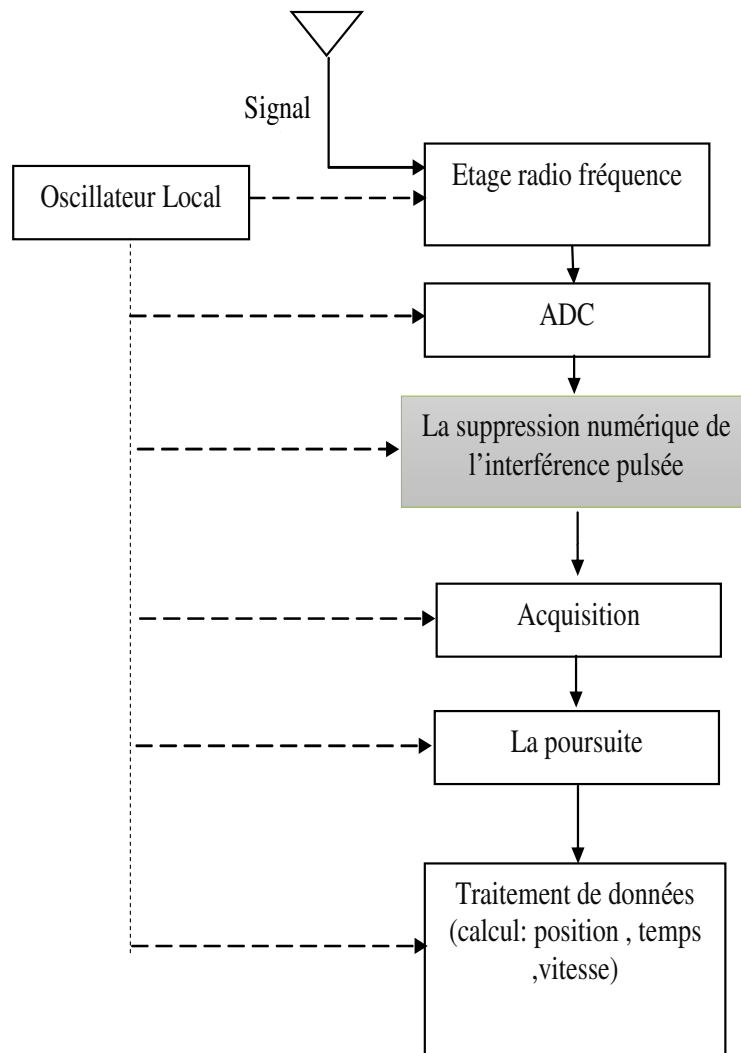


Figure IV. 11. Schéma bloc de la suppression numérique de l'interférence pulsée.

Dans la suite, nous présentons les résultats des différentes architectures CNN sur les bases de données GNSS, suivis d'une analyse comparative des différents types de modèles CNN. Pour évaluer les performances des différentes architectures CNN, nous avons utilisé les critères suivants :

✚ La qualité du signal en terme de RMSE :

RMSE est une mesure fréquemment utilisée des différences entre les vraies valeurs (valeurs d'échantillon du signal original sans interférence) et les valeurs de signal récupéré après la suppression des interférences. Le critère d'évaluation est donné par l'équation suivante :

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (IV.8)$$

✚ Une autre métrique utilisée dans de nombreux domaines tels que : applications GNSS, domaine biomédical et autres. Ce critère est bien connu sous le nom de coefficient de corrélation R et il indique le degré de similarité entre deux signaux (récupéré et l'origine) :

$$R = \frac{\sum_{i=1}^N (y_i - \bar{y}_i)(y_i - \bar{\hat{y}}_i)}{\sqrt{\sum_{i=1}^N (y_i - \bar{y}_i)^2} \sqrt{\sum_{i=1}^N (y_i - \bar{\hat{y}}_i)^2}} \quad (IV.9)$$

Ce travail utilise les signaux GNSS pour vérifier les performances de notre méthode de suppression et d'identification d'interférences, basée sur trois architectures de réseaux de neurones convolutifs (CNN). Nous évaluons cette méthode en utilisant différents niveaux de rapport signal sur interférence (ISR), allant de 10 dB à 60 dB avec un pas de 10dB (voir les Figures IV.2, IV.3, IV.4 et IV.5).

Pour trouver une architecture adaptée à notre problème de détection d'interférences, nous avons proposé d'utiliser trois types d'architectures de CNN dans les sous-sections suivantes.

VI.4.1. Conception du modèle neuronal CNN 28×28

Comme le montre la Figure IV.12, la première architecture proposée est constituée d'une seule couche de convolution avec une entrée de dimension 28×28. Un filtre de taille 5×5 est utilisé, et 20 filtres ont été utilisés dans cette couche. Après cette couche convolutif, une fonction d'activation linéaire Relu est appliquée pour forcer les neurones à retourner des valeurs positives. Ensuite, une couche de Maxpooling est appliquée pour réduire la taille de l'image et la quantité des paramètres. La dernière couche est une couche entièrement connectée, à laquelle est ajouté un classifieur complètement connecté, qui suit le même principe que les réseaux de neurones multicouches.

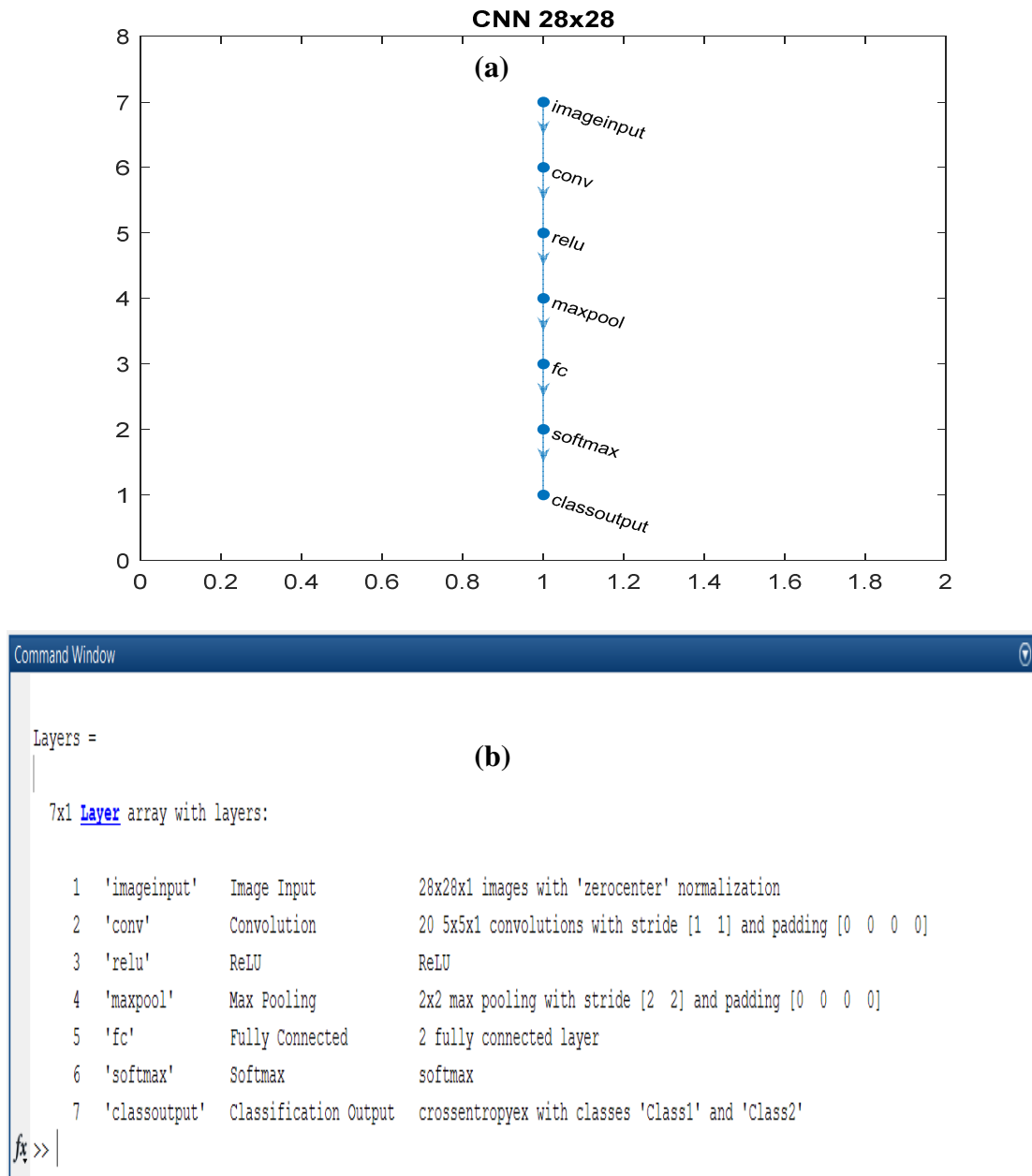


Figure IV. 12. Architecture CNN de dimension 28×28, (a) Architecture et (b) Détail des couches de convolution sous MATLAB.

L'évolution de l'entraînement du modèle CNN avec une entrée de dimension 28×28 sur l'apprentissage et la validation est présentée dans la Figure IV.13. On peut observer que le modèle semble commencer à surapprendre après environ 200 itérations, ce qui se manifeste par une divergence entre les courbes d'Accuracy et Loss pour les données d'apprentissage et de validation.

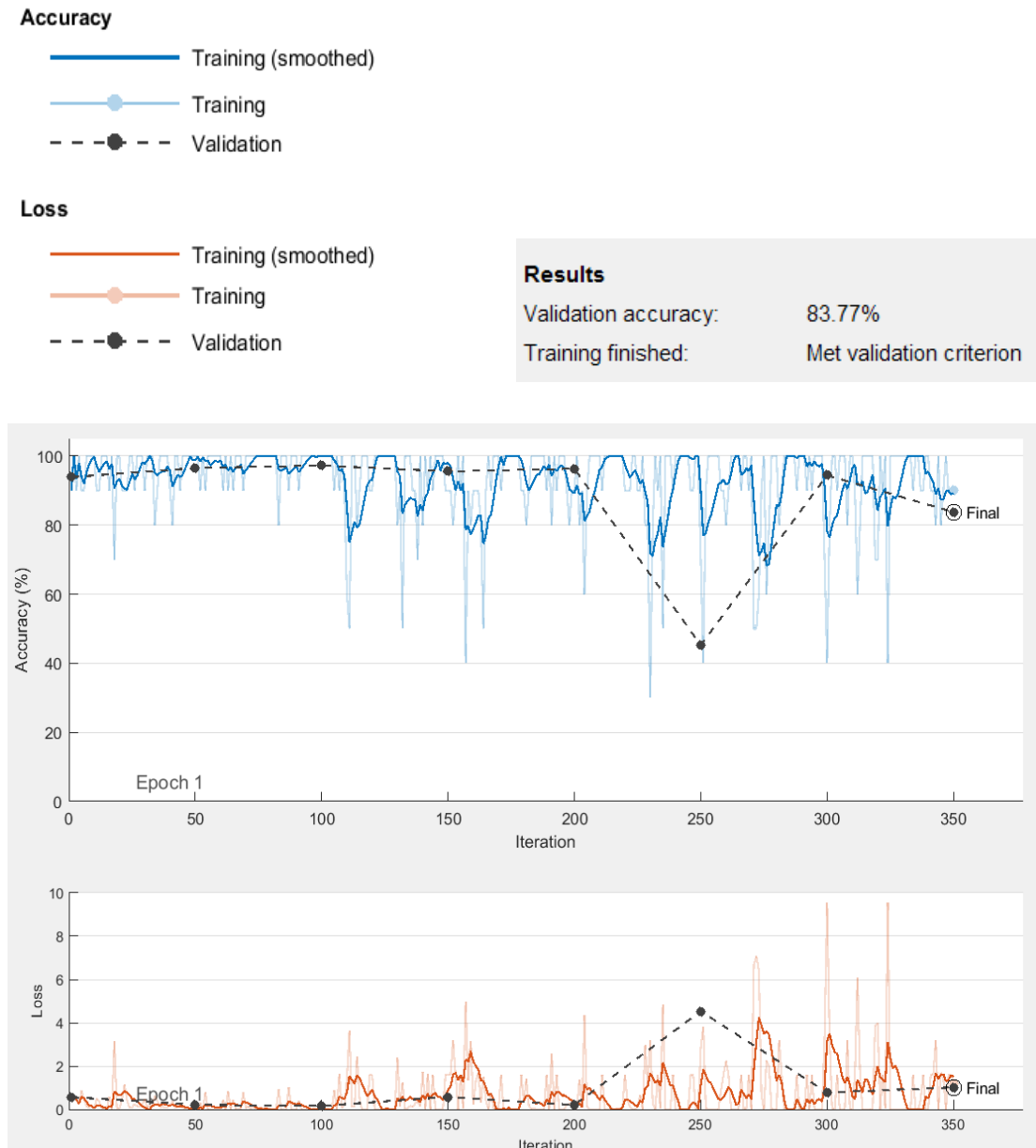


Figure IV. 13. Évolution des critères Accuracy et Loss en fonction du nombre d’itérations sur la base de données GNSS durant la phase d’apprentissage avec une entrée de dimension 28×28 .

VI.4.2. Conception du modèle neuronal CNN 48×48

La deuxième architecture est composée de trois couches de convolution avec la configuration présentée dans la Figure IV.14. L’image en entrée est de taille 48×48 pixels. L’image passe d’abord à la première couche de convolution, qui est composée de 100 filtres de taille 7×7 . Après cette convolution, 150 filtres de taille 4×4 sont créés dans la deuxième couche de

convolution. Les trois couches de convolution sont suivies d'une fonction d'activation Relu qui force les neurones à retourner des valeurs positives. La dernière couche de convolution est constituée de 250 filtres de taille 4×4. Ensuite, une couche de Maxpooling, une couche entièrement connectée (FC) et une couche de Dropout sont appliquées pour améliorer la généralisation du modèle. La couche Dropout, est une technique de régularisation dans les réseaux de neurones qui consiste à désactiver aléatoirement un certain pourcentage de neurones pendant l'entraînement. Cela permet de prévenir le surapprentissage en évitant la dépendance excessive des neurones à certains motifs (pattern) de l'ensemble de données d'apprentissage. Après la couche FC, une fonction de probabilité Softmax est utilisée pour obtenir la sortie en deux classes (avec et sans interférence).

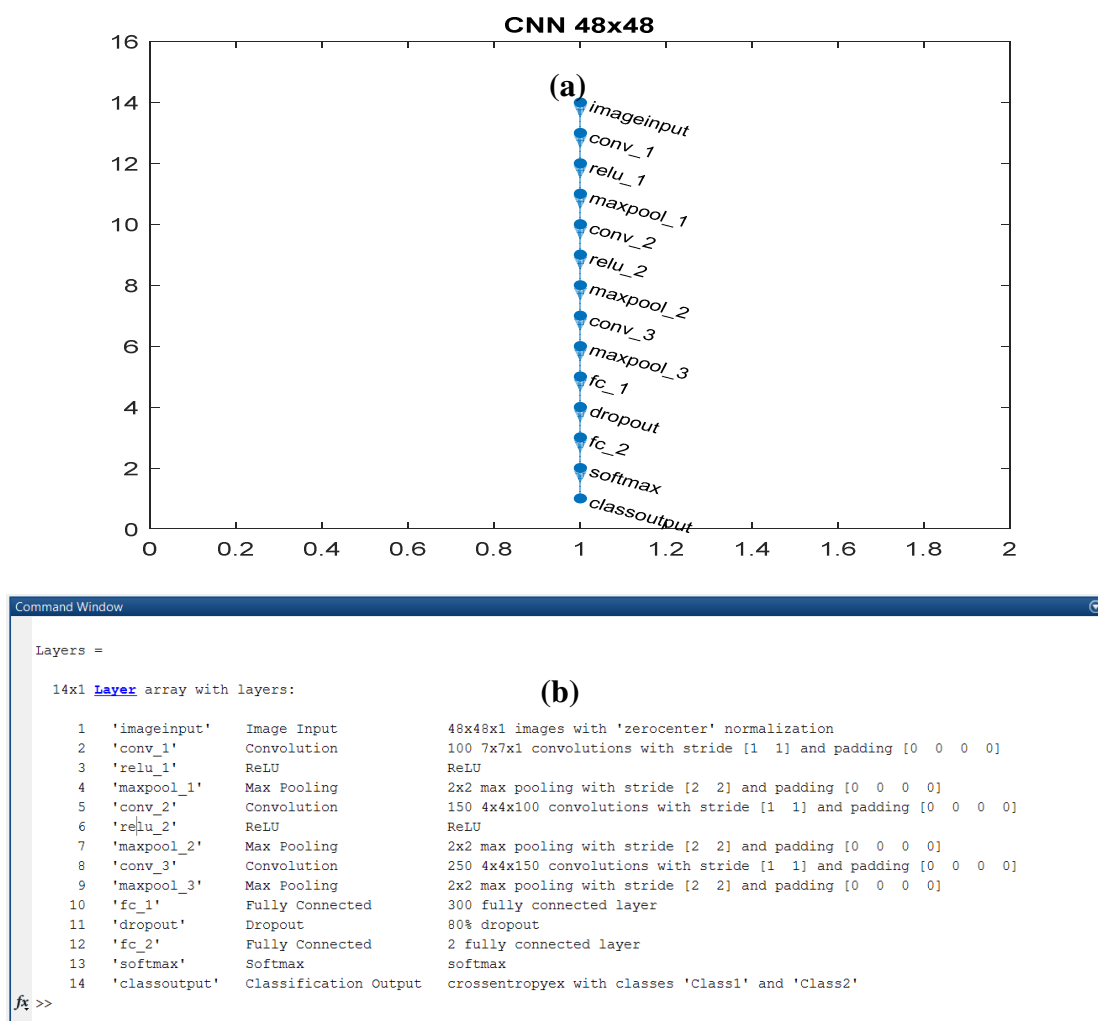


Figure IV. 14. Architecture CNN de dimension 48×48, (a) Architecture et (b) Détail des couches de convolution sous MATLAB.

La simulation du modèle CNN avec une dimension d'entrée 48×48, est donnée dans la Figure IV.15, on peut observer que l'Accuracy augmente et que la Loss diminue au fil des itérations,

ce qui indique une amélioration de la performance du modèle. Dans cette architecture, une erreur de validation égale à 2,06% a été obtenue, ce qui correspond à une Accuracy de 97,94%. Ces résultats sont très encourageants et suggèrent que cette architecture de CNN est bien adaptée pour la détection d'interférences dans les signaux GNSS.

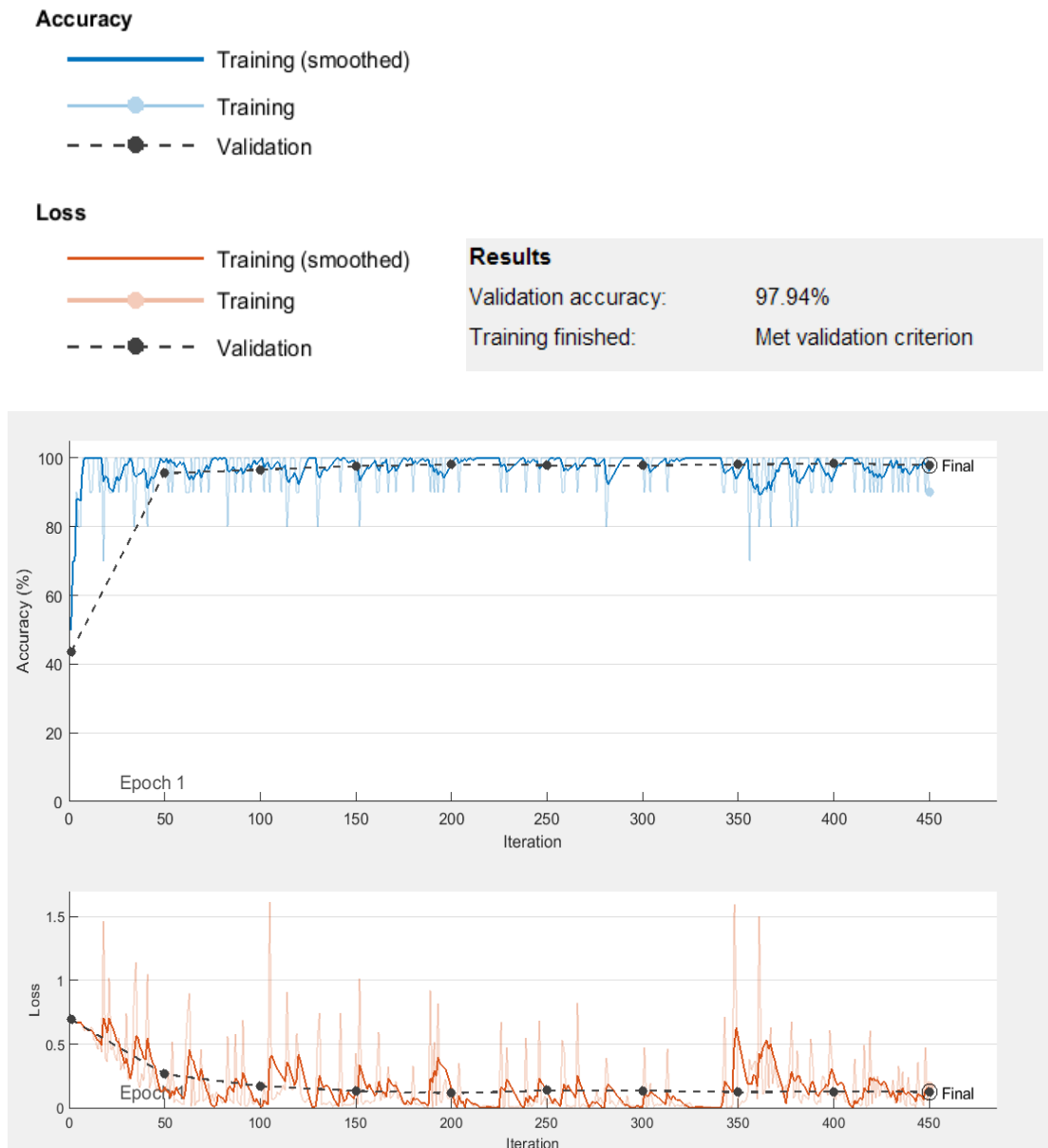


Figure IV. 15. Évolution des critères Accuracy et Loss en fonction du nombre d'itérations sur la base de données GNSS durant la phase d'apprentissage avec une entrée de dimension 48×48.

VI.4.3. Conception du modèle neuronal CNN 64×64

La troisième architecture CNN (Figure IV.16) utilise des combinaisons pour extraire automatiquement les caractéristiques de différents états des signaux GNSS, tels que le signal utile et le signal avec interférence. Dans ce cas, le processus d'identification d'interférences comprend plusieurs étapes:

- L'image en entrée du modèle CNN est de taille 64×64 pixels.
- Trois couches successives sont utilisées : une couche de convolution, suivie d'une couche BatchNorm et enfin d'une fonction d'activation ReLU.

Batchnorm : Batch normalization (ou normalisation de paquet) est une technique de normalisation des données dans les réseaux de neurones artificiels. Elle consiste à normaliser les activations de chaque couche en les mettant à une moyenne de zéro et une variance de un, sur un paquet d'exemples d'entraînement. Cela permet d'accélérer la convergence de l'apprentissage, de réduire la dépendance des résultats par rapport à l'initialisation des poids, et de régulariser le modèle en réduisant le risque de surapprentissage (overfitting).

- Après chaque fonction d'activation Relu, une couche de Maxpooling est utilisée pour réduire la taille de l'image.
- Deux couches entièrement connectées (FC) pour obtenir les caractéristiques de l'image.
- Une couche de Dropout pour éviter le surapprentissage.
- Une couche de classification Softmax pour classifier l'image en différentes classes.

Cette architecture permet d'extraire automatiquement les caractéristiques d'image à partir de différents états des signaux GNSS, ce qui permet une meilleure identification des interférences.

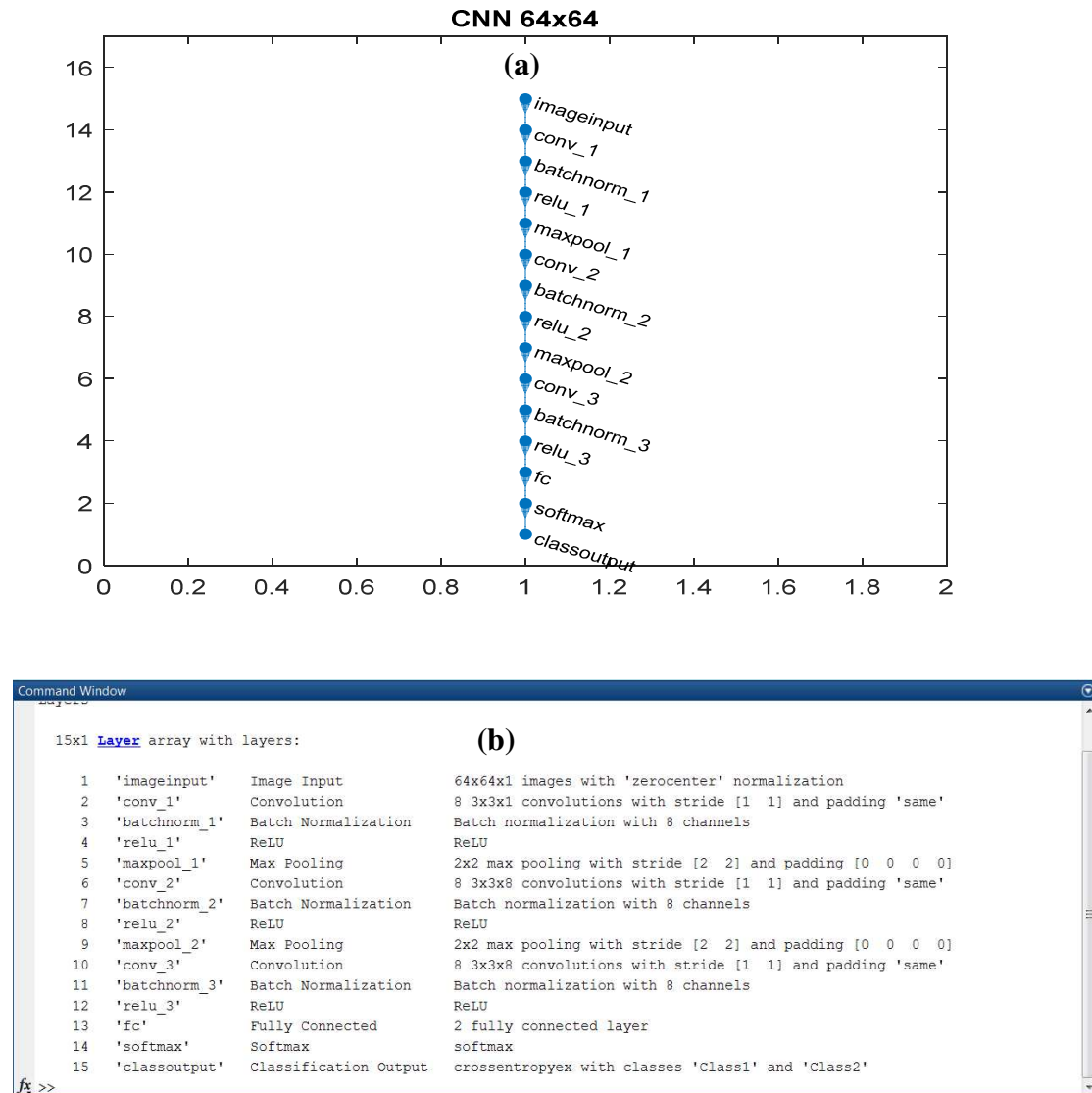


Figure IV. 16. Architecture CNN de dimension 64×64, (a) Architecture et (b) Détail des couches de convolution sous MATLAB.

Comme illustre la Figure suivante, on peut observer l'évolution des critères Accuracy et Loss en fonction du nombre d'itérations durant la phase d'apprentissage avec une entrée de dimension 64×64. Cette architecture donne de bonnes performances dans la validation, avec un Accuracy d'environ 97,18%.



Figure IV. 17. Évolution des critères Accuracy et Loss en fonction du nombre d’itérations sur la base de données GNSS durant la phase d’apprentissage avec une entrée de dimension 64×64 .

VI.4.4. Comparaison entre les différents modèles CNN

Lors de la phase de conception des modèles CNN pour les trois architectures proposées précédemment, nous avons remarqué que l’architecture CNN 28×28 donnait de mauvais résultats lors de la validation. En revanche, il est intéressant de noter que la résolution plus élevée de l’entrée de dimension 48×48 peut contribuer à une meilleure représentation des signaux GNSS, ce qui se traduit par de meilleures performances du modèle.

Cependant, dans le cas de l'architecture CNN 64×64, nous avons constaté que l'Accuracy dans la validation était inférieure à celle de l'architecture CNN 48×48. La différence entre ces deux architectures CNN 48×48 et CNN 64×64 réside dans l'ajout des couches Dropout et Batchnorm, respectivement. Il est donc intéressant de noter que la couche Dropout a donné de meilleurs résultats en termes de performance.

Ces observations suggèrent que l'ajout de la couche Dropout permet de régulariser le modèle et de réduire le surapprentissage, ce qui conduit à de meilleures performances dans la validation. Ainsi, dans le cas de la détection d'interférences dans les signaux GNSS, l'architecture CNN avec une entrée de dimension 48×48 et l'utilisation de la couche Dropout semble être la meilleure configuration pour obtenir des résultats de qualité.

Les tests des différents modèles CNN sont répertoriés dans le tableau IV.2. En particulier, la racine de l'erreur quadratique RMSE et le coefficient de correction R sont calculés pour chaque cas d'ISR sur les bases de données de tests. Nous avons effectué les mêmes simulations pour les différentes architectures CNN : 28×28, 48×48 et 64×64.

Tableau IV. 2. Les résultats obtenus par les différentes architectures CNN (28×28, 48×48, et 64×64) sur la base de données de test des signaux GNSS.

ISR	CNN 28×28		CNN 48×48		CNN 64×64	
	RMSE	R	RMSE	R	RMSE	R
10	0.2815	0.9998	0.2815	0.9998	0.2815	0.9998
20	0.8903	0.9975	0.8903	0.9975	0.8903	0.9975
30	2.8154	0.9757	2.3493	0.9823	2.4755	0.9807
40	5.6468	0.9088	2.6432	0.9775	3.2076	0.9672
50	11.366	0.7356	3.0060	0.9712	6.1451	0.8932
60	3.8592	0.9526	3.2752	0.9655	4.8190	0.9286

Nous avons abouti à la même conclusion lors de la phase de validation, à savoir que l'architecture CNN 48×48 offre de meilleurs résultats dans tous les cas d'ISR. Ceci est soutenu par les résultats du tableau IV.2, où l'on constate des valeurs plus faibles de RMSE et des coefficients de correction R plus élevés pour l'architecture CNN 48×48 par rapport aux autres architectures. Ces résultats confirment que l'utilisation de la résolution 48×48 et de la couche Dropout permet d'obtenir de meilleures performances de détection d'interférences dans les signaux GNSS.

Pour valider notre méthode, nous avons utilisé un classifieur robuste aux valeurs aberrantes, l'OSVM (One-Class Support Vector Machine). Les Figures IV.18 et IV.19 illustrent la comparaison entre l'OSVM standard et les différents résultats obtenus par la meilleure architecture CNN 48×48.

À partir de ces comparaisons, nous pouvons conclure que notre méthode basée sur les réseaux de neurones conventionnels dépasse de manière significative l'OSVM standard en termes de RMSE et de coefficient de correction R. Les valeurs obtenues pour la racine de l'erreur quadratique RMSE et le coefficient de correction R avec notre méthode sont considérablement meilleures que celles obtenues avec l'OSVM standard.

Ces résultats démontrent l'efficacité et la pertinence de notre approche basée sur les réseaux de neurones CNN pour la détection d'interférences dans les signaux GNSS. L'utilisation de l'architecture 48×48 spécifiquement a permis d'améliorer significativement les performances par rapport à l'OSVM standard.

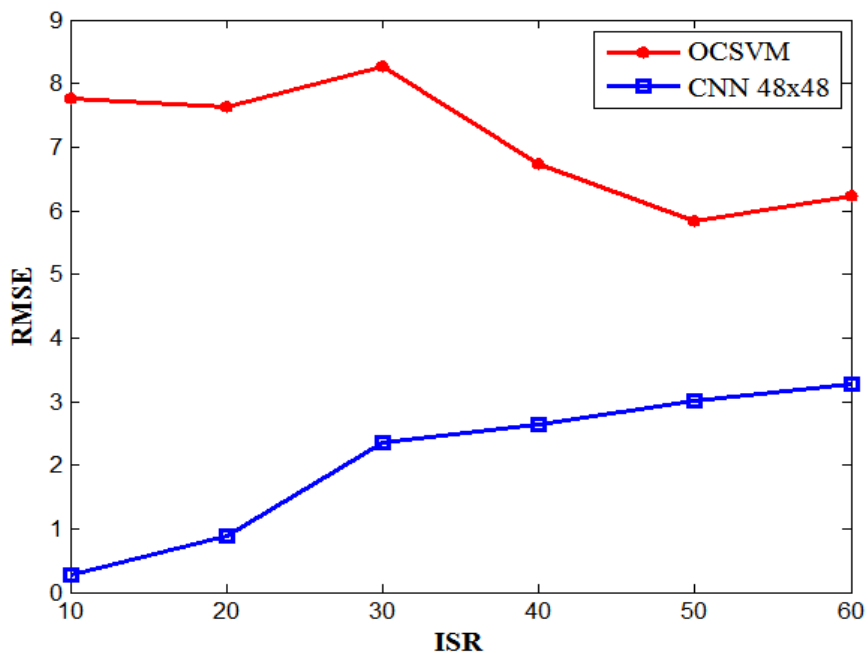


Figure IV. 18. Evolution de l'architecture CNN 48×48 en terme de RMSE en fonction des différentes valeurs ISR.

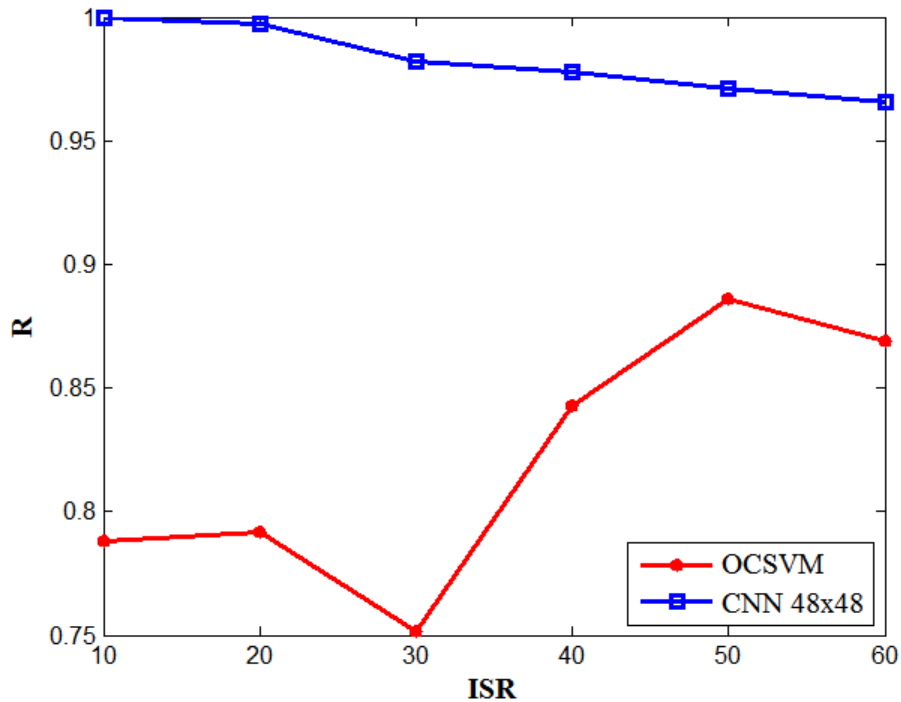
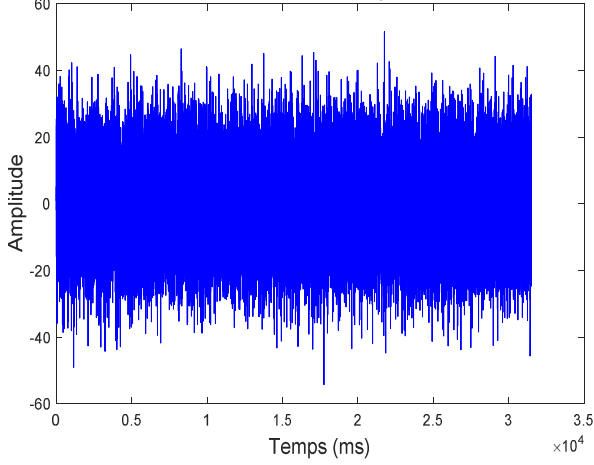


Figure IV. 19. Evolution de l'architecture CNN 48×48 en terme de R en fonction des différentes valeurs ISR.

L'application de notre algorithme de classification CNN 48×48 sur les différentes valeurs d'ISR est représentée dans la Figure suivante. Cette figure illustre les résultats obtenus pour chaque valeur d'ISR. Elle met en évidence la capacité de notre algorithme à détecter et supprimer les interférences dans les signaux GNSS avec précision et efficacité.

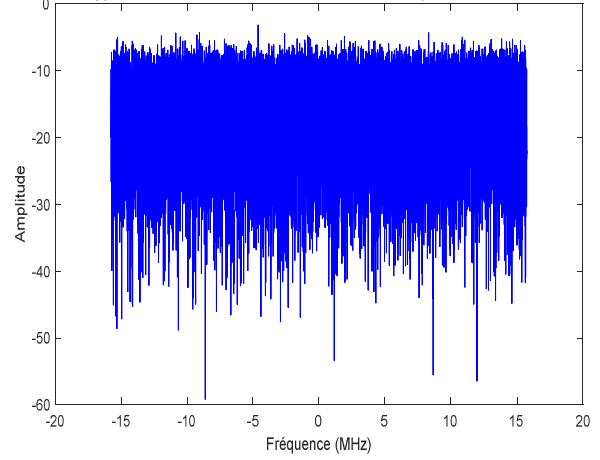
En analysant la figure dans le domaine fréquentiel, on constate que les composantes du signal d'interférence sont clairement identifiées et séparées du spectre du signal GNSS. Cette séparation permet de cibler spécifiquement les interférences et de les atténuer, tout en maintenant la qualité et l'intégrité du signal GNSS.

Détection des interférences en utilisant CNN 48x48 (ISR=10, RMSE=0.28154, R=0.99975)

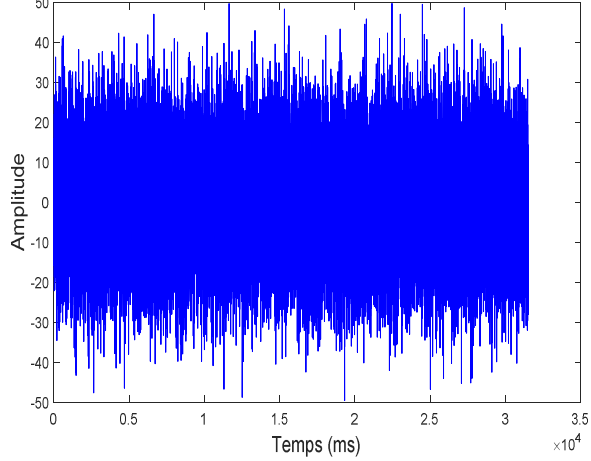


(a)

Détection et Suppression des interférence en utilisant CNN 48x48 (ISR=10, RMSE=0.28154, R=0.99975)

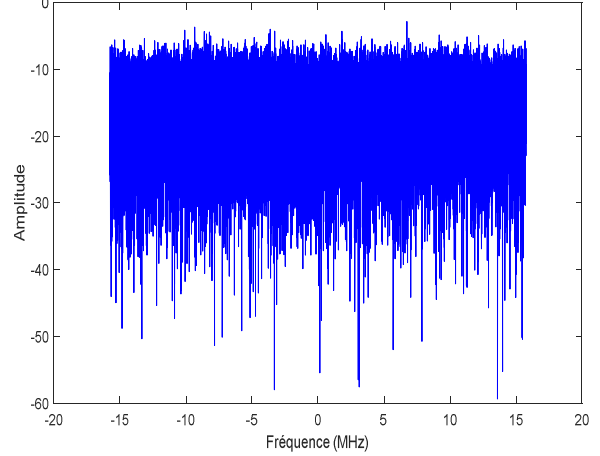


Détection des interférences en utilisant CNN 48x48 (ISR=20, RMSE=0.89032, R=0.99747)

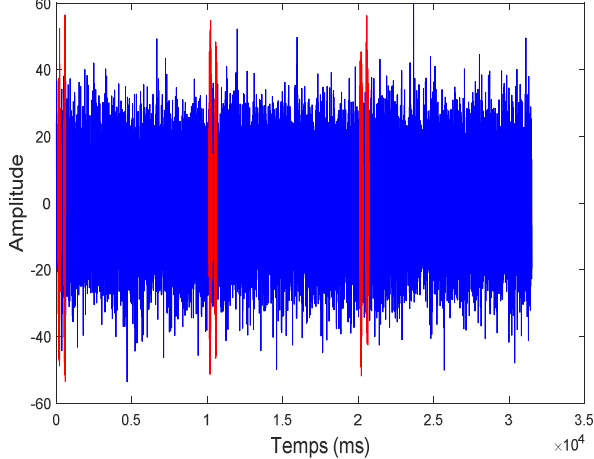


(b)

Détection et Suppression des interférence en utilisant CNN 48x48 (ISR=20, RMSE=0.89032, R=0.99747)

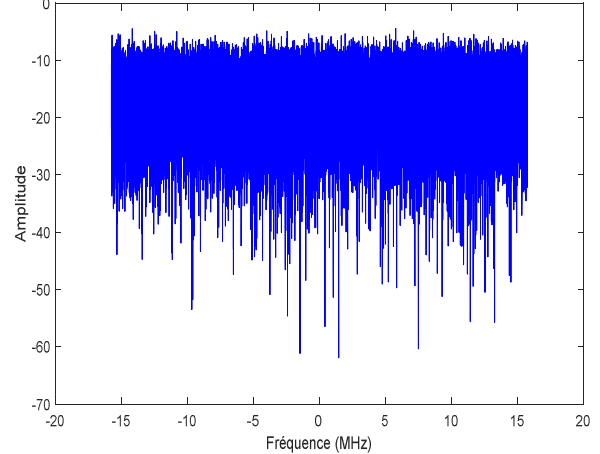


Détection des interférences en utilisant CNN 48x48 (ISR=30, RMSE=2.3493, R=0.98229)



(c)

Détection et Suppression des interférence en utilisant CNN 48x48 (ISR=30, RMSE=2.3493, R=0.98229)



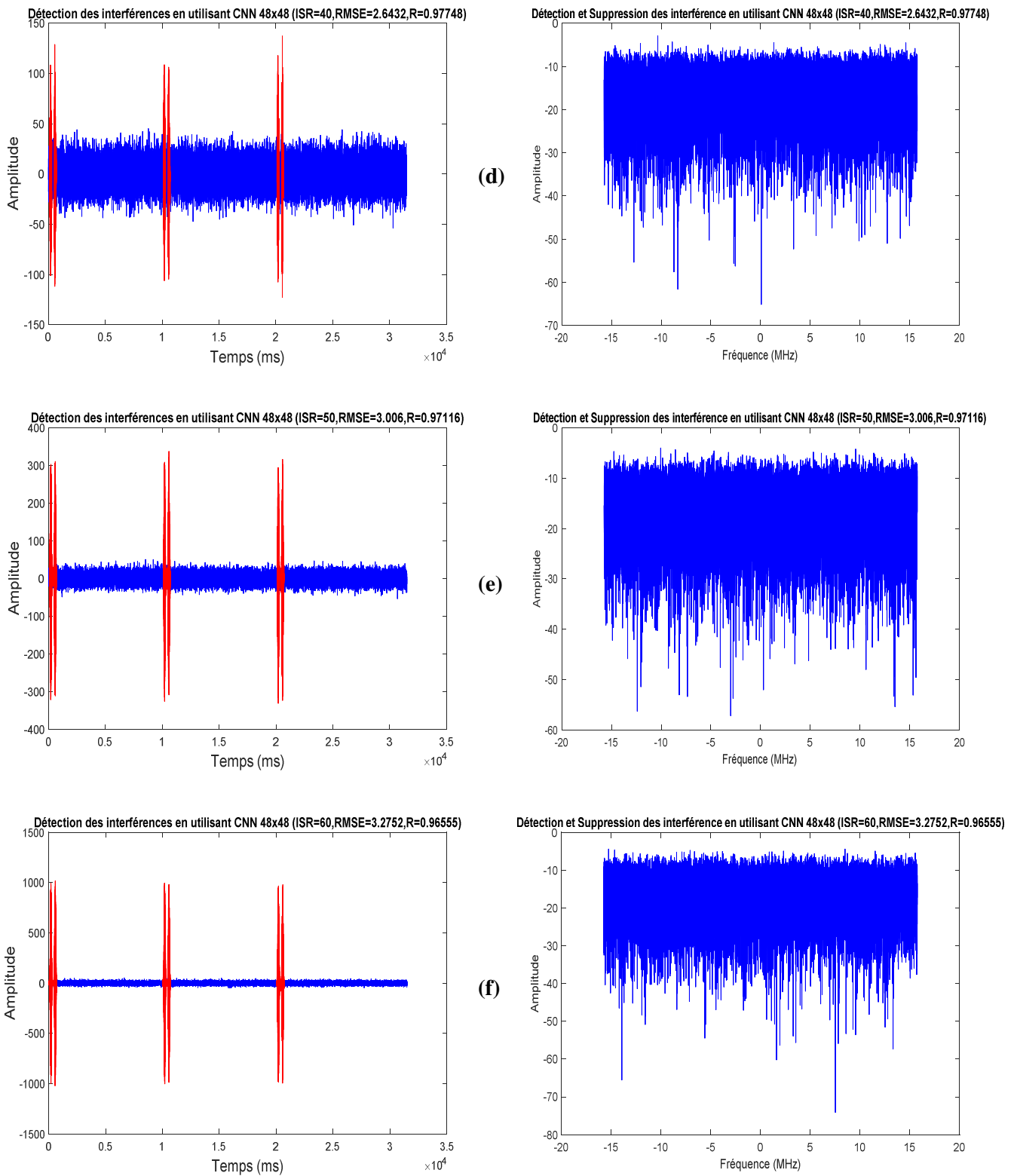


Figure IV. 20. Application de l'architecture CNN 48×48 pour la détection et la suppression des interférences, (a) ISR=10dB, (b) ISR=20dB, (c) ISR=30dB, (d) ISR=40dB, (e) ISR=50dB, (f) ISR=60dB.

IV.5. Conclusion

L'objectif principal de ce dernier chapitre était de développer un modèle de classification robuste basé sur les réseaux de neurones convolutifs (CNN) pour la détection et la suppression des interférences dans les récepteurs GNSS. Pour atteindre cet objectif, nous avons utilisé un ensemble de données de signaux GNSS et exploré différentes architectures CNN.

Nous avons entraîné plusieurs modèles CNN en utilisant ces données pour la classification des signaux GNSS, en les distinguant entre les signaux avec interférence et les signaux sans interférence. Ensuite, nous avons comparé les performances de ces modèles avec une méthode de classification traditionnelle, l'OCSVM (one-class support vector machine).

Les résultats de notre étude ont démontré que les modèles CNN ont surpassé la méthode OCSVM en termes d'erreur (RMSE) et de coefficient de corrélation (R). Cela indique que les modèles CNN ont la capacité d'apprendre à extraire des caractéristiques pertinentes à partir des signaux GNSS et à les utiliser pour différencier les signaux avec interférence des signaux sans interférence.

Plus spécifiquement, nous avons constaté que l'architecture CNN avec une dimension d'entrée de 48×48 a montré des performances supérieures. Cette architecture a démontré sa capacité à classifier les signaux GNSS avec une précision élevée, ce qui en fait un outil puissant pour l'identification et la suppression des interférences dans les récepteurs GNSS.

Conclusion générale

Conclusion générale

Récemment, les sociétés de recherche se sont concentrées sur les contre-mesures pour les interférences GNSS, causées par les signaux d'accès faibles des systèmes GNSS. Des expériences pratiques ont montré que les récepteurs GNSS peuvent être facilement perturbés par des interférences intentionnelles et non intentionnelles.

Dans ce mémoire, notre objectif est de détecter et de supprimer les différents types d'interférences aux récepteurs GNSS afin d'améliorer les performances des processus de traitement. Nous proposons une méthode basée sur les réseaux de neurones convolutionnels (CNN).

Pour atteindre notre objectif, nous avons mené plusieurs expériences. Tout d'abord, nous avons formulé le problème en utilisant la méthode CNN, puis nous l'avons comparée notre méthode avec la méthode traditionnelle de classification OCSVM. Notre méthode commence par la division du signal en blocs, chaque bloc étant converti en une image bidimensionnelle 2D. Nous avons utilisé trois bases de données avec des dimensions de bloc de 28×28 , 48×48 et 64×64 pixels. Ensuite, nous avons procédé à la phase de classification CNN. Il convient de souligner que le choix de la base d'apprentissage est crucial dans la conception d'un classifieur CNN. Pour évaluer notre méthode, nous avons utilisé plusieurs critères tels que l'erreur de classification, la racine de l'erreur quadratique moyenne (RMSE) et le coefficient de corrélation (R).

Nous avons proposé trois architectures différentes et avons constaté que l'architecture avec des blocs de 28×28 pixels donnait de mauvais résultats lors de l'apprentissage. En revanche, nous avons observé que l'architecture avec des blocs de 48×48 pixels permettait une meilleure représentation des signaux GNSS. Pour l'architecture avec des blocs de 64×64 pixels, nous avons constaté une précision inférieure par rapport à celle de l'architecture avec des blocs de 28×28 et 48×48 pixels. La différence entre ces architectures réside dans l'ajout des couches de régularisation, Dropout et BatchNormalization. Nous avons constaté que la couche Dropout produisait de meilleurs résultats en termes de performances. Ainsi, l'architecture CNN avec des blocs de 48×48 pixels et l'utilisation de la couche Dropout semble être la meilleure configuration pour obtenir de bons résultats.

En conclusion, notre travail vise à détecter et à supprimer les interférences dans les récepteurs GNSS en utilisant une approche basée sur les CNN. Nous avons réalisé plusieurs expériences et comparé notre méthode avec l'OCSVM. Les résultats montrent que notre méthode, notamment avec l'architecture CNN de blocs de 48×48 pixels et l'utilisation de la couche Dropout, offre de bonnes performances pour la détection et la suppression des interférences.

Notre travail reste ouvert à de futures comparaisons et hybridations avec d'autres méthodes de classification. Il reste encore beaucoup de potentiel à explorer pour améliorer la détection et la suppression des interférences dans les récepteurs GNSS.

Une direction future intéressante serait d'explorer l'hybridation de notre approche basée sur les CNN avec d'autres techniques de classification, telles que les réseaux de neurones récurrents (RNN) ou les machines à vecteurs de support (SVM). Cette hybridation pourrait potentiellement améliorer les performances de détection et de suppression des interférences GNSS.

De plus, il serait intéressant d'explorer d'autres méthodes de conversion des signaux GNSS en images 2D, en utilisant des techniques d'analyse temps-fréquence telles que la transformée en ondelettes (WT), la transformée de Fourier à court terme (STFT) et la transformée S. Ces méthodes permettent de capturer les caractéristiques temporelles et fréquentielles des signaux GNSS, ce qui pourrait potentiellement améliorer la représentation des données et la capacité de détection des interférences.

En tant que dernière perspective, dans notre travail, nous avons utilisé les méthodes de régularisation Dropout et BatchNormalization pour éviter le surapprentissage dans notre modèle CNN. Cependant, il existe d'autres méthodes de régularisation que nous pourrions explorer pour améliorer nos résultats, telles que : l'augmentation des données, le DropConnect, et les régularisations L1 et L2.

Bibliographie

Bibliographie

- [1] M. A. Fatoumi, M. I. Kalache, and A. Krobba, "Étude et amélioration des performances de signaux GNSS," *Mémoire de Master, Université Medea*, 2022.
- [2] G. Georges, "Algorithmes de calcul de positions GNSS basés sur les méthodes des moindres carrés avancées," Université de Technologie de Belfort-Montbeliard, 2016.
- [3] Z. A. Khezzar, "Traitement des signaux GNSS, Option : Micro Ondes," Thèse de doctorat, Université de Batna, 2021.
- [4] I. Mahrez and M. Badaoui, "Détection d'interférences pour les systèmes de navigation par satellites en utilisant les méthodes d'optimisation multi-objectif NSGAI," Université Abbes Laghrour Khenchela, 2022.
- [5] C. Lubeigt, "Traitement du signal pour la télédétection par GNSS," Toulouse, ISAE, 2023.
- [6] R. W. Renbiao, W. D. L. Wenyi, L. Wang, and J. Qiongqiong, "Adaptive Interference Mitigation in GNSS," *Springer Nature Singapore*, 2015.
- [7] Z. A. Khezzar, R. Benzid, and L. Saidi, "New Thresholding Technique in DCT Domain for Interference Mitigation in GNSS Receivers," *Traitement du Signal*, vol. 37, 2020.
- [8] I. E. Mehr and F. Dovis, "Detection and Classification of GNSS Jammers Using Convolutional Neural Networks," in *2022 International Conference on Localization and GNSS (ICL-GNSS)*, 2022, pp. 01-06.
- [9] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *Journal of big Data*, vol. 8, pp. 1-74, 2021.
- [10] E. D. Kaplan, C. Hegarty, and A. House. (2017). *Understanding GPS/GNSS*.
- [11] L. Musumeci, J. T. Curran, and F. Dovis, "A comparative analysis of adaptive notch filtering and wavelet mitigation against jammers interference," *Navigation: Journal of The Institute of Navigation*, vol. 63, pp. 533-550, 2016.
- [12] S. Savasta, F. Dovis, R. Lesca, D. Margaria, and B. Motella, "On the interference mitigation based on ADC parameters tuning dans Proceedings of the IEEE/ION Position,," pp. 689- 695, 2000.
- [13] R. Lahaye and S. Ladet, "Les principes du positionnement par satellite: GNSS," *Le Cahier des Techniques de l'INRA*, pp. 9-18, 2014.
- [14] D. F. Nahimana, "Impact des multitrajets sur les performances des systèmes de navigation par satellite: contribution à l'amélioration de la précision de localisation par modélisation bayésienne," Ecole centrale de Lille, 2009.
- [15] T. Donal, " Système de positionnement par satellite," *Institut National de l'Information Géographique et Forestière*, vol. 109, pp. 59-84, 2019.
- [16] A.-M. TOBIE, "GNSS/5G Hybridization for urban navigation," Thèse de doctorat, université de Toulouse, 2021.
- [17] B. Beldjilali. (2017, Traitement des signaux GNSS pour optimiser les récepteurs de localisation.
- [18] N. Azi and S. Anki, "Conception et réalisation d'une application de géolocalisation Cas d'étude: Entreprise Bejaia Logistique," Université A/Mira de Bejaia, 2016.
- [19] E. D. Kaplan and C. J. Hegarty, "Understanding GPS: Principles and Applications, Norwood, MA: Artech House," ed: Inc, 2006.
- [20] L. Fillatre, " Méthodes de poursuite de phase pour signaux GNSS multifréquence en environnement dégradé," Thèse de doctorat, Thales Alenia Space, 2013.
- [21] S. Tay, "Analyse et modélisation de l'utilisation de signaux GNSS en environnement marin," Thèse de doctorat, Thèse de doctorat. Télécom Bretagne, Université de Rennes 1, Université de Rennes 1, 2012.

- [22] M. J. Lesouple, "Estimation parcimonieuse de biais multi trajets pour les systèmes GNSS," Thèse de doctorat, Université de Toulouse, 2019.
- [23] Z. A. Khezzar, "Acquisition Rapide des Signaux GNSS Par un Récepteur Logiciel," Mémoire de Magister en Electronique, Université de Batna 2, 2011.
- [24] S. Rougerie, "Algorithmes de diversité d'antennes appliqués à la réception des signaux GNSS en environnement urbain et sur terminal mobile," Toulouse, ISAE, 2012.
- [25] F. Dovis, *GNSS interference threats and countermeasures*: Artech House, 2015.
- [26] H. Denks, A. Hornbostel, and V. Chopard, "GNSS receiver testing by hardware simulation in an standardized pulsed and CW interference environment," 2009.
- [27] J.-M. Commenge, "Big Data et Intelligence Artificielle pour le Génie des Procédés."
- [28] P. Flach, *Machine learning: the art and science of algorithms that make sense of data*: Cambridge university press, 2012.
- [29] K.-L. Du and M. N. Swamy, "Neural networks and statistical learning," *Springer Science & Business Media*, 2013.
- [30] N. Linty, A. Farasin, A. Favenza, and F. Dovis, "Detection of GNSS ionospheric scintillations based on machine learning decision tree," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 55, pp. 303-317, 2018.
- [31] B. Abassi, "Approche de sélection des données biomédicales pour l'identification de pathologies en utilisant les supports vecteurs machines," Thèse de doctorat, Thèse de doctorat. Université Mohamed Boudiaf, M'sila, 2019.
- [32] M. D. Youcef, "Deep learning pour la classification des images," mémoire de master LMD en Informatique, Université Abou bakr Belkaid-Tlemcen, 2017.
- [33] S. Reguieg and A. Zeghaba, "Étude comparative de quelques outils de classification sur des données médicales," Mémoire de Master, Université Abou Bekr Belkaid - Tlemcen, 2017.
- [34] S. Samy, "Réseaux de neurones convolutionnels pour la détection précoce de la rétinopathie diabétique," Université Mouloud Mammeri, 2019.
- [35] H. Mebarki and O. Ghedir, "Système de communication entre les véhicules (VNET) en utilisant la communication sans fil et les méthodes de l'intelligence artificielle," *Université Abbes Laghrour Khenchela*, 2019.
- [36] N. Benoudjit, C. Archambeau, A. Lendasse, J. A. Lee, and M. Verleysen, "Width optimization of the Gaussian kernels in Radial Basis Function Networks," *ESANN*, vol. 2, pp. 425-432, 2002.
- [37] O. Bousquet, "Introduction au Support Vector Machines (SVM)," Centre de mathématiques appliquées, Ecole polytechnique de Palaiseau, 2001.
- [38] S. Lecomte, C. Richard, R. Lengellé, F. Capman, and B. Ravera, "Machines à vecteur de support 1-classe (OC-SVM) pour la détection non supervisée d'événements sonores anormaux," in *GRETSI 2011*, 2011.
- [39] G. M. X. He, D. Maquin, J. Ragot, P. Beuseroy, and A. Smolarz, "One-class SVM in multi-task learning," *HAL-00593397*, 2012.
- [40] D. Takouachet and R. Belhabes, "Classification de l'environnement intérieur à base des mesures du signal WiFi," Mémoire de Master, Systèmes des Télécommunications, Université Abbes Laghrour, Khenchela, 2021.
- [41] S. Hadj Moussa, H. Dahlouk, and S. Tchoketch Kebir, "Détection et Segmentation Automatique Des Tumeurs Cérébrales Par Apprentissage Profond (Deep Learning)," 2020.
- [42] B. Marir, M. Kalla, F. Douak, and A. Daamouche, "A modular support vector machine for active learning of urban remote sensing images classification in Algeria," *Journal of the Indian Society of Remote Sensing*, vol. 46, pp. 515-529, 2018.
- [43] A. Habba and O. Ishak, "La classification des images satellitaires par l'apprentissage profonde (deep learning)," *Mémoire Master, Option: Systèmes Intelligents, Université Ahmed Draïa-Adrar*, 2019.
- [44] B. Hakim and S. Brahim, "Classification des images avec les réseaux de neurones convolutionnels, Mémoire de Master,," Université Abou Bakr Belkaid Tlemcen., 2018.

- [45] M. Boughaba, B. Boukhris, and M. Meflah, "L'apprentissage profond (Deep Learning) pour la classification et la recherche d'images par le contenu," 2017.
- [46] M. E. M. Choini, "Détection d'objets basé Faster R-CNN," *Université de Guelma*, 2020.
- [47] A. Das, M. W. Ansari, and R. Basak, "Covid-19 face mask detection using TensorFlow, Keras and OpenCV," *IEEE 17th India Council International Conference (INDICON)*, pp. 1-5, 2020.
- [48] A. S. Bennacer and I.-A. Rabahi, "Détection Et Reconnaissance De Caractères Par L'algorithme De Deep Learning East Sous L'ocr Tesseract," *Mémoire de Master, Université Saad Dahleb - Blida*, 2020.
- [49] F. Abid and B. Boulares, "Détection de la rétinopathie diabétique avec le deep learning transfer learning, CNN, U-Net," *Mémoire de Master, Université Saad Dahlab, Blida*, 2021.
- [50] A. Mahdjoubi, A. Halbigue, and R. Kara, "Implémentation d'un réseau de neurone convolutionnel CNN pour la reconnaissance automatique des mots manuscrits en arabe," *Université Yahia Fares de Medea*, 2019.
- [51] M. Parizeau, "Réseaux de neurones," *GIF-21140 et GIF-64326*, vol. 124, 2004.
- [52] H. Fethallah, M. Mohammed, B. Akkacha, and S. M. Ismail, "Classification des images par les réseaux de neurones," 2017.
- [53] J. Dean, G. S. Corrado, R. Monga, K. Chen, M. Devin, Q. V. Le, M. Z. Mao, M. A. Ranzato, A. Senior, P. Tucker, K. Yang, and A. Y. Ng, "Large Scale Distributed Deep Networks," *Advances in neural information processing systems*, vol. 25, pp. 1-9, 2012.
- [54] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, USA*, 2015.
- [55] M. Ben Lazreg, "Recherche de l'information dans les réseaux de neurones convolutifs pré-entraînés," *École de technologie supérieure*, 2020.
- [56] Y. J. Heo, S. J. Kim, D. Kim, K. Lee, and W. K. Chung, "Super-high-purity seed sorter using low-latency image-recognition based on deep learning," *IEEE Robotics and Automation Letters*, vol. 3, pp. 3035-3042, 2018.
- [57] X. Zhang and D. Wang, "Deep learning based binaural speech separation in reverberant environments," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 25, pp. 1075-1084, 2017.
- [58] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *IEEE Computational Intelligence Magazine*, vol. 13, pp. 55-75, 2018.
- [59] D. A. de Diego, N. Ferrara, J. Nurmi, and E.-S. Lohan, "Interference mitigation in the E5a galileo band using an open-source simulator," *Inside GNSS*, vol. 11, pp. 55-63, 2016.
- [60] O. Julien, L. Priya, J.-L. Issler, and L. Lestarquit, "Estimating the ionospheric delay using GPS/Galileo signals in the E5 band," *Inside GNSS*, vol. 10, pp. 55-64, 2015.
- [61] L. Wen, X. Li, L. Gao, and Y. Zhang, "A new convolutional neural network-based data-driven fault diagnosis method," *IEEE Transactions on Industrial Electronics*, vol. 65, pp. 5990-5998, 2017.
- [62] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, pp. 541-551, 1989.