

# Université Mohamed Boudiaf - M'sila

FACULTE DE TECHNOLOGIE

DEPARTEMENT ELECTRONIQUE



Numéro de série : .....

Numéro d'inscription : .....

## Thèse

Présentée pour l'obtention du diplôme de

**DOCTORAT SCIENCES**

**Spécialité : Electronique**

**Option : Electronique**

## THEME

**La segmentation d'images par les modèles de Markov cachés**

**(HMM) pour l'identification biométrique**

Présenté par

**Djalab Abdelhak**

Soutenue le : .... / ..... / .....

**Devant le jury composé de :**

Pr : Laadjal Mohamed	Université Mohamed Boudiaf M'sila	Président
Pr : Lalaoui Lahouaoui	Université Mohamed Boudiaf M'sila	Encadreur
Dr : Abdenour hacine gharbi	Université Borj Bou Arreridj BBA	Co-Encadreur
Pr : Djouane Lotfi	Université Mostefa Benboulaid Batna 2	Examineur
Dr : Khalfa Ali	Université Mohamed Boudiaf M'sila	Examineur
Dr : Nabil Boukhenoufa	Université Mostefa Benboulaid Batna 2	Examineur

**Année universitaire :2024 / 2025**

## **Remerciements**

*Ce travail de thèse, présenté dans ce manuscrit, a été réalisé au sein du département d'électronique de la faculté de technologie de l'Université Mohamed Boudiaf - M'sila, sous la direction éclairée du Professeur **Lalaoui Lahouaoui**.*

*Je tiens tout d'abord à exprimer ma profonde gratitude au Professeur **Lalaoui Lahouaoui** et au Docteur **Abdenour Hacine Gharbi** (Co-Encadreur) de l'Université BORJ Bou Arreridj BBA, pour leur encadrement précieux et l'aide inestimable qu'ils m'ont apportée tout au long de cette recherche. Leur guidance et leurs conseils avisés ont été des piliers essentiels à la réalisation de ce travail.*

*Je remercie sincèrement le Président du jury, le Professeur **Laadjal Mohamed**, de l'Université Mohamed Boudiaf M'sila, pour l'honneur qu'il me fait en présidant ma soutenance.*

*Je suis également profondément reconnaissant envers les examinateurs de ma thèse, le Professeur **Djouane Lotfi** et le Docteur **Nabil Boukhenoufa** (Maître de Conférences A) de l'Université Mostefa BENBOULALD Batna 2, ainsi que le Docteur **Khalfa Ali** (Maître de Conférences A) de l'Université Mohamed Boudiaf M'sila, pour avoir accepté d'évaluer mon travail et pour leurs remarques constructives, qui m'ont permis d'enrichir ma réflexion et d'améliorer la qualité de mon travail.*

---

# *Sommaire*

---

## ***Sommaire***

*Remerciements*

*Liste des figures*

*Liste des tableaux*

*Résumé*

***Introduction générale..... 6***

### ***Chapitre I :***

#### ***Généralités sur les réseaux de neurones en classification d'images***

I.1 Introduction .....	11
I.2. Réseaux de neurones artificiels .....	11
I.2.1 Contexte de développement de RNA (artificial neural network) .....	11
I.2.2 Architecture des RNA.....	13
I.2.3. Présentation de quelques types de réseaux de neuronaux.....	13
I.2.3.1. Réseaux de neurones multicouches.....	13
I.2.3.2. Réseaux de neurones récurrents (ou bouclés) .....	14
I.2.3.3. Réseau de neurones convolutionnels.....	15
I.2.4 Processus et l'algorithme d'apprentissage d'un réseau de neurone.....	15
I.2.5. Propriétés des RNA.....	16
I.2.6. Domaines d'application des RNA.....	16
I.2.7. Avantages et inconvénients des RNA.....	18
I.3 Apprentissage profond (Deep Learning) des réseaux de neurones.....	18
I.3.1 Définition.....	18
I.3.2 Domaine d'application de l'apprentissage profonds : .....	20
I.3.3 Types d'apprentissage profonds .....	21
I.3.3.1 Apprentissage non supervisé : .....	22
I.3.3.2 Apprentissage semi-supervisé : .....	22
I.3.3.3 Apprentissage par renforcement.....	23
I.3.4. Techniques du deep learning : .....	23

I.4. Réseaux neuronaux convolutifs (CNN).....	24
I.4.1. Introduction au réseau de neurones convolutifs (CNN).....	24
I.4.2 Architecture des réseaux de neurones convolutifs (CONV).....	24
I.4.2.1. Principe d'architecture d'un CNN.....	25
I.4.2.2. Entraînement d'un nouveau CNN.....	27
I.4.3 Éléments importants d'un CNN (couches de CNN) .....	27
I.4.3.1 Couche de convolution (Convolution layer (CONV)) .....	28
I.4.3.2 Couche de pooling (Pooling layer (POOL)) .....	29
I.4.3.3 Couche de correction (Patch layer (RELU)) .....	30
I.4.3.4 Couche entièrement connectée (fully connected layer (FC)) .....	30
I.4.4 Avantage des CNNs.....	32
I.5. Conclusion.....	32
Références du chapitre I.....	34

## ***Chapitre II :***

### ***Modèles de Markov cachés HMM en classification d'images***

II.1 Introduction.....	38
II.2 Modèles de Markov cachés.....	38
II.2.1 Méthodes locales contextuelles.....	40
II.2.2 Champs de Markov cachés.....	41
II.2.3 Classification bayésienne avec les champs de Markov cachés.....	44
II.2.4 Chaînes de Markov cachées.....	44
II.2.5 Arbres de Markov cachés.....	46
II.2.6 Choix de modèle et de méthode.....	47
II.3 Modèles de Markov couple.....	48
II.4 Apprentissage des modèles de Markov.....	49
II.5 Extensions.....	50
II.6 Modèle probabiliste et classification bayésienne d'images .....	50
II.7 Avantages et limites de l'approche HMM pour la segmentation d'images.....	52
II.7.1 Avantages du HMM.....	52
II.7.2 Limites du HMM.....	52
II.8 Conclusion.....	53
Références du chapitre II.....	54

## **Chapitre III :**

### ***Méthode hybride HMM-CNN pour la classification d'images***

III.1 Introduction.....	57
III.2 Méthodes de classification d'images.....	57
III.2.1 CNN pour les méthodes de classification d'images.....	58
III.2.2 HMM pour les méthodes de classification d'images.....	59
III.3 Méthode hybride HMM-CNN.....	59
III.3.1 Avantages de HMM-CNN hybride.....	60
III.3.2 Limitations de HMM-CNN hybride.....	61
III.4 Architecture et algorithmes utilisés .....	61
III.4.1 Architecture.....	61
III.4.2 Algorithme.....	63
III.5 Méthode proposée pour la classification d'images.....	64
III.6 Performance du modèle.....	66
III.7 Conclusion.....	66
Références du chapitre III.....	68

## **Chapitre IV :**

### ***Implémentation et évaluation de CNN-HMM pour la classification d'images***

IV.1 Introduction.....	70
IV.2 Développement et application sur le jeu de données CIFAR-10.....	70
IV.3 Logiciels et outils.....	71
IV.3.1 Python.....	71
IV.3.2 Architecture du modèle CNNs.....	72
IV.3.3 Hardware .....	76
IV.4 Jeu de données (Data set) .....	76
IV.4.1 Section d'Entraînement.....	77
IV.4.2 Section ensemble de test.....	79
IV.4.3 Accuracy et perte du modèle.....	81
IV.4.4 Taux d'erreur.....	83
IV. 5 Section de Test.....	83

IV. 6 Évaluation métrique détaillée du modèle CNN-HMM sur un ensemble de données d'images.....	85
IV.7 Efficacité du modèle hybride CNN-HMM pour la classification d'images.....	88
IV. 8 Etude comparative entre le modèle proposé et d'autres modèles de classification.	89
IV. 9 Conclusion.....	90
Références de chapitre IV .....	91
<b>Conclusion générale</b> .....	92

---

## *Liste des figures*

---

## **Chapitre I :**

### **Généralités sur les réseaux de neurones en segmentation d'images**

Figure I-1 : Réseau multicouches (MLP).....	14
Figure I-2 : Réseau de neurones récurrent (bouclé).....	15
Figure I-3 : La relation entre l'intelligence artificielle et le ML et le deep Learning .....	19
Figure I-4 : Différence entre l'apprentissage profond et l'apprentissage machine traditionnel	20
Figure I-5 : Schéma synoptique des types d'apprentissage .....	21
Figure I-6 : Apprentissage non supervisé .....	22
Figure I-7 : Apprentissage supervisé .....	22
Figure I-8 : Architecture des Réseaux de Neurones convolutifs. (CNNs).....	25
Figure I-9 : Différentes couches de CNN .....	27
Figure I-10 : Processus de Couche convolutive .....	28
Figure I-11 : Exemple de produit de convolution .....	28
Figure I-12: Pooling Maximale et Pooling average.....	29
Figure I-13 : Exemple du Max Pooling .....	29
Figure I-14 : Aspect de la fonction ReLU.....	30
Figure I-15 : Application de ReLU à une image.....	30
Figure I-16 : Couche entièrement connectée (FC).....	31
Figure I-17 : Exemple de réseau convolutif deux étapes CONV-POOL suivies par deux couches FC.....	31

## **Chapitre II :**

### **Modèles de Markov cachés HMM en classification des images**

Figure II-1 : Modèles de Markov cachés en Classification d'images .....	40
Figure II-2 : Graphe d'orientation des dépendances pour la chaîne de Markov cachée .....	45
Figure II-3 : Représentation graphique d'un modèle de Markov observable.....	46

## **Chapitre III :**

### **Méthode Hybride HMM-CNN Pour la classification d'images**

Figure III.1 : Processus d'une approche hybride HMM-CNN.....	60
Figure III-1 : Algorithme globale de l'approche hybride CNN-HMM proposée.....	63
Figure III-2 : Diagramme de la méthode proposée .....	64

## Chapitre IV :

### Implémentation et évaluation de CNN-HMM pour la classification d'images

Figure IV.1 : Jeu de données CIFAR-10.....	73
Figure IV.2 : Un échantillon aléatoire de l'ensemble de données .....	77
Figure IV.3. Répartition des classes dans l'ensemble d'entraînement (training set) .....	78
Figure IV.4 : Répartition des classes dans le jeu de test.....	79
Figure IV.5 Accuracy du test et de perte du modèle.....	81
Figure IV.6. La matrice de confusion du modèle.....	82
Figure IV.7 : Le taux d'erreur du modèle.....	83
Figure IV.8 : Résultats de la section de test.....	84
Figure IV.9 : Une sélection aléatoire d'images de test.....	85
Figure IV.10 : Une sélection aléatoire d'images de test .....	85
Figure IV.11 : Prédictions correctes et incorrectes mises en évidence.....	88

---

## *Liste des tableaux*

---

## **Chapitre I :**

### **Généralités sur les réseaux de neurones en classification d'images**

Tableau I-1 : L'évolution des réseaux de neurones artificiels (ANN).....	12
Tableau I-2 : Exemples d'applications des ANN dans différents domaines.....	17
Tableau I.3 : Avantages et inconvénients des réseaux de neurones.....	18

## **Chapitre III :**

### **Méthode hybride HMM-CNN pour la classification d'images**

Tableau III-1 : Exemple représentatif d'une architecture CNN .....	62
--	----

## **Chapitre IV :**

### **Implémentation et évaluation de CNN-HMM pour la classification d'images**

Tableau IV-1 : L'architecture du modèle utilisé pour mon test .....	74
Tableau IV.2 : Le matériel utilisé pour exécuter les tests.....	76
Tableau IV.3 : Valeurs des critères d'exécution des tests.....	80
Tableau IV.4 : Comparaison de nos résultats avec d'autres études utilisant CIFAR-10 et divers types d'images.....	89

---

## *Résumé*

---

## Résumé

Cette thèse étudier l'amélioration la classification et segmentation des images par une méthode hybride CNN-HMM élaborée par l'intégration des réseaux de neurones convolutifs (CNNs) et des modèles de Markov cachés (HMMs).

Les modèles CNNs excellent dans l'extraction de données et de fonctionnalités spatiales à partir d'images, tandis que le modèles HMMs maîtrisent la modélisation de données séquentielles.

En combinant ces deux méthodologies, la recherche a pour objectif d'accroître la précision dans la classification et la segmentation des images.

Le modèle HMM-CNN par L'approche hybride proposée, tire parti des forces des CNN et des HMM pour saisir les dépendances spatiales et améliorer la performance de classification des ensembles de données de référence.

La thèse aborde les approches pour la classification et la segmentation des images, comme l'extraction de caractéristiques CNN et la formation HMM, ainsi que l'approche algorithmique pour le modèle hybride HMM-CNN.

Les résultats obtenus suggèrent que le modèle hybride proposée peut améliorer la précision et Augmenter la fiabilité de la classification et de la segmentation des images en combinant les capacités d'extraction de fonctionnalités des CNN avec la modélisation séquentielle des HMM. En fin de compte, la recherche démontre le potentiel de l'intégration des CNN et des HMM pour améliorer les tâches de classification des images, en particulier dans les applications où les dépendances temporelles sont critiques.

Les tests réalisés avec le modèle HMM-CNN proposée donner une précision de classification des images de 87,09%.

**Mot clés :** Apprentissage Profond, Réseaux de Neurones Convolutifs (CNNs), Modèles de Markov Cachés (HMMs), Modèle Hybride HMM-CNN, Classification et Segmentation des Images.

## Abstract

This thesis investigates the enhancement of image classification and segmentation through a hybrid CNN-HMM method developed by integrating Convolutional Neural Networks (CNNs) and Hidden Markov Models (HMMs).

CNN models excel in extracting spatial features from images, while HMM models specialize in modeling sequential data. By combining these two methodologies, the research aims to increase the precision of classifying and segmenting images.

The proposed hybrid CNN-HMM approach leverages the strengths of CNNs and HMMs to capture spatial dependencies and enhance the classification performance on benchmark datasets.

The thesis covers various approaches for image classification and segmentation, including CNN feature extraction and HMM training, as well as the algorithmic approach for the hybrid HMM-CNN model.

The obtained results suggest that the proposed hybrid model can improve the precision of image classification and segmentation by combining the feature extraction capabilities of CNNs with the sequential modeling of HMMs.

Ultimately, the research demonstrates the potential of integrating CNNs and HMMs to enhance image classification tasks, especially in applications where temporal dependencies are critical.

Tests conducted with the proposed HMM-CNN model yielded an image classification accuracy of 87.09%.

**Keywords:** Deep Learning (DL), Convolutional Neural Networks (CNNs), Hidden Markov Models (HMMs), Hybrid HMM-CNN Model, Image Classification and Segmentation.

## ملخص

تدرس هذه الأطروحة تحسين تصنيف الصور وتقسيمها باستخدام طريقة هجينة تجمع بين الشبكات العصبية التلافيفية (CNNs) ونماذج ماركوف المخفية (HMMs) تتفوق نماذج الشبكات العصبية التلافيفية في استخراج البيانات والميزات المكانية من الصور، بينما تتقن نماذج ماركوف المخفية نمذجة البيانات المتسلسلة. من خلال دمج هاتين المنهجيتين، تهدف الدراسة إلى زيادة دقة تصنيف الصور وتجزئتها.

يعتمد النموذج الهجين HMM-CNN المقترح على الاستفادة من نقاط القوة في كل من CNN و HMM لالتقاط التبعيات المكانية وتحسين أداء التصنيف في مجموعات البيانات المرجعية. تتناول الأطروحة الأساليب المستخدمة في تصنيف الصور وتقسيمها، مثل استخراج الميزات باستخدام CNN وتدريب HMM ، بالإضافة إلى النهج الخوارزمي للنموذج الهجين HMM-CNN. تشير النتائج المحققة إلى أن النموذج الهجين المقترح يمكنه تحسين الدقة وزيادة موثوقية تصنيف الصور وتقسيمها من خلال الجمع بين قدرات CNN في استخراج الميزات ونمذجة HMM التسلسلية. في النهاية، توضح الدراسة إمكانات دمج CNN و HMM في تحسين مهام تصنيف الصور، خاصة في التطبيقات التي تكون فيها التبعيات الزمنية ذات أهمية بالغة.

أظهرت الاختبارات التي أجريت باستخدام النموذج الهجين HMM-CNN دقة تصنيف للصور بلغت 87.09% .  
**الكلمات المفتاحية:** التعلم العميق، الشبكات العصبية التلافيفية (CNNs) ، نماذج ماركوف المخفية (HMMs) ، النموذج الهجين HMM-CNN، تصنيف الصور، تقسيم الصور.

---

## *Introduction générale*

---

## Introduction Générale

Le traitement d'images inclut une variété de techniques visant à modifier les images numériques pour améliorer leur qualité ou extraire des informations précieuses, en s'appuyant sur des transformations mathématiques telles que le filtrage, la classification, la reconnaissance, la segmentation et la restauration [1]-[5]. La classification d'images, une tâche cruciale et complexe du traitement d'images, consiste à diviser une image en différentes zones (régions) distinctes ayant des attributs similaires tels que l'intensité, la texture ou la couleur [6]. Il existe différentes méthodes de classification, notamment le seuillage, la détection de contours, la segmentation par régions, la classification d'images par réseaux neuronaux et le regroupement, chacune offrant des approches uniques pour améliorer l'analyse et la précision du traitement des images. Les avancées technologiques en matière de classification d'images contribuent non seulement à l'identification des objets mais améliorent également de manière significative la qualité et l'efficacité globale des systèmes de traitement d'images dans divers domaines d'application tels que la vision par ordinateur, l'imagerie médicale, la télédétection et les applications multimédia.

La classification des images représente l'un des défis les plus difficiles et complexes lorsque des pixels présentant des caractéristiques similaires sont regroupés en régions homogènes. De nombreuses tâches avancées de traitement, notamment l'extraction de fonctionnalités, la reconnaissance d'objets et le diagnostic médical, sont étroitement liées à la qualité des méthodes de classification utilisées.

Pour obtenir une classification appropriée, plusieurs critères doivent être satisfaits, tels que la compacité, la séparation et le chevauchement. Parfois, des connaissances de haut niveau sur la forme et l'apparence des objets sont nécessaires. Dans de nombreuses applications, cependant, ces informations ne sont pas disponibles ou peu pratiques à utiliser.

Les réseaux de neurones convolutifs (CNN) sont devenus essentiels dans les tâches de classification et de segmentation des images en raison de leur capacité à apprendre automatiquement les représentations hiérarchiques à partir de données de pixels bruts.

Les CNN excellent dans des domaines, notamment la reconnaissance d'images et de vidéos, l'analyse d'images médicales et le traitement du langage naturel, en utilisant des blocs de construction tels que le pooling, la convolution et des couches entièrement connectées.

Les progrès récents dans les méthodes de classification d'images basées sur CNN couvrent les architectures de réseau, les techniques de formation et les mesures d'évaluation, fournissant une compréhension complète de ce domaine pour les chercheurs et les praticiens.

De plus, les CNN se sont montrés prometteurs dans l'extraction de fonctionnalités pour améliorer la classification des images, avec des expériences démontrant des performances supérieures en utilisant des images couleur RVB sur des images binaires en termes de temps de simulation moyen et de précision. L'utilisation des CNN dans le traitement des images présente des perspectives d'application significatives, en particulier dans les tâches de la classification et de la segmentation des images médicales.

Les études des chercheurs soulignent collectivement l'efficacité des CNN dans les tâches de classification et de segmentation des images, mettant en évidence leur polyvalence et leurs performances dans diverses applications.

Les modèles de Markov cachés (HMMs) offrent des avantages significatifs dans la classification des images. Ils sont utilisés dans un large éventail d'applications en raison de leur capacité à modéliser efficacement les données de séries chronologiques.

Dans le domaine de la classification d'images à l'aide de modèles hybrides, les recherches récentes ont montré des avancées significatives grâce à l'intégration des approches d'apprentissage profond et des modèles statistiques. Par exemple, Pieczynski, 2003 a confirmé que les modèles de Markov sont des outils puissants pour le traitement d'images, en particulier dans des tâches non supervisées telles que la segmentation d'images naturelles [7]. De plus, Langrock et al., 2012 ont démontré que les modèles de Markov cachés (HMM) sont particulièrement adaptés à la modélisation de données séquentielles, notamment dans des applications comme le suivi animalier ou l'analyse de données environnementales [8]. De plus, Wang et al., 2017 ont présenté une étude détaillée sur l'utilisation des champs aléatoires de Markov pour la modélisation stochastique de données géologiques, mettant en évidence leur capacité à traiter des données multidimensionnelles complexes [9]. Dans un autre contexte, Dutordoir et al., 2019 ont exploré l'utilisation de processus gaussiens convolutionnels couplés aux réseaux neuronaux profonds pour la classification d'images, ce qui a permis de mieux capturer les relations non linéaires entre les caractéristiques extraites [10]. Enfin, Pham et al., 2020 ont proposé une méthode innovante combinant des champs aléatoires de Markov avec des réseaux neuronaux convolutifs (CNN) pour améliorer la précision de la segmentation d'images médicales du cerveau, soulignant ainsi l'importance des approches hybrides dans le traitement des données complexes [11].

Les modèles de Markov par couple (Couple Hidden Markov Models, CHMM), une extension des HMM, se sont révélés prometteurs en combinaison avec des réseaux de neurones profonds pour des tâches telles que la classification d'image non supervisée, surpassant les modèles probabilistes classiques.

En outre, des algorithmes de clustering basés sur HMM, tels que l'algorithme EM hiérarchique bayésien variationnel, ont été proposés pour déterminer automatiquement le nombre de clusters et d'états cachés, améliorant les performances de clustering par rapport aux méthodes traditionnelles. En outre, le cadre CHNM (Hidden Neural Markov Chain) introduit une nouvelle famille de modèles neuronaux séquentiels basés sur les HMM, présentant des résultats supérieurs dans les tâches de marquage de séquence par rapport aux modèles RNN classiques, indiquant le potentiel de ce nouveau cadre neuronal en compétition avec des modèles établis comme BiLSTM et BiGRU.

Le CNN-HMM hybride est une approche de classification d'images qui combine les avantages des réseaux de neurones convolutifs (CNN) et des modèles de Markov cachés (HMM). Les CNN sont particulièrement performants pour extraire des caractéristiques visuelles pertinentes des images, tandis que les HMM sont efficaces pour modéliser les dépendances temporelles ou séquentielles des données.

Dans le domaine de la classification et la segmentation d'images, l'approche le CNN-HMM Hybride exploite d'abord les capacités d'extraction (détection) de caractéristiques des CNN, puis utilise les HMM pour modéliser la dynamique temporelle ou la structure séquentielle des images, afin d'améliorer la performance de classification.

La thèse est structurée et organiser en quatre chapitres principaux, chacun explorant des aspects fondamentaux et essentiel de l'étude.

Le premier chapitre présente une introduction sur introduction sur les réseaux de neurones artificiels, couvrant leur historique, structure, architecture, ainsi que leurs avantages et inconvénients. Divers types de réseaux neuronaux, tels que les réseaux multicouches, récurrents et convolutifs (CNN), sont explorés, et présente les concepts de l'apprentissage profond, ses domaines d'application et techniques. Enfin, il décrit l'architecture des CNN, leurs avantages et différentes couches.

Le deuxième chapitre décrit les différents modèles de Markov cachés (HMM) utilisés dans le contexte de la classification d'images, Il commence par les modèles classiques, tel que les champs, chaînes et arbre ; Ensuite, il explore les modèles de Markov plus récents et avancés, notamment les modèles de Markov couples (CMM), modèles de Markov triplets (TMM) et modèles de Markov flous (FMM).

Dans le troisième chapitre, nous allons explorer la méthode hybride et le modèle proposée CNN-HMM qui combine les réseaux neuronaux à convolution (CNN) et les modèles de Markov cachés (HMM) pour améliorer la classification des images. Nous discuterons des avantages de cette approche hybride qui vise à tirer parti des capacités d'extraction de caractéristiques spatiales des CNN et des capacités de modélisation séquentielle des HMM.

Nous discuterons des limitations, ainsi que de l'architecture et des algorithmes utilisés dans cette approche.

Dans le quatrième chapitre, nous exposerons les résultats des tests réalisés en utilisant la méthode hybride CNN-HMM pour la classification des images. Nous discuterons des métriques d'évaluation comme l'exactitude, la précision, le rappel et le score F1 pour évaluer la performance du modèle hybride sur l'ensemble de test.

Finalement, nous clôturons ce mémoire par une conclusion et des perspectives

## Références d'introduction générale

- [1] Chetan Sharma and Advin Manhar, "Development of Image Processing Tools," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 3307, pp. 535–543, Jun. 2023.
- [2] L. Kumari, "International Journal for Multidisciplinary Research (IJFMR) A Study of AI based Technique in Image Processing," vol. 5, no. 2, pp. 1–6, 2023.
- [3] K Ganapathi Babu, Sunil Kumar Dasari, and C Yosepu, "An Overview: Image Processing Techniques and Its Applications," *Int. J. Eng. Technol. Manag. Sci.*, vol. 7, no. 3, pp. 883–888, 2023.
- [4] R. Goyal and Punam, "A Survey of Diverse Segmentation Methods in Image Processing," in *2022 IEEE International Conference on Current Development in Engineering and Technology (CCET)*, 2022, pp. 1–5.
- [5] J. Zhu, B. Sun, J. Cai, Y. Xu, F. Lu, and H. Ma, "Inspection and classification of wheat quality using image processing," *Qual. Assur. Saf. Crop. Foods*, vol. 15, no. 3, pp. 43–54, Jul. 2023.
- [6] E. ALKAN and A. AYDIN, "Image Processing Techniques based Feature Extraction for Insect Damage Areas," *Eur. J. For. Eng.*, vol. 9, no. 1, pp. 34–40, Jun. 2023.
- [7] W. Pieczynski, "Modèles de Markov en traitements d'images," *Traitement du Signal*, vol. 20, no. 3, pp. 255-278, 2003.
- [8] R. Langrock et al., "Modélisation de données séquentielles par HMM," *J. Pattern Recognition*, vol. 45, no. 8, pp. 2875-2887, 2012.
- [9] H. Wang et al., "Champs markoviens pour données géologiques," *IEEE Trans. Geoscience*, vol. 55, no. 4, pp. 2017-2032, 2017.
- [10] V. Dutoit et al., "Processus gaussiens convolutionnels," *NeurIPS*, pp. 10234-10244, 2019.
- [11] T.X. Pham et al., "Segmentation médicale hybride," *IEEE Trans. Med. Imaging*, vol. 39, no. 5, pp. 1541-1552, 2020.

---

# *Chapitre I*

## *Généralités sur les réseaux de neurones en classification d'images*

---

## I.1 Introduction

L'apprentissage automatique (ML), les réseaux de neurones artificiels (ANN) et l'apprentissage profond (DL) sont des sous-domaines de l'intelligence artificielle (IA) qui ont connu une croissance significative en termes de recherche et d'applications ces dernières années. Le ML se concentre sur le développement d'algorithmes capables d'apprendre des données et de faire des prédictions ou de prendre des décisions sans avoir été explicitement programmés. Les ANN, une classe spécifique de modèles au sein du ML, s'inspirent de la structure et du fonctionnement des réseaux neuronaux biologiques. L'apprentissage profond (DL), une sous-catégorie des ANN, repose sur des architectures multicouches appelées réseaux de neurones profonds, permettant de traiter des ensembles de données massifs et complexes, tout en capturant des représentations hiérarchiques des informations.

Dans ce chapitre, nous avons présenté une introduction aux réseaux de neurones artificiels. Nous avons commencé par un bref historique de leur développement, suivi d'une description de leur structure, de leur architecture, ainsi que de leurs avantages et inconvénients. Ce chapitre explore également divers types de réseaux neuronaux, notamment les réseaux multicouches, les réseaux récurrents et les réseaux convolutifs.

Nous avons ensuite introduit une vue d'ensemble sur l'apprentissage profond, en abordant ses domaines d'application, ses types et les techniques d'apprentissage utilisées. Enfin, une section spécifique est consacrée à l'étude des réseaux de neurones convolutifs (CNN), avec une exploration détaillée de leur architecture, de leurs avantages et des différentes couches qui les composent

## I.2. Réseaux de neurones artificiels

### I.2.1 Contexte de développement du réseau de neurones artificiels

La motivation des réseaux de neurones artificiels (ANN) réside dans leur capacité à imiter le traitement parallèle et distribué des systèmes biologiques. En 1986, McClelland et ses collaborateurs [1] ont développé une machine intelligente intégrant des principes d'intelligence artificielle. Cependant, la recherche de solutions aux problèmes complexes posait des défis majeurs dans ce modèle. Par conséquent, de nombreuses approches heuristiques ont été développées pour accomplir les tâches requises, tandis que des approches basées sur des règles ont été proposées pour résoudre les problèmes liés à la représentation des connaissances. Le tableau I.1 illustre l'évolution des ANN dans la littérature, offrant une meilleure compréhension de leur développement.

Les ANN s'inspirent du cerveau humain biologique, qui est composé d'environ 60 milliards de neurones interconnectés, collaborant pour réaliser des schémas complexes de prise de décision. Etablie sur cette idée fondamentale, la conception des réseaux de neurones artificiels commence par des neurones interconnectés simples, chacun fonctionnant comme un processeur élémentaire. Le concept de perceptron, introduit sur la base du modèle de neurone de McCulloch et Pitts [2], constitue une avancée clé dans ce domaine. Les fondements d'un ANN reposent sur une architecture comprenant une seule couche d'éléments : d'entrée, de traitement et de sortie. Ainsi, à partir d'un concept fondamental du cycle de traitement de l'information, les ANN ont évolué vers des formulations mathématiques complexes capables de produire des résultats optimaux pour divers ensembles de données ou segments de problèmes. Pour compléter un cycle de réseau, un neurone doit être testé de deux manières : via l'algorithme de propagation avant et celui de rétropropagation. Ce dernier, en particulier, a fait l'objet d'études approfondies et reste un domaine de recherche actif, tant par le passé qu'à l'heure actuelle [3].

Tableau I.1 : Evolution des réseaux de neurones artificiels (ANN)

Année	Auteurs	Modèle proposé
1943 [4]	Mcculloch et Pitts	Réseau Perceptron avec deux neurones artificiels.
1949 [5]	Héb	Règle d'apprentissage hébbien.
1958 [6]	Rosenblatt	Modèles de réseau Perceptron.
1960 [7]	Veuve et Hoff	Réseau neuronal Adaline.
1962 [8]	Veuve	Réseau neuronal Madaline.
1964 [9]	<b>Zadeh</b>	Logique floue.
1982 [10]	Hopfield	Réseau Hopfield (récurrent).
1986 [11]	Rumelhart et coll	Réseau neuronal de rétropropagation.
1988 [12]	Chang et Yang	Réseau de neurones cellulaires (la communication existe uniquement entre les neurones voisins).
1995 [13]	Cortés et Vapnik	Machine à vecteurs de support.
2002 [14]	Gerstner et Kistler	Réseau neuronal en pointe.
2012 [15]	Hinton	Réseau neuronal d'apprentissage profond.

---

## I.2.2. Architecture des réseaux de neurones

Dans une architecture de réseau neuronal typique, il existe trois principaux types de couches : les couches d'entrée, les couches cachées et les couches de sortie, chacune remplissant une fonction vitale dans le traitement de l'information et la génération de résultats.

Il est essentiel de comprendre les rôles et les interactions de ces couches pour comprendre comment les réseaux neuronaux fonctionnent et accomplissent les tâches qui leur sont confiées.

- **Couches d'entrée** : Il s'agit de la première couche du réseau, reçoit des données externes et les transmet aux couches cachées pour traitement. Chaque neurone de cette couche correspond à une caractéristique ou une variable d'entrée. Par exemple, dans le contexte de la reconnaissance d'images, chaque neurone peut représenter un pixel de l'image [16].

- **Couches cachées** : Ces couches sont qualifiées de "cachées" car elles ne sont pas directement visibles ou accessibles depuis l'extérieur du réseau. Sont des couches situées entre la couche d'entrée et la couche de sortie (couches intermédiaires), dans lesquelles le réseau apprend et extrait les caractéristiques des données d'entrée, jouent un rôle crucial dans la capacité du réseau à comprendre des modèles et des relations complexes au sein des données[17].

- **Couches de sortie** : C'est la dernière couche du réseau et elle produit les résultats finaux ou les prévisions sur la base des informations traitées provenant des couches cachées, fournissant la sortie souhaitée du réseau, pour finalement fournir les résultats finaux des calculs du réseau. Chaque neurone de la couche de sortie représente une classe ou une étiquette de sortie. Par exemple, dans le cas de la classification d'images, chaque neurone peut représenter une classe spécifique, comme "chien" ou "chat" [18].

La compréhension des fonctions et des interactions de ces couches est essentielle pour concevoir des réseaux neuronaux efficaces pour diverses applications, de la détection de maladies à l'aide de réseaux neuronaux avec des algorithmes génétiques à la modélisation de systèmes neuronaux pour les études du cortex visuel.

## I.2.3. Présentation de quelques types de réseaux de neurones

### I.2.3.1. Réseaux de neurones multicouches (MLP)

L'architecture du perceptron multicouches (MLP), se compose de plusieurs couches cachées où les neurones sont organisés selon une structure en couches avec des connexions uniquement entre les neurones des couches adjacentes, ce qui permet d'introduire la notion de direction du flux d'informations.

Cette disposition permet au réseau de capturer des relations complexes au sein d'ensembles de données en traitant les informations couche par couche, améliorant ainsi sa capacité à apprendre et à faire des prédictions. En outre, le concept de directionnalité dans le flux d'informations est crucial pour comprendre comment les réseaux de neurones traitent et transforment les données d'entrée en sorties significatives, soulignant l'importance du modèle de connectivité en couches pour faciliter des processus d'apprentissage et de prise de décision efficaces au sein de l'architecture MLP.[19][20], Comme illustré sur la figure I.1, la couche d'entrée correspond à l'ensemble des neurones d'entrée, tandis que la couche de sortie correspond à l'ensemble des neurones de sortie. Les couches intermédiaires sont appelées couches cachées.

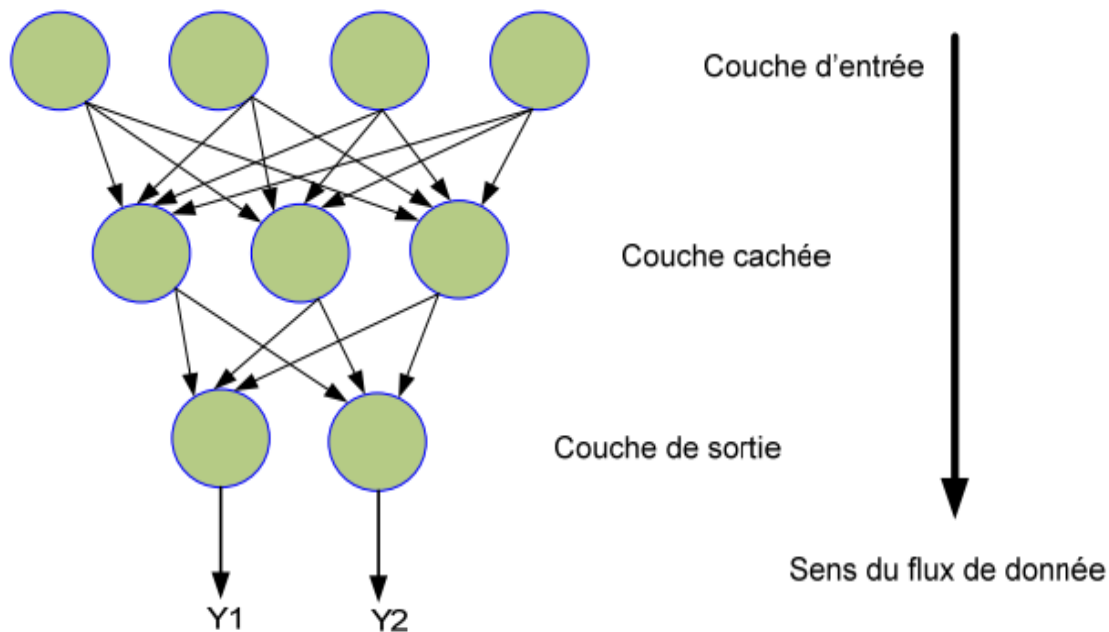


Figure I.1 : Réseau multicouches (MLP)

### I.2.3.2. Réseaux de neurones bouclés (ou récurrents)

Réseaux de neurones récurrents (RNN) sont des réseaux de neurones qui utilisent des connexions en boucle pour renvoyer des informations depuis le sens de propagation, leur permettant de traiter efficacement les données séquentielles.

RNN, peuvent conserver des informations au fil du temps, ce qui leur permet de détecter des relations temporelles dans les données d'entrée et de modéliser les résultats avec précision. Ces réseaux transmettent le résultat d'un pas temporel en tant qu'entrée au suivant, améliorant ainsi la capacité du réseau à apprendre et à faire des prédictions sur la base de données séquentielles.

L'utilisation de réseaux neuronaux récurrents a connu un succès significatif dans diverses applications, notamment la reconnaissance vocale, en raison de leur capacité à capturer efficacement les dépendances dans les données séquentielles [21], figure I.2.

Ces réseaux sont très efficaces car ils fonctionnent de manière séquentielle et présentent un comportement dynamique.

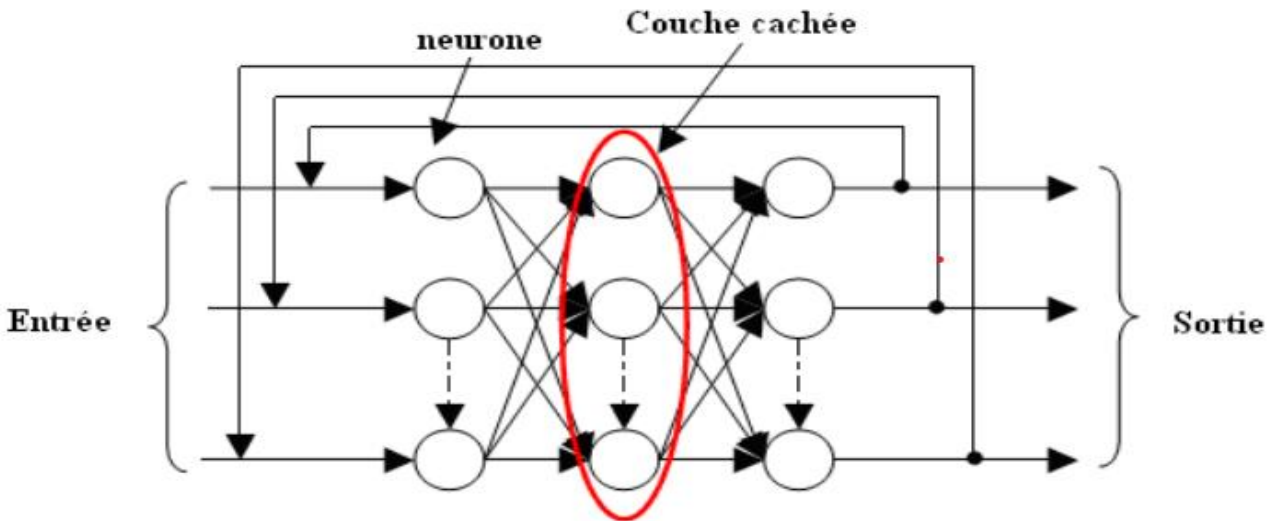


Figure I.2 : Réseau de neurones récurrent (bouclé).

### I.2.3.3. Réseau de Neurones convolutionnels

En revanche, les réseaux de neurones convolutifs (CNN) sont structurés avec la convolution, la mise en commun et des couches entièrement connectées, excellant dans l'apprentissage automatique des fonctionnalités à partir d'images avec des ressources de calcul élevées, ce qui les rend populaires pour diverses tâches comme la classification, la segmentation et le traitement du langage naturel [19].

Chaque architecture joue un rôle unique dans la conception de réseaux de neurones, avec les MLP se concentrant sur le flux d'informations d'alimentation, les réseaux récurrents incorporant des boucles de rétroaction pour le comportement dynamique, et les CNN spécialisés dans les tâches liées aux images grâce à l'extraction automatique des fonctionnalités.

### I.2.4. Processus et algorithme d'apprentissage d'un réseau de neurones

Le processus d'apprentissage dans les réseaux de neurones implique de modifier la structure interne, en ajustant spécifiquement les poids et les biais, afin d'obtenir le comportement souhaité et d'améliorer les performances pour des tâches telles que le contrôle ou l'identification de systèmes dynamiques [22].

---

Divers algorithmes d'apprentissage ont été développés, notamment des paradigmes d'apprentissage supervisé, non supervisé, hybride et par renforcement, pour optimiser les poids des réseaux de neurones en les adaptant pour qu'ils convergent vers des valeurs constantes et optimales.

De plus, des techniques d'apprentissage des réseaux neuronaux utilisant des autoencodeurs ont été proposées pour apprendre les paramètres des fonctions de densité de probabilité avec une grande précision, améliorant ainsi les performances du réseau grâce à des mises à jour des paramètres basées sur des valeurs d'évaluation calculées à l'aide de fonctions de coût [23].

En outre, il a été démontré qu'une méthode d'apprentissage des réseaux neuronaux impliquant la conversion de données de spectres de puissance logarithmiques et un entraînement sur la base de données cibles continues améliore la capacité d'apprentissage et les performances du réseau [15]. La dynamique des processus de déclenchement et d'apprentissage dans les systèmes neuronaux est étudiée afin de prédire avec précision les probabilités de déclenchement et les corrélations, avec des principes d'apprentissage minimisant l'énergie libre pour déterminer la plasticité des synapses biologiques et comprendre les structures émergentes dans les systèmes neuronaux [24].

Dans la plupart des algorithmes utilisés actuellement, les poids des connexions sont les variables qui sont ajustées pendant l'apprentissage.

### **I.2.5. Propriétés des réseaux de neurones**

L'intérêt porté aux réseaux de neurones tient sa justification dans les quelques propriétés fascinantes qu'ils possèdent, parmi lesquelles on peut citer [25].

- Parallélisme : Capacité des réseaux de neurones à traiter simultanément et de manière parallèle les données.
- Capacité d'adaptation : Aptitude des réseaux de neurones à apprendre et à s'adapter aux changements des données ou aux nouvelles contraintes.
- Mémoire distribuée : Stockage des informations dans les réseaux de neurones grâce à la répartition de l'activité des neurones.
- Capacité de généralisation : Capacité des réseaux de neurones à généraliser à partir d'exemples connus et à reconnaître des schémas non enseignés.
- Facilité de construction : Simplicité de création des réseaux de neurones et de développement d'applications grâce à la technologie informatique [25].

### **I.2.6. Domaines d'application des réseaux de neurones**

Les applications de réseaux neuronaux artificiels (ANN) ne se limitent pas à un domaine spécifique ; il y a une grande variété d'applications.

Un réseau de neurones artificiels a des possibilités d'applications diverses. Certaines applications sont présentées dans le tableau I.2. Le réseau de neurones artificiels a de multiples applications, mais un contenu non limité, et il peut couvrir tous les domaines. Il s'agit donc d'un domaine interdisciplinaire.

Tableau I.2 : Exemples d'applications des ANN dans différents domaines [26].

	<b>Domaine</b>	<b>Application</b>
ANN Réseau de neurones artificiels	Informatique et ingénierie	Traitement du langage naturel (NLP) Interface homme-machine Traitement d'image Réalité virtuelle. Exploration de données Reconnaissance de motifs Classification d'images et de motifs Identification de défauts
	Ingénierie électrique et électronique	Système de processus et de contrôle Intégration des systèmes Prévision de l'énergie et de la puissance
	Génie civil	Conception d'une structure optimale Sélection des matériaux et prise de décision
	Génie mécanique	Robotique Optimisation des processus Conception des matériaux
	Ingénierie aéronautique	Conception d'aéronefs Applications satellites et spatiales
	La science médicale	Classification, identification et diagnostic des maladies Système d'aide à la décision
	Génie de l'environnement	Prévision météorologique Prévision des précipitations
	Autres domaines	Prévision économique Prévision Forex

### I.2.7. Avantages et inconvénients des réseaux de neurones

Les réseaux neuronaux, tels que mis en évidence dans le tableau, présentent divers avantages et inconvénients.

Malgré ces limites, les réseaux neuronaux sont de plus en plus utilisés dans divers domaines tels que la médecine, la linguistique et la bibliométrie, démontrant ainsi leur potentiel pour améliorer la productivité de la recherche et résoudre des tâches laborieuses de traitement des données.

En comprenant à la fois les forces et les faiblesses des réseaux neuronaux, les chercheurs peuvent tirer parti de leurs avantages tout en atténuant leurs limites pour des résultats optimaux dans diverses applications.

Tableau I.3 : Avantages et inconvénients des réseaux de neurones.

Avantages	Inconvénients
Limitation des fonctions cérébrales humaines	Risque de surajustement
Résolution de tâches complexes (reconnaissance d'écriture manuscrite, diagnostic médical)	Manque d'explicabilité
Adaptabilité	Consommation élevée de ressources informatiques.
Capacités de généralisation	Difficulté d'interprétation
Tolérance élevée au bruit	Vulnérabilités aux attaques adverses
Efficacité dans divers domaines (commerce, finance, médecine, ingénierie alimentaire)	Apprentissage de mappages entrées-sorties discontinus
Apprentissage efficace de modèles et de relations complexes (détection de la maladie par logiciel d'analyse d'images)	Limité pour le test d'hypothèse en sciences sociales

## I.3 Apprentissage profond (deep learning) des réseaux de neurones

### I.3.1 Définition

L'apprentissage profond, un sous-ensemble de l'apprentissage automatique, utilise des réseaux de neurones profonds inspirés de la structure du cerveau humain pour apprendre des relations complexes entre les données. Cette approche a attiré l'attention pour ses performances remarquables, en particulier dans les tâches de classification d'images, en tirant parti d'algorithmes de réseaux neuronaux qui imitent les fonctions cérébrales.

L'apprentissage profond, une forme d'intelligence artificielle issue de l'apprentissage automatique, permet aux machines d'améliorer leurs capacités en fonction des volumes de données disponibles. Les progrès en matière d'apprentissage profond ont été facilités par la puissance de calcul accrue des ordinateurs et la disponibilité de vastes ensembles de données [27].

L'apprentissage profond, repose largement sur des réseaux neuronaux artificiels comportant plusieurs couches de nœuds interconnectés, appelés neurones, pour traiter des données complexes telles que les images et la voix.

Ces réseaux neuronaux profonds, caractérisés par leur profondeur, décomposent hiérarchiquement le contenu des données d'entrée en couches successives de neurones, les résultats d'une couche servant d'entrées pour les calculs de la couche suivante.

Ce traitement hiérarchique permet au réseau d'extraire des caractéristiques et des modèles complexes des données, permettant ainsi des tâches telles que la reconnaissance d'images et le traitement de la parole.

En tirant parti de ces couches interconnectées de neurones, les modèles d'apprentissage profond peuvent classer et identifier les éléments des données, par exemple en associant des balises descriptives à d'images ou en reconnaissant des mots dans des entrées vocales [28].

La figure I.3 illustre la relation entre trois concepts : intelligence artificielle (IA), apprentissage automatique (ML) et apprentissage profond (DL).

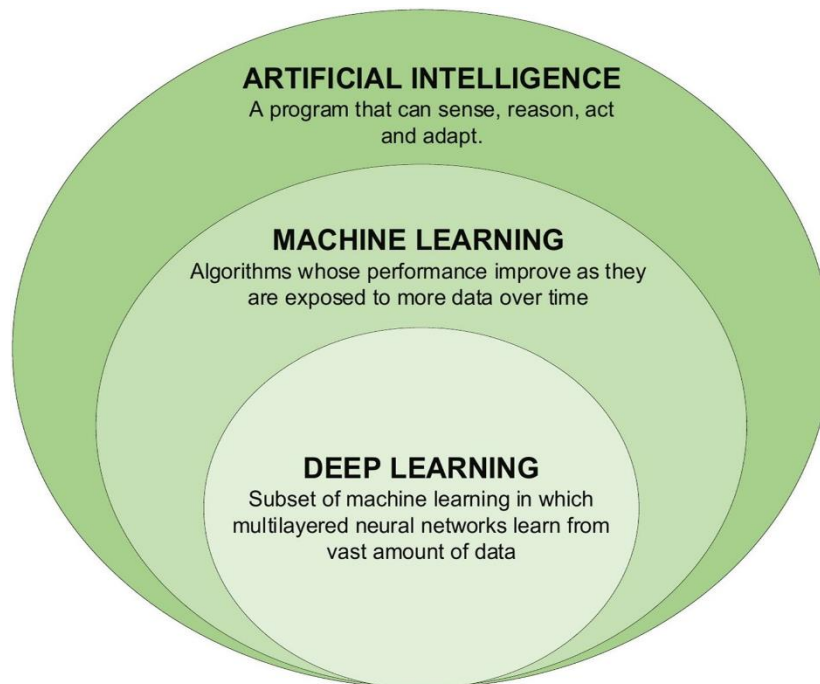


Figure I.3 : Relation entre l'intelligence artificielle, l'apprentissage automatique et l'apprentissage profond.

La figure I.4 représente la différence entre l'apprentissage profond (deep learning (DL)) et l'apprentissage machine traditionnel (machine learning (ML))

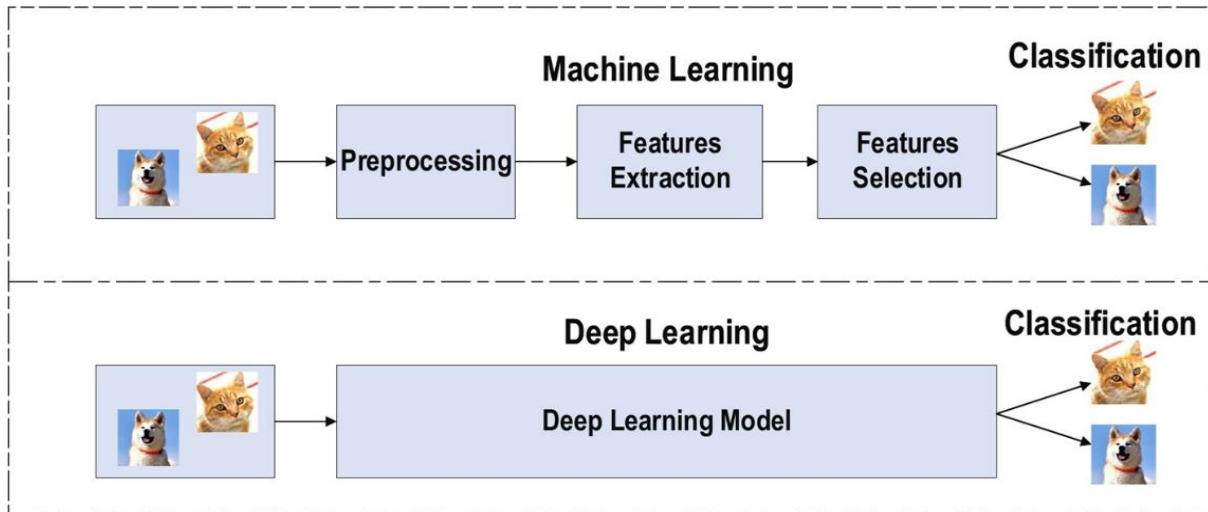


Figure I.4 : Différence entre l'apprentissage profond et l'apprentissage machine traditionnel.

### I.3.2 Domaine d'application du deep learning

L'apprentissage profond, est largement appliqué dans divers domaines tels que la vision par ordinateur, le traitement du langage naturel, la robotique, les soins de santé et l'Internet des objets (IoT).

Cette technologie s'appuie sur des réseaux neuronaux profonds pour extraire des motifs et des représentations complexes à partir de vastes ensembles de données, ce qui entraîne des avancées significatives dans les domaines de la reconnaissance d'objets, de la compréhension de scènes, de la production d'images, de la traduction automatique, de l'analyse des sentiments, des systèmes de dialogue, etc.

Les architectures d'apprentissage profond, notamment les réseaux neuronaux convolutifs (CNN) et les réseaux neuronaux récurrents (RNN), jouent un rôle crucial dans la gestion d'ensembles de données complexes et la résolution de problèmes complexes dans divers domaines, tels que la bio-informatique, l'analyse des mégadonnées et les soins de santé.

L'impact transformateur de l'apprentissage profond est évident dans sa capacité à stimuler la créativité, à favoriser les avancées et à ouvrir la voie au développement de systèmes d'IA plus intelligents dans de nombreux secteurs [29].

En vision par ordinateur, les réseaux neuronaux convolutifs (CNN) sont largement utilisés pour le traitement d'images en raison de leurs performances supérieures en extraction de caractéristiques et en ajustement de modèles [30].

Dans le domaine de la santé, les techniques d'apprentissage profond révolutionnent la découverte de médicaments, l'imagerie médicale et les dossiers médicaux électroniques, en mettant l'accent sur la lutte contre les épidémies comme la COVID-19 grâce à des prédictions rapides et à une analyse de données efficace [30].

De plus, l'intégration de réseaux neuronaux profonds avec les appareils IoT permet des applications en temps réel et des interactions entre les individus et leurs environnements physiques, élargissant ainsi les capacités des technologies IoT dans les domaines scientifiques et de l'ingénierie.

La polyvalence et l'adaptabilité de l'apprentissage profond en font un outil puissant ayant des implications de grande envergure dans divers domaines, améliorant l'efficacité, la rentabilité et la qualité de vie globale [31].

### I.3.3 Types d'apprentissage profonds

En fonction du niveau de supervision utilisé, les techniques de l'apprentissage profond sont classées en trois grandes catégories : Non supervisées, partiellement supervisées (semi-supervisées) et supervisées. De plus, l'apprentissage par renforcement profond, également connu sous le nom de RL, est un autre type de technique d'apprentissage, qui est principalement considéré comme appartenant à la catégorie des techniques d'apprentissage partiellement supervisées (et parfois non supervisées) [32][33].

La figure I.5 illustre le schéma synoptique des différents types d'apprentissage profond.

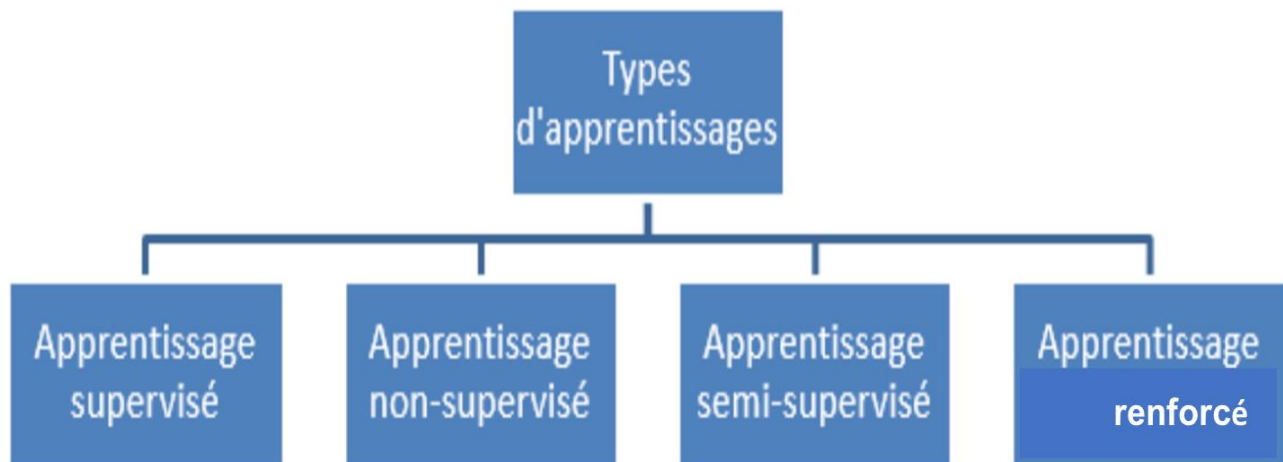


Figure I.5 : Schéma synoptique des types d'apprentissage.

**I.3.3.1 Apprentissage non supervisé (unsupervised learning)**

Dans l'apprentissage non supervisé ou auto-organisation, les poids du réseau sont ajustés en fonction de critères internes, tels que l'activation des connexions entre les cellules neuronales, ce qui entraîne un comportement similaire aux techniques de classification non supervisée.

Le principe fondamental de ce type d'apprentissage est d'organiser les entrées présentées au système en catégories ou groupes partageant des caractéristiques communes, en utilisant des mesures de similarité pour distinguer ces catégories [34].

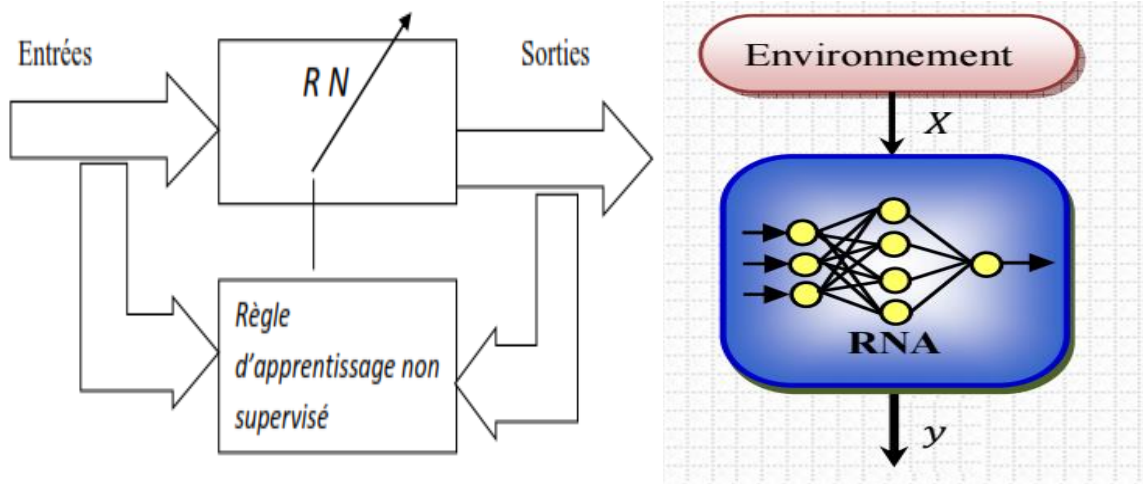


Figure I.6 : Apprentissage non supervisé.

**I.3.3.2 Apprentissage supervisé (supervised learning)**

Dans l'apprentissage supervisé, la sortie souhaitée ou correcte du réseau pour chaque entrée donnée est connue à l'avance. L'apprentissage du réseau consiste à mesurer l'erreur entre la sortie réelle (comportement actuel) et la sortie souhaitée (comportement attendu), puis à ajuster les poids du réseau pour réduire cette erreur [35].

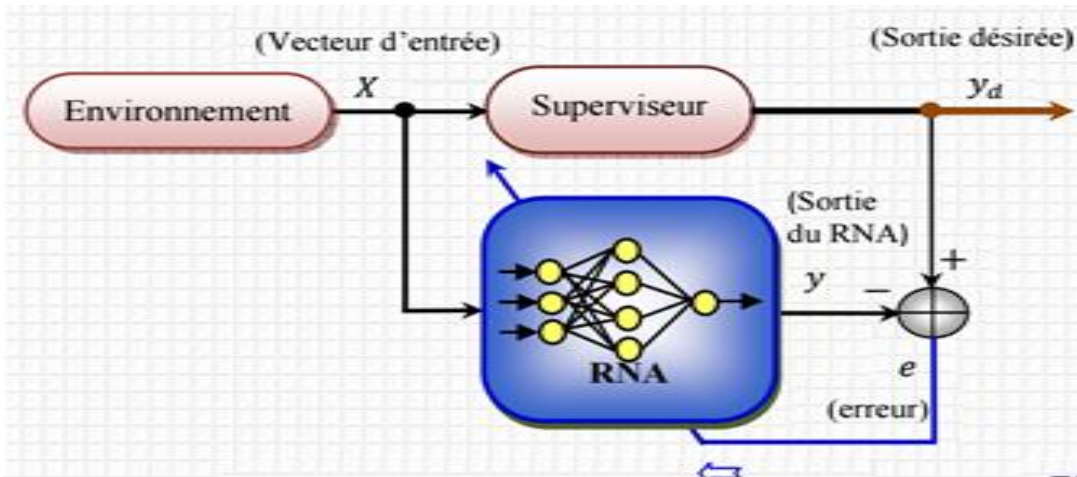


Figure I.7 : Apprentissage supervisé

### I.3.3.3 Apprentissage par renforcement

L'apprentissage par renforcement implique un réseau neuronal interagissant avec un environnement où des récompenses sont accordées pour les réponses souhaitables du réseau et des pénalités pour les réponses insatisfaisantes, dans le but de découvrir des actions qui maximisent les récompenses.

Le réseau apprend les politiques optimales par essais et erreurs, en ajustant ses actions en fonction des commentaires reçus de l'environnement. Dans les environnements partiellement observables, les réseaux de neurones récurrents approximent les fonctions de valeur en filtrant la distribution de probabilité postérieure de l'état actuel en fonction de l'historique, améliorant ainsi la capacité du réseau à prendre des décisions éclairées. De plus, lorsque la dynamique de l'environnement change, des méthodes telles que la dynamique inverse peut aider à déduire ces changements et à ajuster les actions du réseau en conséquence afin de maintenir des performances optimales.

Dans l'ensemble, l'apprentissage par renforcement constitue un cadre fondamental permettant aux agents d'apprendre des comportements et de prendre des décisions dans divers contextes, avec des applications allant du jeu au contrôle de robots.

### I.3.4. Techniques de l'apprentissage profond

Les techniques de l'apprentissage profond, sont un ensemble d'outils et de méthodes utilisés dans le secteur de l'intelligence artificielle afin d'analyser et de comprendre des données complexes. Ces techniques comprennent plusieurs modèles et réseaux neuronaux avancés qui nous permettent d'extraire des informations précieuses et de prendre des décisions intelligentes [36].

Parmi ces techniques importantes :

- Les réseaux neuronaux convolutifs (CNN)
- Les réseaux neuronaux récurrents (RNN)
- Les réseaux neuronaux à long terme et court terme (LSTM)
- Radial basis function network (RBFN)
- Les réseaux antagonistes génératifs (GAN)
- K plus proches voisins (KNN)
- Les réseaux de neurones artificiels (ANN)

En utilisant ces techniques, nous pouvons améliorer les performances des systèmes intelligents et des applications d'intelligence artificielle dans des domaines variés tels que la reconnaissance vocale, la traduction automatique, le diagnostic médical, l'analyse de données [36].

---

Dans la section suivante, nous avons approfondi notre étude sur la technique des réseaux neuronaux convolutifs (CNN). Les CNN sont des modèles d'apprentissage profond spécialement conçus pour l'analyse et le traitement d'images. Ils sont couramment utilisés dans des domaines tels que la vision informatique, l'identification d'objets et la segmentation d'images.

#### **I.4. Réseaux neuronaux convolutifs (CNN)**

##### **I.4.1. Introduction au réseau de neurones convolutifs (CNN)**

L'intelligence artificielle (IA) est devenue la pierre angulaire de la technologie moderne, révolutionnant les industries de la santé à la finance. Les réseaux de neurones convolutifs (CNN), une branche de l'intelligence artificielle, se sont imposés comme un outil performant pour une multitude d'applications, telles que la reconnaissance d'images, la reconnaissance vocale et le traitement du langage naturel (PNL) et même dans le domaine de la génomique, où ils ont été utilisés pour classer les séquences d'ADN [37][38].

Pour résoudre des problèmes de classification d'objets à partir d'images, on utilisera une variante spécifique de réseau neuronal appelée réseau neuronal convolutif, également connu sous le nom de CNN (Convolutional Neural Network). Les réseaux neuronaux convolutifs (CNN) représentent l'un des principaux types de réseaux neuronaux employés pour la reconnaissance, la classification et la segmentation d'images. Les CNN ont plusieurs utilisations, dont certaines sont la reconnaissance d'objets, le traitement d'images, la vision par ordinateur et la reconnaissance faciale. L'entrée pour les réseaux de neurones convolutifs est fournie par d'images. Les réseaux de neurones convolutifs sont utilisés pour apprendre automatiquement une hiérarchie de caractéristiques qui peuvent ensuite être utilisées pour la classification, par opposition à la création manuelle de caractéristiques. Pour ce faire, une hiérarchie de cartes de caractéristiques est construite en convoluant itérativement l'image d'entrée avec des filtres appris. En raison de la méthode hiérarchique, les couches supérieures peuvent apprendre des caractéristiques plus complexes qui sont également invariantes en distorsion et en traduction [39][40].

##### **I.4.2 Architecture des réseaux de neurones convolutifs (CONV)**

Lorsqu'on discute des architectures couramment utilisées pour les réseaux convolutifs, la majorité des architectures CNN adhèrent aux mêmes principes de conception fondamentaux, qui impliquent l'application séquentielle des couches convolutives à l'entrée, tout en réduisant périodiquement les dimensions spatiales et en augmentant le nombre de cartes de caractéristiques [39].

Alors que les architectures de réseau traditionnelles étaient principalement composées de couches convolutives empilées, les architectures modernes explorent des approches novatrices pour créer et construire des couches convolutives favorisant un apprentissage plus efficace. Ces architectures modernes se basent généralement sur une unité de base reproductible qui est utilisée dans tout le réseau.

Ces architectures servent de directives de conception globales que les praticiens de l'apprentissage automatique adaptent ensuite pour s'attaquer à diverses tâches de vision par ordinateur. Ils agissent comme de puissants extracteurs de fonctionnalités adaptés à la classification d'images, à la détection d'objets, à la segmentation d'images et à de nombreuses autres tâches avancées. J'aborderai brièvement certaines de ces architectures plus tard.

La forme la plus répandue d'une architecture CNN consiste à empiler quelques couches CONVRELU, suivies de couches de POOL, et à répéter ce motif jusqu'à ce que l'image ait été fusionnée spatialement en une taille plus petite. Il est courant de passer à des couches entièrement connectées à un certain stade. La dernière couche entièrement connectée contient la sortie, telle que les scores de classe [39][41].

La figure I.8 montre l'architecture réseau de CNN.

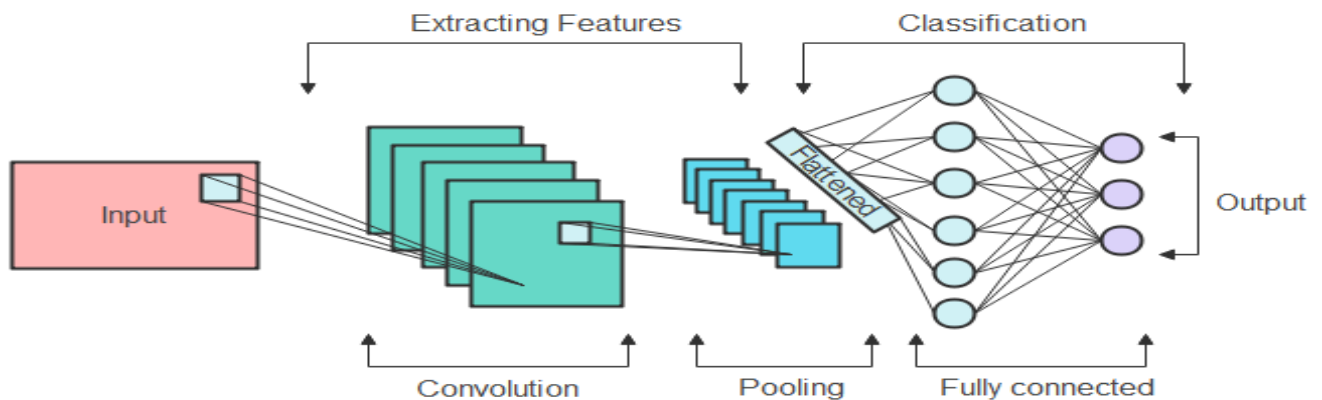


Figure I.8 : Architecture des réseaux de neurones convolutifs (CNNs).

#### I.4.2.1. Principe d'architecture d'un CNN

Un réseau neuronal convolutif (CNN) est un type spécialisé de réseau neuronal profond conçu pour imiter le fonctionnement du cortex visuel dans le cerveau pour des tâches telles que la reconnaissance d'images, la classification et la détection d'objets. Les CNN excellent dans le traitement d'images grâce à leur architecture unique, qui comprend des couches convolutives, des couches de regroupement et des couches entièrement connectées, inspirée de l'organisation du cortex visuel.

---

En utilisant des fonctionnalités telles que les pondérations partagées, les connexions locales et la mise en commun, les CNN réduisent la complexité, augmentent l'invariance en termes d'échelle et de distorsion, et améliorent la robustesse et la tolérance aux pannes dans les tâches de traitement d'image. La conception des CNN s'aligne étroitement sur les principes structurels et fonctionnels du système visuel des primates, ce qui permet de comparer les réseaux artificiels et leurs homologues biologiques afin d'explorer comment les fonctions visuelles et les représentations neuronales émergent dans le cerveau à partir des principes informatiques.

Les CNN se composent de deux parties principales : la couche d'entrée, où les images sous forme de matrices de pixels bidimensionnelles (pour les images en noir et blanc) ou de tableaux tridimensionnels (pour les images en couleur avec des canaux rouge, vert et bleu) sont introduites dans le réseau pour traitement.

Dans un réseau neuronal convolutif (CNN), l'étape initiale implique la partie convolutionnelle, où le réseau extrait les caractéristiques de l'image par le biais d'opérations de convolution. Ce processus implique de faire glisser un filtre sur de petites sections de l'image pour générer des cartes d'activation, appelées cartes de convolution.

Les couches convolutives, qui sont des composants fondamentaux des CNN, appliquent des filtres aux images d'entrée afin d'identifier des caractéristiques par le biais de circonvolutions mathématiques. Ces couches jouent un rôle crucial dans l'extraction des caractéristiques et sont essentielles pour que le réseau puisse apprendre et reconnaître des modèles dans les données d'entrée. En utilisant des couches convolutives, les CNN peuvent capturer efficacement les hiérarchies spatiales des caractéristiques d'images, ce qui leur permet d'effectuer des tâches telles que la classification d'images, la segmentation et la détection d'objets avec une précision et une efficacité élevée.

Dans les réseaux de neurones convolutifs (CNN), les filtres intermédiaires peuvent effectivement réduire la résolution de l'image [17]. Les cartes de caractéristiques résultant de ces filtres sont ensuite généralement aplaties dans un vecteur de caractéristiques servant d'entrée pour les couches entièrement connectées.

La fonction principale des couches entièrement connectées est de fusionner les caractéristiques extraites du vecteur d'entrée pour faciliter la classification d'images. La sortie finale d'un CNN est constituée de neurones, chaque neurone représentant une classe spécifique, et les valeurs de sortie sont généralement normalisées entre 0 et 1. Cette normalisation facilite l'interprétation de la sortie sous forme de probabilités, ce qui permet de classer les images en fonction de la probabilité la plus élevée attribuée à une classe particulière [42].

### I.4.2.2. Entraînement d'un nouveau CNN

La création d'un réseau de neurones convolutifs est une tâche exigeante en termes de compétence, de matériel et de quantité de données nécessaires. La première étape consiste à établir l'architecture du réseau, y compris le nombre de couches, leur taille et les opérations matricielles qui les relient, puis l'entraînement consiste à optimiser les paramètres du réseau pour réduire l'erreur de classification en sortie. Le temps d'exécution des meilleurs réseaux CNN peut prendre plusieurs jours, car les unités de traitement graphique (GPU) entraînent des centaines de milliers d'images [19].

### I.4.3 Éléments importants d'un CNN (couches de CNN)

Dans cette partie, nous discutons des composants fondamentaux d'un CNN (couches de CNN) et de leur rôle dans l'ensemble de l'architecture.

Un CNN est généralement composé d'un certain nombre de couches (ou blocs multicouches, comme on les appelle) ; chacune avec une fonction spécifique dans le réseau, est décrite en détail ci-dessous :

- Couche de convolution (CONV)
- Couche Pooling
- Couche entièrement connectée (FC)
- Couche de correction (ReLU)

La figure I.9 montre différentes couches d'un réseau de CNN.

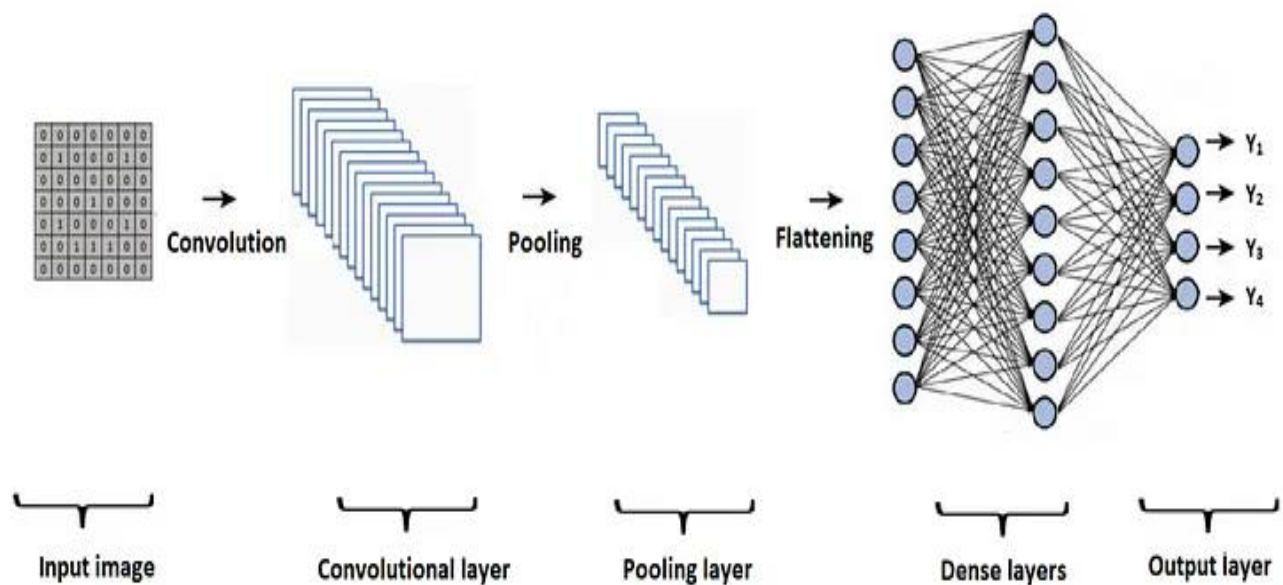


Figure I.9 : Différentes couches de CNN.

I.4.3.1 Couche de convolution (CONV)

La couche convolutive (couche conv) est un composant fondamental d'un CNN. Il exploite la nature stationnaire d'images, où un motif dans une région peut être trouvé dans une autre région. Pour traiter de grandes images, nous extrayons de petites sections et les glissons sur toute l'image (entrée). A chaque position, ces sections sont convoluées en une seule position de sortie. Ces petites sections sont appelées filtres ou noyaux, qui sont ajustés à l'aide de la rétropropagation.

Les filtres divisent l'image en grilles plus petites et les convolent avec les noyaux pour produire une carte des caractéristiques. La formule mathématique suivante est utilisée pour déterminer les valeurs de la carte de caractéristiques suivante, où le noyau est représenté par  $h$  et l'entrée de l'image est indiquée par  $f$ . Les indices de lignes et de colonnes de la matrice résultante sont représentés par  $m$  et  $n$ . La couche convolutive est cruciale dans les CNN car elle consomme la majorité du temps du réseau, extrait des informations spatiales locales et effectue des tâches intensives en calcul [42][46].

La figure I.10 illustre l'opération de convolution.

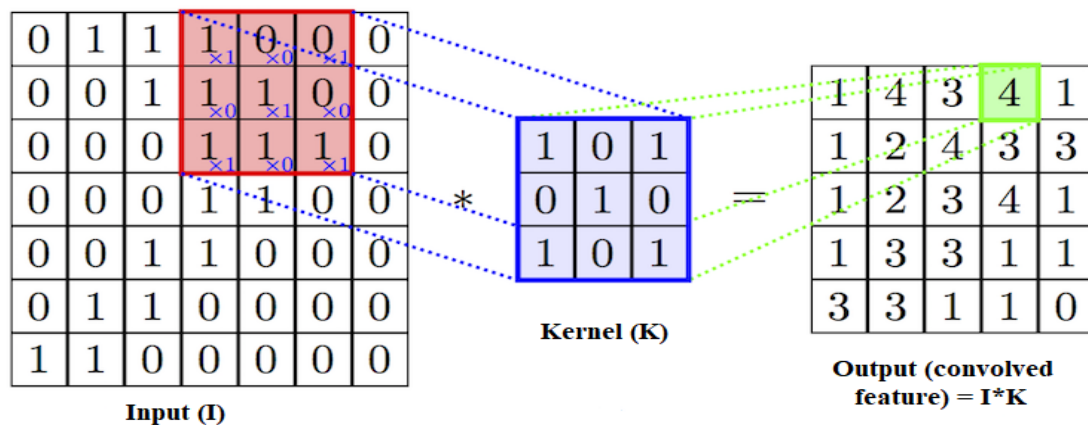


Figure I.10 : Processus de couche convolutive.

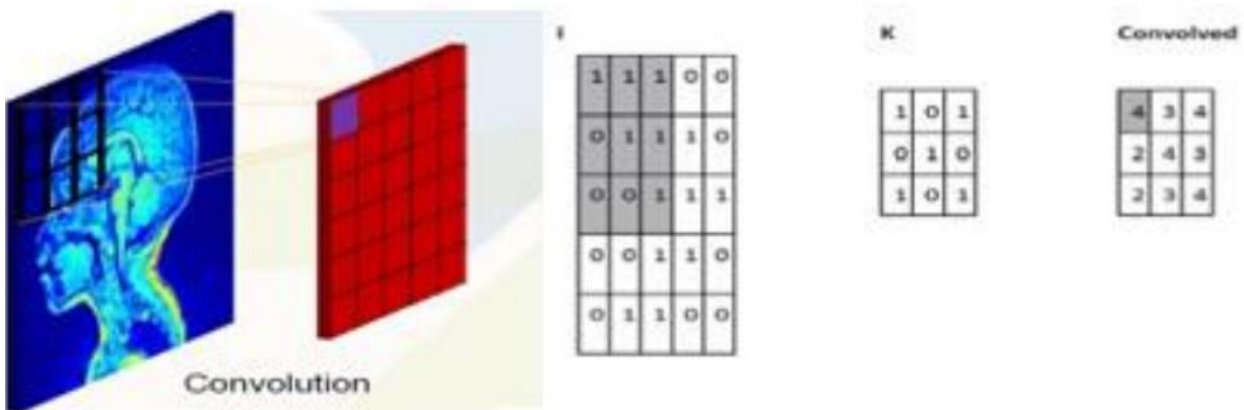


Figure I.11: Exemple de produit de convolution

I.4.3.2 Couche de pooling (POOL)

Le pooling est un processus de réduction de la taille d'une image dans un CNN. Cela implique de prendre une petite région de la sortie convolutive et de la réduire à une seule valeur de sortie. Diverses techniques de pooling existent, notamment le max pooling, le moyenne pooling et le average pooling.

**(a) Max pooling :** Est utilisée lorsque la valeur maximale est souhaitée pour chaque patch sur la carte des fonctionnalités [43].

Le max pooling sélectionne la valeur de pixel la plus élevée dans une région. Le pooling réduit le nombre de paramètres à calculer et introduit une invariance à la translation dans le réseau, le rendant insensible au changement de forme, de taille et d'échelle. Cette couche est positionnée après la couche ReLU et avant la couche de convolution. Le pooling favorise l'invariance translationnelle, renforce la robustesse des caractéristiques de sortie contre les changements de leur emplacement dans l'image et aide à prévenir le surajustement [18].

**(b) Average pooling :** Cette option est utilisée lorsque la valeur moyenne est souhaitée pour chaque patch sur la carte des caractéristiques [19][45]. La figure I.12, illustre le fonctionnement de la mise en commun moyenne et maximale.

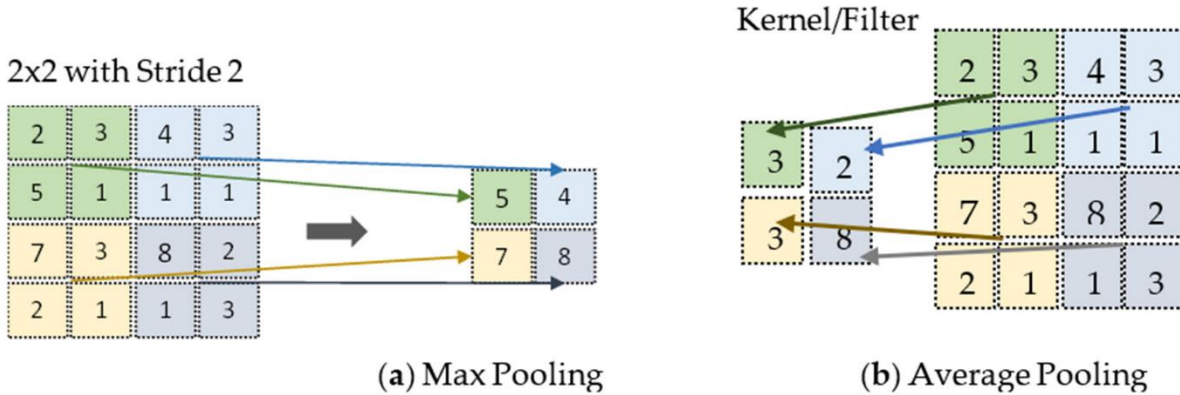


Figure I.12. Pooling maximale et pooling average.

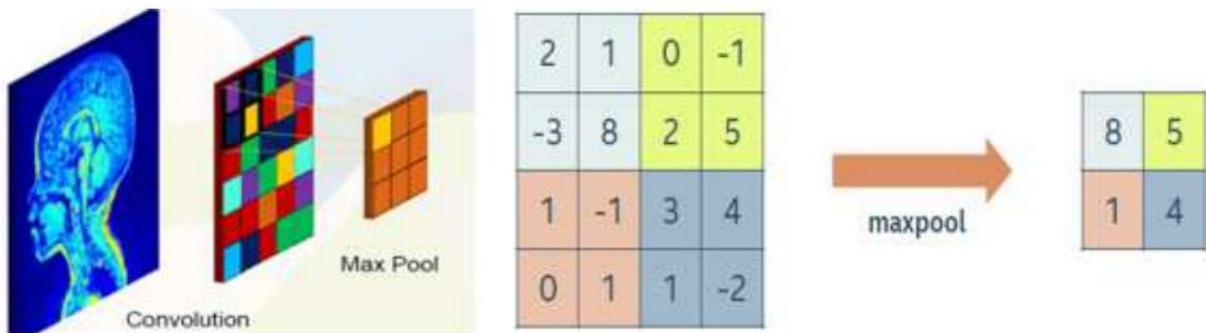


Figure I.13 : Exemple du max pooling

**I.4.3.3 Couche de correction (ReLU)**

La fonction ReLU (Rectified Linear Units), appelée aussi "fonction d'activation non saturante", est une fonction réelle non linéaire définie par l'équation suivante [46]:

**La correction  $\text{ReLU}(x)=\max(0, x)$ .**

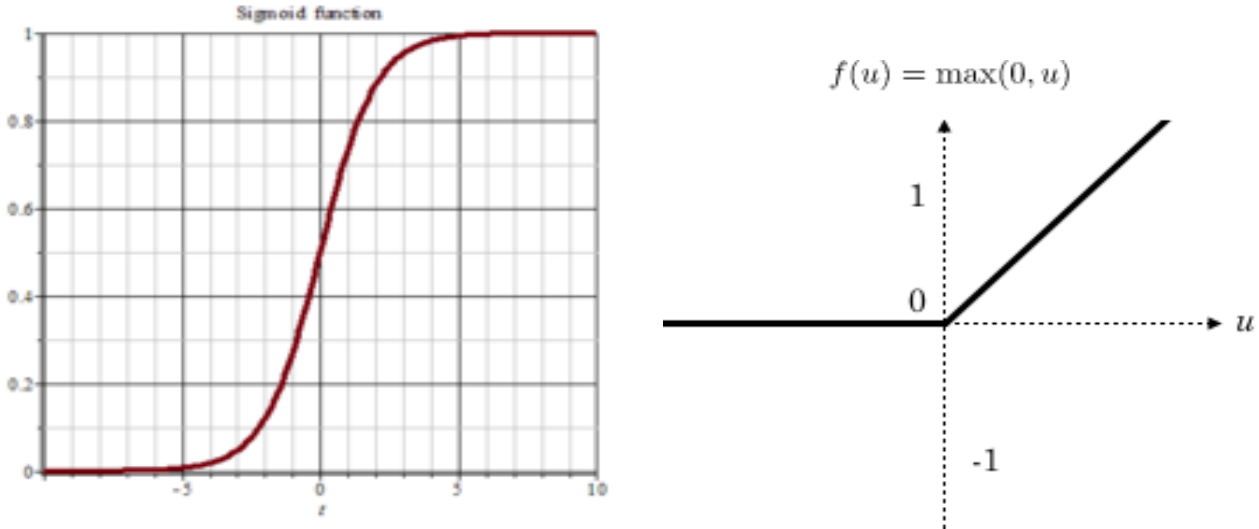


Figure I.14 : Aspect de la fonction ReLU

La couche de correction ReLU agit en remplaçant toutes les valeurs négatives reçues en entrée par des zéros, ce qui lui confère un rôle de fonction d'activation [46].



Figure I.15 : Application de ReLU à une image.

**I.4.3.4 Couche entièrement connectée « Fully Connected » (FC)**

Cette couche joue un rôle crucial dans la réalisation d'une analyse complète de la sortie de toutes les couches précédentes du CNN.

Il est chargé d'intégrer les entrées de toutes les couches d'extraction des caractéristiques et de procéder à une analyse globale des données. Cette couche est entièrement connectée, ce qui signifie qu'elle est connectée à toutes les activations des couches précédentes [19], comme le montre la figure I.16.

Il prend l'entrée de tous les neurones de la couche précédente et effectue des opérations avec des neurones individuels de la couche actuelle pour générer la sortie.

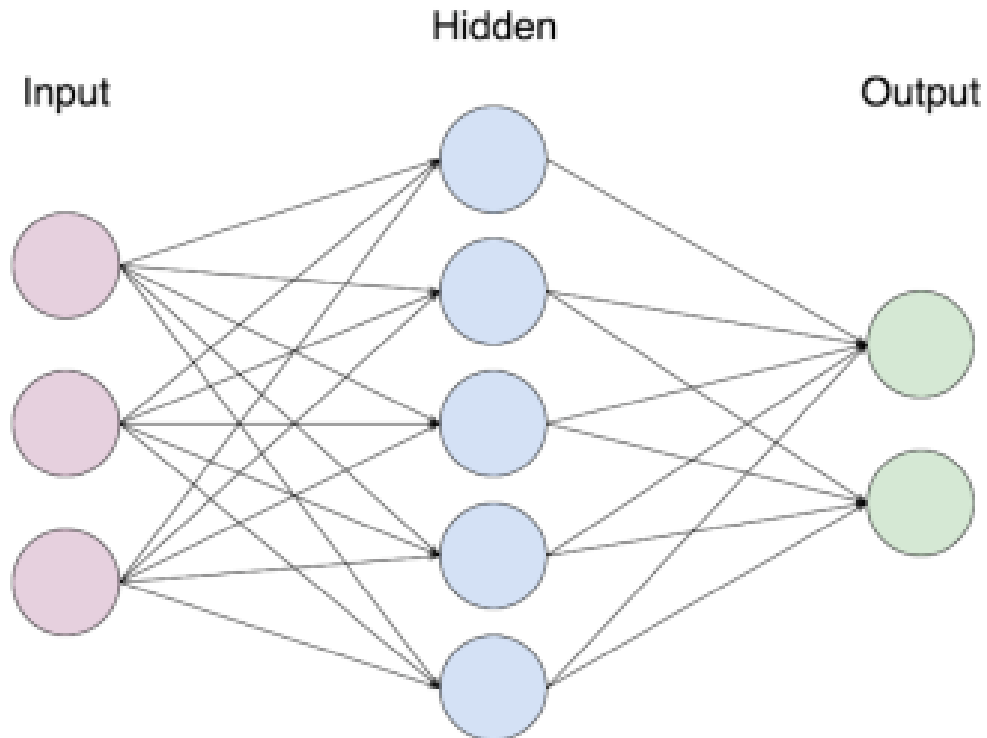


Figure I.16 : Couche entièrement connectée (FC).

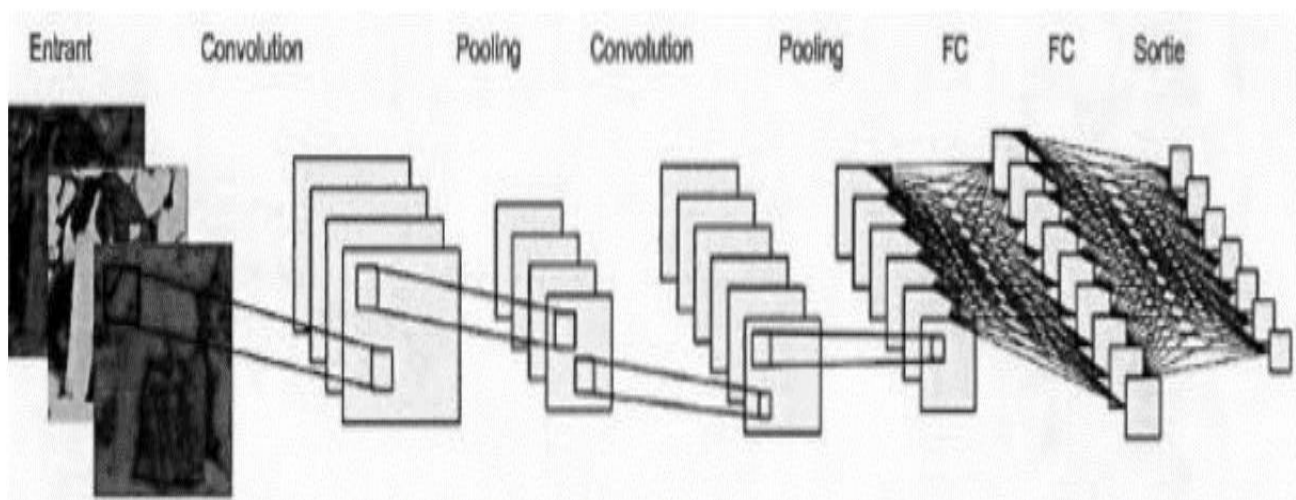


Figure I.17 : Exemple de réseau convolutif deux étapes CONV-POOL suivies par deux couches FC.

#### I.4.4 Avantage des CNN

Les réseaux de neurones convolutifs (CNN) sont largement utilisés dans le traitement d'images. Ils offrent plusieurs avantages dans les tâches de traitement d'images.

1 - Les CNN sont capables d'apprendre automatiquement et d'extraire des caractéristiques hiérarchiques à partir d'images. Ils peuvent facilement identifier les motifs, les contours, les formes, les textures, etc. dans l'image et comprendre les informations visuelles complexes grâce aux couches de convolution [44].

2- Les CNN possèdent une invariance de translation, c'est-à-dire la capacité à apprendre les propriétés invariantes par translation de l'image. Cela leur permet de reconnaître les motifs indépendamment de leur emplacement dans l'image [44].

3- L'un des principaux atouts des réseaux de neurones convolutifs est leur adaptabilité et leur transfert d'apprentissage. Les modèles CNN disponibles préalablement entraînés sur de grands ensembles de données comme ImageNet peuvent être réutilisés pour des tâches d'imagerie spécifiques. L'utilisation de ces modèles préentraînés améliore considérablement les performances des réseaux et réduit le temps nécessaire au développement des modèles [47].

4- Le partage des paramètres est une technique fondamentale dans les CNN qui utilise les mêmes poids et biais dans l'ensemble de l'image. Le partage des paramètres permet aux CNN d'appliquer les mêmes caractéristiques apprises sur différentes régions de l'image. La réduction du nombre de paramètres permet d'économiser de la mémoire et des ressources de calcul, ce qui rend les CNN applicables aux problèmes du monde réel. Grâce au partage des paramètres, le réseau apprend à reconnaître les caractéristiques essentielles indépendamment de leur emplacement dans l'image [30].

5- Les CNN ont révolutionné le traitement d'images, notamment dans les tâches d'imagerie médicale [48].

#### I.5. Conclusion

Ce chapitre de notre étude a abordé les réseaux de neurones artificiels et le deep learning, en mettant l'accent sur les réseaux neuronaux convolutifs (CNN). Nous avons commencé par introduire les réseaux de neurones artificiels, leur histoire et leur structure. Nous avons également exploré l'apprentissage des réseaux de neurones, l'algorithme d'apprentissage et les domaines d'application.

Ensuite, nous avons plongé dans le deep learning, en définissant cette approche et en présentant des exemples d'applications.

Nous avons également examiné différents types d'apprentissage profond tels que l'apprentissage non supervisé, l'apprentissage semi-supervisé et l'apprentissage par renforcement.

Dans la deuxième moitié du chapitre, nous nous sommes concentrés sur les réseaux neuronaux convolutifs. Nous avons expliqué leur architecture, en détaillant les différentes couches telles que la couche de convolution, la couche de pooling, la couche de correction, la couche entièrement connectée et la couche de perte. Nous avons également présenté des exemples de modèles de CNN et mis en évidence leurs avantages.

En général, ce chapitre a fourni une introduction aux réseaux de neurones artificiels et au deep learning, en mettant en évidence l'importance des réseaux neuronaux convolutifs. Les CNN sont particulièrement puissants dans le domaine de la vision par ordinateur et offrent des capacités avancées pour l'analyse et la classification d'images. Ils sont largement utilisés dans de nombreux domaines et ouvrent la voie à de nouvelles opportunités et avancées dans le domaine de l'intelligence artificielle.

---

**Références du chapitre I**

- [1] D. E. Rumelhart, G. E. Hinton, and J. L. McClelland, "A General framework for Parallel Distributed Processing," *Parallel distributed processing: explorations in the microstructure of cognition*. (Vol. 1, pp. 45–76). Cambridge, 1986.
- [2] C. Grosan and A. Abraham, "Artificial Neural Networks," in *Intelligent Systems Reference Library*, vol. 17, 2011, pp. 281–323.
- [3] S. P. Reddy, M. G. Prasad, A. N. RadhaKrishna, K. Saujanya, N. V. K. Raviteja, and B. Deepthi, "Correlation between salivary cortisol levels and dental anxiety in children of smokers and nonsmokers," *Eur. J. Dent.*, vol. 11, no. 02, pp. 192–195, Apr. 2017.
- [4] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, 1943.
- [5] D. O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*. New York: Wiley, 1949.
- [6] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958.
- [7] B. Widrow and M. E. Hoff, "Adaptive switching circuits," in *IRE WESCON Convention Record*, 1960, vol. 4, pp. 96–104.
- [8] B. Widrow, "Generalization and information storage in networks of adaline neurons," in *Self-Organizing Systems*, 1962, pp. 435–461.
- [9] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
- [10] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the National Academy of Sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.
- [11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- [12] L. O. Chua and L. Yang, "Cellular neural networks: Theory," *IEEE Transactions on Circuits and Systems*, vol. 35, no. 10, pp. 1257–1272, 1988.
- [13] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [14] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge, U.K.: Cambridge University Press, 2002.

- 
- [15] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," arXiv preprint arXiv:1207.0580, 2012.
- [16] S. R. Shomali, S. N. Rasuli, M. N. Ahmadabadi, and H. Shimazaki, "Uncovering hidden network architecture from spiking activities using an exact statistical input-output relation of neurons.," *Commun. Biol.*, vol. 6, no. 1, p. 169, Feb. 2023.
- [17] A. Ovchinnikov, G. Pogudin, and P. Thompson, "Input-Output Equations and Identifiability of Linear ODE Models," *IEEE Trans. Automat. Contr.*, vol. 68, no. 2, pp. 812–824, Feb. 2023.
- [18] M. L. Kovalchuk, P. I. Ushtan, Y. O. Ushenko, and I. V. Soltys, "Advantages and Disadvantages of Learning a Multilayer Neural Network Using a Genetic Algorithm," *Optoelectron. Information-Power Technol.*, vol. 43, no. 1, pp. 19–23, Dec. 2022.
- [19] M. A. Saleem, N. Senan, F. Wahid, M. Aamir, A. Samad, and M. Khan, "Comparative Analysis of Recent Architecture of Convolutional Neural Network," *Math. Probl. Eng.*, vol. 2022, pp. 1–9, Mar. 2022.
- [20] F. C. Farias, T. B. Ludermir, and C. J. A. Bastos-Filho, "Embarrassingly Parallel Independent Training of Multi-Layer Perceptrons with Heterogeneous Architectures," *AI*, vol. 4, no. 1, pp. 16–27, Dec. 2022.
- [21] J. Naskath, G. Sivakamasundari, and A. A. S. Begum, "A Study on Different Deep Learning Algorithms Used in Deep Neural Nets: MLP SOM and DBN," *Wirel. Pers. Commun.*, vol. 128, no. 4, pp. 2913–2936, Feb. 2023.
- [22] J. Liu and F.-P. An, "Image Classification Algorithm Based on Deep Learning-Kernel Function," *Sci. Program.*, vol. 2020, no. 1, pp. 1–14, Jan. 2020.
- [23] A. Alhudhaif *et al.*, "A Particle Swarm Optimization Based Deep Learning Model for Vehicle Classification," *Comput. Syst. Sci. Eng.*, vol. 40, no. 1, pp. 223–235, 2022.
- [24] V. Moskalenko and A. Moskalenko, "Neural network based image classifier resilient to destructive perturbation influences – architecture and training method," *Radioelectron. Comput. Syst.*, vol. 2022, no. 3, pp. 95–109, Oct. 2022.
- [25] P. Minist, E. Sup, and R. Scientifique, "la démarche RNA - Analyse statistique :," 2021.
- [26] M. Madhjarasan and M. Louzazni, "Analysis of Artificial Neural Network: Architecture, Types, and Forecasting Applications," *J. Electr. Comput. Eng.*, vol. 2022, no. i, pp. 1–23, Apr. 2022.

- 
- [27] M. M. Taye, “Understanding of Machine Learning with Deep Learning :,” *Comput. MDPI*, vol. 12, no. 91, pp. 1–26, 2023.
- [28] M. C. c. G. - and P. S. B. -, “Implement of Deep Learning-based Model for the Prediction of Successive Words in a Sentence: A Review,” *Int. J. Multidiscip. Res.*, vol. 5, no. 3, pp. 1–9, May 2023.
- [29] E. O. Arkhangelskaya and S. I. Nikolenko, “Deep Learning for Natural Language Processing: A Survey,” *J. Math. Sci.*, vol. 273, no. 4, pp. 533–582, Jul. 2023.
- [30] U. A. Bhatti, H. Tang, G. Wu, S. Marjan, and A. Hussain, “Deep Learning with Graph Convolutional Networks: An Overview and Latest Applications in Computational Intelligence,” *Int. J. Intell. Syst.*, vol. 2023, pp. 1–28, Feb. 2023.
- [31] S. Jin, G. Liu, and Q. Bai, “Deep Learning in COVID-19 Diagnosis, Prognosis and Treatment Selection,” *Mathematics*, vol. 11, no. 6, p. 1279, Mar. 2023.
- [32] K. Chelladurai and N. Sujatha, “A Survey on Different Algorithms Used in Deep Learning Process,” *E3S Web Conf.*, vol. 387, p. 05008, May 2023.
- [33] K. Khan, “A Taxonomy for Deep Learning in Dynamic Adaptive Video Streaming Over HTTP,” *Int. J. Multidiscip. Res. Anal.*, vol. 06, no. 05, pp. 2324–2332, May 2023.
- [34] J. Praful Bharadiya, “A Comprehensive Survey of Deep Learning Techniques Natural Language Processing,” *Eur. J. Technol.*, vol. 7, no. 1, pp. 58–66, May 2023.
- [35] Q. Shen *et al.*, “Unsupervised Learning of Accurate Siamese Tracking,” in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, vol. 2022-June, pp. 8091–8100.
- [36] V. Rani, S. T. Nabi, M. Kumar, A. Mittal, and K. Kumar, “Self-supervised Learning: A Succinct Review,” *Arch. Comput. Methods Eng.*, vol. 30, no. 4, pp. 2761–2775, May 2023.
- [37] S. Paramés-Estévez, A. Carballosa, D. Garcia-Selfa, and A. Munuzuri, “Artificial Intelligence Techniques Used to Extract Relevant Information from Complex Social Networks,” *Entropy*, vol. 25, no. 3, p. 507, Mar. 2023.
- [38] K. Athanasopoulou, G. N. Daneva, P. G. Adamopoulos, and A. Scorilas, “Artificial Intelligence: The Milestone in Modern Biomedical Research,” *BioMedInformatics*, vol. 2, no. 4, pp. 727–744, Dec. 2022.
- [39] X. Yang, J. Wu, and X. Chen, “Application of Artificial Intelligence to the Diagnosis and Therapy of Nasopharyngeal Carcinoma,” *J. Clin. Med.*, vol. 12, no. 9, p. 3077, Apr. 2023.
- [40] A. Celeghin *et al.*, “Convolutional neural networks for vision neuroscience: significance,

- 
- developments, and outstanding issues,” *Front. Comput. Neurosci.*, vol. 17, no. July, pp. 1–13, Jul. 2023.
- [41] S. Goumiri, D. Benboudjema, and W. Pieczynski, “A new hybrid model of convolutional neural networks and hidden Markov chains for image classification,” *Neural Comput. Appl.*, vol. 35, no. 24, pp. 17987–18002, Aug. 2023.
- [42] G. A. Ali and A. hussein ali, “Convolutional Neural Networks using FPGA-based Pipelining,” *Iraqi J. Comput. Sci. Math.*, vol. 4, no. 2, pp. 215–223, May 2023.
- [43] I. Vezakis, A. Vezakis, S. Gourtsoyianni, V. Koutoulidis, G. K. Matsopoulos, and D. Koutsouris, “CEC-CNN: A Consecutive Expansion-Contraction Convolutional Network for Very Small Resolution Medical Image Classification,” Sep. 2022.
- [44] M. M. Taye, “Theoretical Understanding of Convolutional Neural Network :,” *Mdpi Ag*, vol. 11, no. 3, p. 52, 2023.
- [45] N.-I. Galanis, P. Vafiadis, K.-G. Mirzaev, and G. A. Papakostas, “Convolutional Neural Networks: A Roundup and Benchmark of Their Pooling Layer Variants,” *Algorithms*, vol. 15, no. 11, p. 391, Oct. 2022.
- [46] I. A. Atoum, “Adaptive Rectified Linear Unit (Arelu) for Classification Problems to Solve Dying Problem in Deep Learning,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 2, pp. 97–102, 2023.
- [47] A. W. Salehi, S. Khan, G. Gupta, B. I. Alabdullah, and A. Almjally, “Cnn1.Pdf,” 2023.
- [48] S. Xie, J. Song, Y. Hu, C. Zhang, and S. Zhang, “Using CNN with Multi-Level Information Fusion for Image Denoising,” *Electronics*, vol. 12, no. 9, p. 2146, May 2023.

---

## *Chapitre II*

### *Modèles de Markov cachés HMM en classification d'images*

---

## II. 1. Introduction

Les modèles de Markov cachés sont largement utilisés dans divers domaines, en particulier dans la reconnaissance vocale, la biologie, le traitement d'images et de signaux et la communication numérique. Ils sont bien connus pour leur efficacité dans la modélisation des corrélations entre les symboles adjacents, les domaines ou les événements, mais ils souffrent souvent de problèmes de grande dimension. Un modèle de Markov caché est un modèle statistique composé d'états cachés, d'observations et de transitions. Chaque état émet des observations.

La classification d'images par les modèles de Markov cachés est basée sur l'approche de classification par région qui sert à classifier l'image selon la similarité de ses régions composantes.

La classification d'image est l'un des problèmes les plus importants et les plus difficiles dans le domaine du traitement d'images lorsque des pixels présentant des caractéristiques similaires sont regroupés en régions homogènes. De nombreuses tâches de traitement de haut niveau telles que l'extraction de fonctionnalités, la reconnaissance d'objets et le diagnostic médical dépendent fortement de la qualité des solutions de classification. Pour obtenir une classification appropriée, plusieurs critères doivent être satisfaits, tels que la compacité, la séparation et le chevauchement. Parfois, des connaissances de haut niveau sur la forme et l'apparence des objets sont nécessaires. Dans de nombreuses applications, cependant, ces informations ne sont pas disponibles ou peu pratiques à utiliser.

Ce chapitre débute par une présentation des différents types de modèles de Markov cachés utilisés pour la classification d'images. Il aborde d'abord les modèles traditionnels, tels que les champs, les chaînes et les arbres, avant de passer aux modèles plus récents, comme les modèles de Markov couplés, triplets et flous. Enfin, il examine les extensions de l'approche HMM, ainsi que ses avantages et ses limites.

## II.2. Modèles de Markov cachés

Les HMM sont une autre classe de modèles probabilistes avec un cadre séquentiel, ce qui les rend particulièrement adaptés aux tâches impliquant des données ordonnées.

Dans le contexte de la classification, les HMMs modélisent la dynamique sous-jacente des intensités d'image en considérant une séquence d'états cachés, chacun correspondant à un tissu ou à une structure différente. Les HMM ont trouvé des applications dans les modalités d'imagerie dynamique.

En considérant les dépendances temporelles des intensités d'image, les HMM peuvent discerner les transitions entre différents états, facilitant ainsi la classification des structures dynamiques au fil du temps. Ceci est particulièrement utile dans l'imagerie fonctionnelle où l'évolution des propriétés des tissus est un facteur clé [1].

La théorie des chaînes de Markov, née en 1913, a trouvé sa première application dans l'analyse du langage par Markov lui-même. Cependant, ce n'est que dans les années 1960 que le plein potentiel des chaînes de Markov a été réalisé grâce à diverses applications telles que les modèles probabilistes d'urnes de Neuwirth, les calculs du maximum de vraisemblance et l'observation de séquences d'états au sein de la chaîne. Ces développements ont permis aux chercheurs d'exploiter la puissance des modèles de Markov dans divers domaines tels que le traitement du langage naturel, l'apprentissage automatique et la modélisation probabiliste. La polyvalence et l'accessibilité analytique des chaînes de Markov les ont rendues indispensables aux sciences quantitatives, avec des applications allant de la biologie computationnelle à la finance computationnelle.

Les années 70 ont vu l'émergence d'algorithmes puissants pour la reconnaissance, l'analyse et l'apprentissage.

Les HMMs étendent le concept des chaînes de Markov en incorporant des états cachés. Dans le contexte de la classification d'images :

- **États cachés** : Représentent les vraies étiquettes de classification (par exemple, objet, arrière-plan) qui ne sont pas directement observables.
- **Observations** : Correspondent aux valeurs des pixels observés ou aux caractéristiques extraites de l'image

Les HMM utilisés en classification d'images peuvent être divisés en deux catégories : Les modèles classiques et les modèles récents.

- **Modèles classiques** : Ces modèles supposent que le processus caché  $X$  est markovien, homogène et stationnaire. Les observations  $Y$  sont indépendantes du processus caché  $X$ . Ces modèles sont appelés modèles de Markov cachés à bruit indépendant.
- **Modèles récents** : Ces modèles sont développés pour surmonter les limitations des modèles classiques.

Les modèles couples sont utilisés pour modéliser le bruit ou la texture en considérant un couple markovien et stationnaire  $X$  et  $Y$ . Les modèles triplets introduisent un processus auxiliaire  $U$  pour modéliser la non-stationnarité du processus caché  $X$  ou le temps de séjour du processus markovien  $X$ .

Le choix d'un modèle HMM pour la classification d'images dépend des caractéristiques des données, notamment de la stationnarité, de la non-stationnarité, du bruit et de la texture.

La figure (II.1) présente les différents modèles de Markov utilisés en classification d'images.

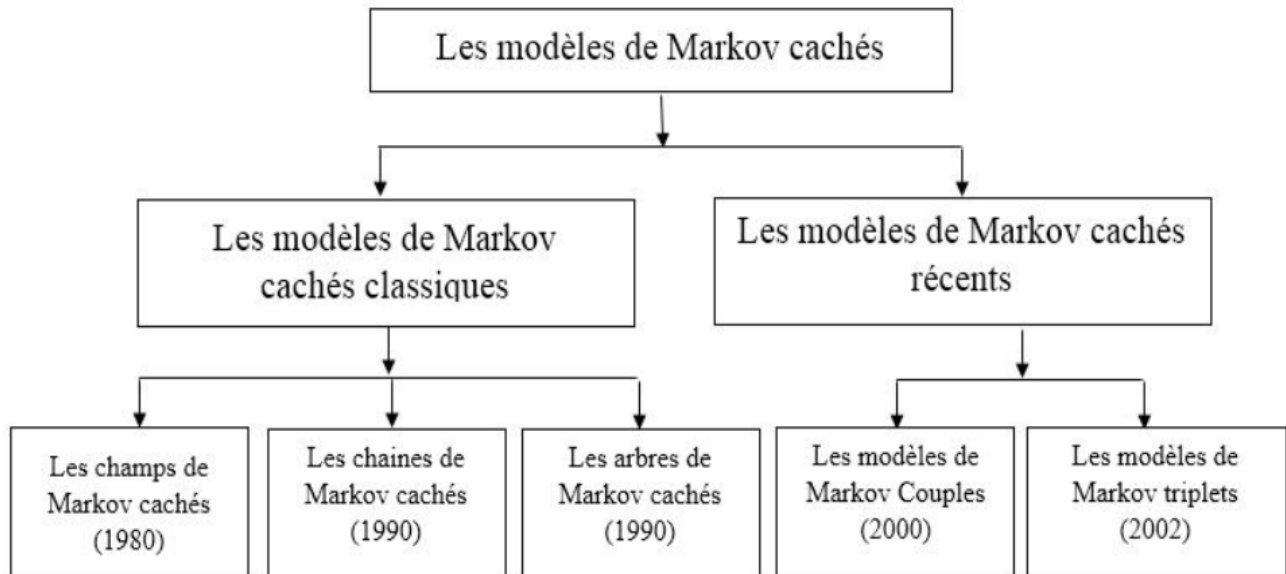


Figure II.1 : Modèles de Markov cachés en classification d'images

### II.2.1 Méthodes locales contextuelles

Les méthodes contextuelles locales des modèles de Markov cachés (HMM) jouent un rôle clé et important dans la classification d'images, en améliorant à la fois la précision et l'efficacité des résultats, notamment pour les images complexes. En tirant parti des relations entre les patches locaux et leurs contextes plus larges, ces méthodes affinent les résultats de classification en intégrant des informations locales et contextuelles. Grâce à cette exploitation simultanée des deux niveaux d'information, les HMM permettent d'obtenir des classifications plus précises et robustes[2].

Dans le cas où l'ensemble  $X$  des états cachés est de cardinal important, notons  $S$  l'ensemble des pixels et considérons les "champs aléatoires"  $X = (X_s)_{s \in S}$  et  $Y = (Y_s)_{s \in S}$ , où  $X_s$  et  $Y_s$  représentent respectivement les variables d'état caché et les observations pour chaque pixel  $s$ .

La décision bayésienne optimale pour attribuer une classe  $x_s$  à un pixel  $s$  est donnée par :

$$dBL(y_s) = \operatorname{argmin}_{x_s} \sum_{x_s} L(x_s, x) P(X_s = x_s | Y_s = y_s) \quad (\text{II.1})$$

Où :

$L(x_s, x)$  : est une fonction de perte.

$P(X_s = x_s | Y_s = y_s)$  : est la probabilité a posteriori.

Cependant, cette approche "pixel par pixel" peut être améliorée en prenant en compte le voisinage  $V_s$  du pixel  $s$ . Soit  $Y_{V_s} = (Y_t)_{t \in V_s}$  les observations dans le voisinage de  $s$ .

On peut alors montrer que :

$$E[L(\text{dBL}(Y_{V_s}), X_s)] \leq E[L(\text{dBL}(Y_s), X_s)] \quad (\text{II.2})$$

Autrement dit, l'utilisation du voisinage  $Y_{V_s}$  permet d'améliorer l'efficacité des stratégies Bayésiennes par rapport à l'utilisation des seules observations  $Y_s$ [3].

Bien que les méthodes locales contextuelles basées sur les modèles de Markov cachés donnent de bons résultats dans certaines situations, elles se heurtent rapidement à des problèmes de calcul lorsque la taille du voisinage  $V$  augmente. Pour deux classes, il est ainsi difficile de prendre en compte des voisinages de taille supérieure à huit. Avec un nombre courant de l'ordre de cinquante mille pixels, une partie non négligeable de l'information disponible ne peut être exploitée par ces méthodes locales.

### II.2.2 Champs de Markov cachés

Les champs aléatoires de Markov (Markov Random Fields, MRF) jouent un rôle fondamental dans le traitement d'images, car ils permettent de modéliser efficacement les dépendances spatiales de variables aléatoires. Ils ont été largement utilisés dans diverses tâches de traitement d'image telles que la classification, la détection d'objets et la reconstruction. Les modèles de Markov cachés (HMM), un type de champ de Markov, sont particulièrement répandus dans la classification d'images en raison de leur capacité à capturer des relations complexes entre les éléments de l'image.

Les champs aléatoires de Markov, y compris les HMM, fournissent un cadre robuste pour comprendre et traiter les images, avec des applications allant des techniques statistiques à la régularisation basée sur la physique. La flexibilité et les capacités de modélisation stochastique des champs aléatoires de Markov en font des outils indispensables pour l'analyse et le traitement d'images, démontrant ainsi leur importance dans le domaine de la vision par ordinateur et au-delà [4][5].

Les modèles de Markov cachés (HMM) sont de puissants outils largement utilisés dans divers domaines tels que la biophysique, le traitement du signal et l'appariement de cartes. Les HMM sont particulièrement utiles pour modéliser des données de séries chronologiques et les regrouper, en facilitant des tâches telles que l'analyse de trajectoires et l'appariement de cartes. Bien que les variables cachées dans les HMM ne soient pas interprétées physiquement directement, elles offrent une certaine flexibilité dans la modélisation de phénomènes complexes, ce qui permet une interprétation a posteriori après l'analyse [6].

Les modèles de champs de Markov cachés, en raison de leur structure inhérente, permettent de prendre en compte les relations spatiales entre les pixels d'une image et d'intégrer des connaissances antérieures sur la classification[7].

Les champs de Markov cachés posent des difficultés lors de la sélection des paramètres qui régissent les interactions spatiales et nécessitent souvent des algorithmes gourmands en calculs. Ces modèles sont fréquemment intégrés dans des cadres de classification bayésiens, où la classification est réalisée en maximisant la probabilité a posteriori sur la base des données d'image à l'aide de techniques itératives telles que l'ICM (pour « Mode Conditionnel Itéré »), le MAP (pour « Maximum A Posteriori ») et le MPM (pour « Mode des Marginales a Posteriori »).

Par exemple, dans le domaine de l'imagerie médicale, ces méthodes sont couramment utilisées pour considérer la prévalence de pixels appartenant à la même catégorie que leurs pixels voisins. Ces techniques sont largement utilisées non seulement dans le but de délimiter des catégories de classification, mais également pour représenter les variations d'intensité qui pourraient se manifester, par exemple sur les images de mammographies [8][9].

Les modèles de Markov cachés sont utilisés pour représenter des zones homogènes, ce qui les rend adaptés à l'analyse de la classification d'images par le biais d'un apprentissage non supervisé. [10].

Soit  $X = \{X_s, s \in S\}$  un champ de Markov caché, où  $S$  représente l'ensemble des pixels de l'image et  $X_s$  la variable d'état cachée (classe) au pixel  $s$ . Soit  $Y = \{Y_s, s \in S\}$  le champ d'observations associé, où  $Y_s$  représente l'observation (niveau de gris) au pixels. Le champ  $X$  est supposé être un champ de Markov discret, c'est-à-dire que la distribution de probabilité de  $X_s$  conditionnellement aux autres variables d'état  $X_t, t \neq s$ , ne dépend que des variables d'état dans un voisinage  $V_s$  de  $s$  :

$$P(X_s | X_t, t \neq s) = P(X_s | X_t, t \in V_s) \quad (\text{II.3})$$

Le champ d'observations  $Y$  est supposé être conditionnellement indépendant étant donné le champ  $X$  :

$$P(Y | X) = \prod P(Y_s | X_s) \quad (\text{II.4})$$

La loi jointe du modèle s'écrit alors :

$$P(X, Y) = P(X) \prod P(Y_s | X_s) \quad (\text{II.5})$$

où  $P(X)$  est la loi du champ de Markov caché  $X$  [3].

- **Estimation des paramètres par algorithme EM**

L'objectif est d'estimer les paramètres du modèle à partir des observations  $Y$ . Cela peut se faire par maximum de vraisemblance en utilisant l'algorithme Espérance-Maximisation (EM) :

1. Initialisation des paramètres  $\theta = (\theta_X, \theta_Y)$
2. Étape E : Calcul de l'espérance de la log-vraisemblance complète

$$Q(\theta | \theta(k)) = E[\log P(X, Y | \theta) | Y, \theta(k)] \quad (\text{II.6})$$

3. Étape M : Maximisation de  $Q(\theta | \theta(k))$  pour obtenir  $\theta(k + 1)$
4. Répéter les étapes 2 et 3 jusqu'à convergence

L'algorithme EM permet d'obtenir un estimateur du maximum de vraisemblance des paramètres  $\theta$ .

- **Inférence sur les états cachés**

Une fois les paramètres estimés, on peut s'intéresser à l'inférence sur les états cachés  $X$  (classification de l'image) à partir des observations  $Y$ . Deux approches principales sont utilisées :

1. Maximum a posteriori (MAP) : Trouver la configuration  $\hat{x}$  qui maximise  $P(X = x | Y = y)$
2. Maximum a posteriori des lois marginales (MPM) : Trouver la configuration  $\hat{x}$  où chaque variable  $X_s$  est dans l'état qui maximise  $P(X_s = x_s | Y = y)$

Ces problèmes d'inférence peuvent être résolus de manière efficace à l'aide d'algorithmes de programmation dynamique tels que l'algorithme forward-backward ou l'algorithme de Viterbi.

Généralisation aux champs de markov cachés flous

Les champs de markov cachés durs peuvent être généralisés au cas flou, où les variables d'état  $X_s$  prennent leurs valeurs dans l'intervalle représentant les degrés d'appartenance aux différentes classes. Cela permet de modéliser des classifications floues, plus adaptées à certaines applications.

La loi jointe du modèle flou s'écrit alors :

$$P(X, Y) = P(X) \prod P(Y_s | X_s) \quad (\text{II.7})$$

Où  $P(X)$  est la loi du champ de Markov caché flou  $X$ .

L'estimation des paramètres et l'inférence sur les états cachés flous peuvent être réalisées à l'aide d'adaptations des algorithmes EM et MPM[3].

### II.2.3 Classification bayésienne avec les champs de markov cachés

La classification bayésienne avec des champs de Markov cachés est une technique puissante largement utilisée dans divers domaines tels que le traitement d'images [11][12]. La classification bayésienne avec les champs de Markov cachés combine les principes bayésiens avec les dépendances spatiales modélisées par les champs de Markov pour obtenir des résultats de classification d'image précis et robustes. Cette approche utilise le cadre probabiliste de l'inférence bayésienne pour estimer la classification la plus probable compte tenu des données d'image observées et des états cachés sous-jacents représentés par les champs de Markov.

La formulation d'équations joue un rôle crucial dans ces modèles bayésiens, aidant à la classification précise d'images et des structures souterraines en gérant le bruit non gaussien et l'hétérogénéité du sous-sol avec des incertitudes quantifiées [13]. Des techniques comme les champs aléatoires cachés de markov et les modèles de mélange gaussiens sont utilisées pour représenter les liens cachés et les paramètres statistiques des ensembles de données, respectivement [14]. En outre, les méthodes d'estimation bayésiennes pour les champs aléatoires cachés de Potts-Markov contribuent à une classification rapide et non supervisée d'images en adaptant automatiquement la régularisation spatiale pendant le processus d'inférence. Ces méthodologies offrent une convergence efficace, des paramètres de régularisation auto-ajustés et des résultats de classification précis, montrant l'importance de la formulation d'équations pour obtenir une classification bayésienne réussie avec des champs de Markov cachés.

La formulation de la classification bayésienne avec des champs de markov cachés peut être exprimée en utilisant l'estimation maximum à posteriori (MAP), où l'objectif est de trouver la classification qui maximise la probabilité postérieure compte tenu des données d'image observées. Cela peut être représenté mathématiquement comme suit[3] :

$$[ S * = \arg \max_S P(S | I) \text{ propto } \arg \max_S P(I | S)P(S) ] \quad (\text{II.8})$$

### II.2.4 Chaînes de Markov cachées

Les chaînes de Markov cachées (HMC) sont des modèles statistiques puissants qui modélisent la relation entre les états cachés des pixels d'une image et les données observées, telles que les intensités de pixels.

En capturant les dépendances spatiales entre ces pixels, les HMC permettent une classification non supervisée plus précise d'images. Elles estiment les probabilités a posteriori des états cachés, ce qui facilite la classification des pixels tout en réduisant le bruit. Grâce à ces relations contextuelles, les HMC jouent un rôle clé dans l'amélioration de la classification d'images, notamment dans les applications complexes comme l'imagerie médicale[15].

Nous considérons que nous avons accès à une version bruitée du signal modélisé par une chaîne de Markov, le problème général étant d'estimer la réalisation non observable de cette chaîne. Nous considérons les processus stochastiques  $X = X_1 : N$  et  $Y = Y_1 : N$ .

$Y$  est observable tandis que  $X$  ne l'est pas. Chaque  $X_n$  prend ses valeurs dans l'ensemble fini de  $K$  classes  $X = \{w_1; \dots, w_K\}$  et chaque  $Y_n$  prend ses valeurs dans l'ensemble des nombres réels  $R$ . Le graphe d'orientation des dépendances pour le modèle HMC avec  $N = 3$  est présenté dans la figure II. 2.

Dans le modèle HMC classique  $(X, Y)$ , nous considérons  $X$  comme une chaîne de Markov avec la distribution

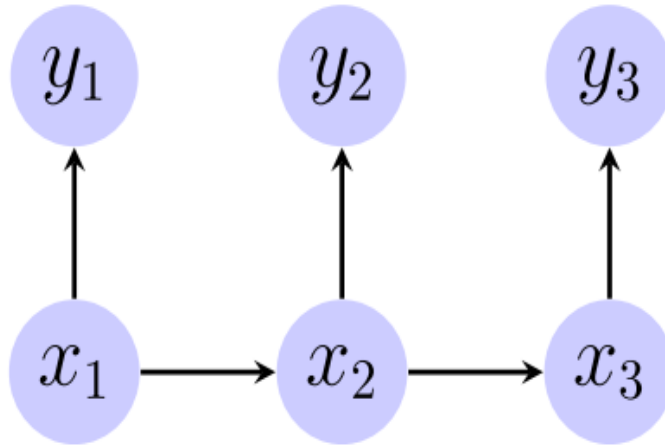


Figure II. 2 : Graphe d'orientation des dépendances pour la chaîne de Markov cachée

$$p(x) = p(x_1) \prod_{n=1}^{N-1} p(x_{n+1} | x_n) \quad (\text{II.9})$$

Et la distribution  $p(y|x)$  est définie par :

$$p(y | x) = \prod_{n=1}^N p(y_n | x_n) \quad (\text{II.10})$$

Où  $p(y_n | x_n)$  sont supposées gaussiennes. Le mode marginal postérieur bayésien (MPM) que nous utilisons pour l'estimation de  $\hat{x} = (\hat{x}_1; \dots; \hat{x}_N)$  de  $X$  à partir de  $Y$  est défini par :

Pour chaque  $n = 1; \dots; N$ ;

$$\hat{x}_n = \underset{w_k}{\operatorname{argmax}} p(x_n = w_k | y) = \underset{w_k}{\operatorname{argmax}} \alpha_n(k) \beta_n(k) \quad (\text{II.11})$$

• **Chaîne observable** : La progression du processus de Markov est illustrée par un graphe illustrant les transitions d'état (figure II.3), décrivant la structure du processus sur la base de directives spécifiques[15].

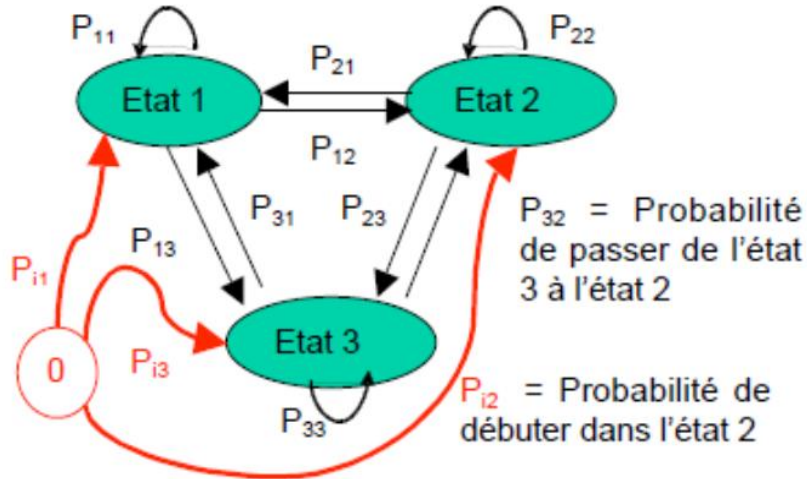


Figure II.3 : Représentation graphique d'un modèle de markov observable

### II.2.5 Arbres de Markov cachés

Les arbres de Markov cachés (HMTs) sont des modèles probabilistes qui ont été appliqués avec succès dans divers domaines tels que la classification d'images et l'analyse génétique [16]. Ces modèles établissent un équilibre entre la complexité des variables modélisées et la facilité d'exécution des procédures d'inférence [17]. Cependant, les hypothèses d'indépendance conditionnelle dans les EGH traditionnelles peuvent être trop simplistes pour certaines applications comme la classification naturelle d'images [18].

Pour remédier à cette limitation, des arbres de Markov de triplets spatiaux (STMTs) ont été introduits comme une extension des HMTs, visant à améliorer les corrélations entre les variables aléatoires tout en maintenant des capacités d'inférence efficaces [19].

Les STMTs ont montré des performances supérieures aux HMTs dans les tâches de classification d'images non supervisées, mettant en évidence de riches dépendances spatiales cruciales pour une classification précise [20].

Cette avancée dans les techniques de modélisation met en évidence l'évolution continue et le raffinement des modèles de Markov cachés pour mieux capturer les relations complexes dans les ensembles de données.

### II.2.6 Choix de modèle et de méthode

Lors de la sélection d'un modèle de Markov caché (HMM) pour une application spécifique, les chercheurs sont confrontés à des défis en raison des observations incomplètes et du bruit dans les mesures. Pour résoudre ces problèmes, une nouvelle approche combine HMM avec les préférences de choix de chemin des décideurs, améliorant la précision de correspondance, en particulier pour les données de trajectoire à haute fréquence [21].

De plus, une méthodologie fusionne l'interaction de plusieurs structures de modèle avec HMM pour améliorer la précision du suivi des cibles en estimant des vecteurs d'état plus précis basés sur les modèles les plus probables sélectionnés [22].

En outre, une méthode de formation HMM implique d'affiner itérativement le modèle en ajoutant des échantillons mal classés à l'ensemble de formation et en comparant les précisions de prédiction, ce qui conduit à une amélioration des performances[3].

La sélection de modèles de Markov cachés (HMMs) et de méthodes pour la classification d'images nécessite une compréhension approfondie de la complexité du modèle, de l'efficacité de calcul et des caractéristiques des données. Les MMT jouent un rôle essentiel dans les scénarios avec des états cachés, permettant une classification robuste d'images en modélisant les dépendances spatiales et en capturant des informations contextuelles[23].

Les principaux critères de sélection des modèles HMM comprennent l'ajustement du modèle d'équilibre à l'aide du critère d'information bayésien (BIC) et du critère d'information Akaike (AIC) pour éviter le surajustement tout en maintenant l'expressivité du modèle, et assurer l'évolutivité pour les grands ensembles de données généralement trouvés dans l'analyse d'images.

Des méthodes telles que l'algorithme Expectation-Maximisation (EM) sont utilisées pour l'estimation des paramètres, tandis que des extensions telles que Hidden Markov Fields (HMF) sont utilisées pour traiter les dépendances de données spatiales. Les techniques de validation croisée garantissent en outre la performance du modèle sur des données invisibles, améliorant la généralisabilité et la robustesse des applications dans des domaines tels que l'imagerie médicale et la télédétection [24].

### II.3. Modèles de Markov couple

Les modèles de markov couples (MMCouples), introduits par W. Pieczynski[25] au début des années 2000, représentent une extension des modèles de markov classiques. Ils apportent des solutions aux problèmes de classification complexes que les modèles traditionnels ne peuvent pas résoudre, comme la classification d'images en tenant compte du bruit et des contours des objets [26].

Les MMCouples intègrent la corrélation entre les observations  $Y$  [27], ce qui améliore considérablement la précision des résultats de classification.

Les modèles de Markov classiques ont été généralisés pour inclure des modèles couplés, donnant naissance à trois types principaux :

- Les champs de Markov couplés,
- Les chaînes de Markov couplées,
- Les arbres de Markov couplés.

En particulier, les modèles de Markov cachés (HMM) au sein de cette famille améliorent la classification d'images en capturant efficacement les dépendances spatiales entre les valeurs des pixels. Des variantes telles que les modèles de Markov par paires et les arbres de Markov triplets spatiaux (STMT) exploitent des cadres probabilistes, ce qui contribue à accroître la précision des résultats, surtout dans des environnements caractérisés par un bruit important [27]. Grâce à ces avancées, les MMCouples se révèlent être des outils puissants pour optimiser la classification d'images, ouvrant de nouvelles perspectives pour des applications variées dans le traitement d'images.

#### Définition :

Un processus  $Z = (X, Y)$  est un processus de markov couplé si et seulement si :

- Le processus  $X$ , conditionnellement à  $Y$ , est un processus de markov.
- Le processus  $Y$ , conditionnellement à  $X$ , est un processus de markov.
- Les observations  $y_n$  sont corrélées entre elles.

Sa loi est définie par :

$$p(z) = p(z_1) \cdot \prod_{n=1}^{N-1} p(z_{n+1} | z_n) \quad (\text{II.12})$$

On peut écrire les transitions  $p(z_{n+1} | z_n)$  comme suit :

$$p(z_{n+1} | z_n) = p(x_{n+1} | x_n, y_n) p(y_{n+1} | x_n, x_{n+1}, y_n) \quad (\text{II.13})$$

Donc :

$$p(z) = p(x_1, y_1) \cdot \prod_{n=1}^{N-1} p(x_{n+1}|x_n, y_n) p(y_{n+1}|x_n, x_{n+1}, y_n) \quad (\text{II.14})$$

La manière dont on modélise le processus  $X$  et les paramètres des modèles couplés varient selon le type de modèle utilisé : champs, chaînes ou arbres. Bien que la modélisation markovienne du processus  $X$  soit restée la même quel que soit le modèle utilisé, les paramètres varient d'un modèle à l'autre. Les modèles couplés présentent une complexité de calcul plus élevée que celle des modèles classiques, car ils nécessitent le calcul d'un grand nombre de paramètres[28].

#### II.4. Apprentissage des modèles de Markov

Dans la classification d'images, différentes méthodes d'apprentissage des modèles de Markov ont été proposées. Une approche consiste à utiliser des algorithmes d'apprentissage par renforcement profond (DRL) pour entraîner les agents à segmenter les régions d'intérêt dans les images médicales, en améliorant progressivement les résultats grossiers à fins. Une autre méthode utilise des modèles de mélange de Gauss de Markov cachés (HMGMs) pour la classification d'image multiclasse, incorporant l'apprentissage supervisé et les algorithmes stochastiques d'espérance-maximisation pour estimer les états et les paramètres cachés, offrant une meilleure classification et homogénéité spatiale par rapport à d'autres méthodes populaires. En outre, les modèles basés sur le champ aléatoire de Markov (MRF) sont utilisés pour la classification paramétrique d'images, calculant la probabilité des données observées à chaque pixel généré par un modèle d'intensité spécifique, avec des informations préalables encodées sous la forme d'un MRF avec des potentiels quadratiques pour une estimation optimale. Ces approches présentent les diverses applications des modèles de Markov dans la classification d'images, offrant des techniques avancées pour des processus de classification précis et efficaces. Ces méthodes améliorent collectivement la précision et l'efficacité de la classification dans les tâches de traitement d'images.

L'apprentissage des modèles de Markov dans la classification d'images implique plusieurs étapes. En premier lieu, l'image est segmentée à l'aide de la méthode de classification de Markov, suivie d'un traitement par l'algorithme de jeu de niveaux pour mettre à jour la fonction de jeu de niveaux, le champ biaisé et la moyenne des zones grises. Par la suite, l'image originale divisée par le champ biaisé mis à jour est corrigée, puis segmentée à nouveau par l'algorithme de classification de Markov.

Ce processus itératif se poursuit jusqu'à ce qu'une valeur prédéfinie, qui est la somme du nombre d'itérations de l'algorithme de jeu de niveaux et de l'algorithme de classification de Markov, soit atteinte, ce qui donne le résultat final de la classification.

De plus, les méthodes de formation sur le modèle de classification d'image comprennent l'acquisition d'échantillons d'images, la division des super pixels, l'obtention d'étiquettes de pixels rigides et souples et la formation du modèle en fonction de ces étiquettes pour améliorer la précision et l'efficacité de la classification [27].

### **II.5. Extensions**

Les extensions des modèles de Markov cachés (HMM) jouent un rôle crucial dans la classification d'images, en particulier dans les scénarios impliquant d'images multi-capteurs et des données incomplètes. La théorie de Dempster-Shafer est utilisée pour la fusion efficace dans la gestion des données incertaines et contradictoires, améliorant la précision de la prise de décision.

De plus, le concept de HMM diversifié (dHMM) introduit un préalable encourageant la diversité pour l'étiquetage séquentiel dynamique, prouvant son efficacité dans des scénarios non supervisés et supervisés comme la reconnaissance optique de caractères (OCR).

En outre, l'intégration des champs de Markov cachés avec la théorie de la preuve offre de nouvelles possibilités pour la modélisation de l'incertitude et la fusion d'informations, en particulier dans le traitement d'images radar à synthèse d'ouverture (SAR) pour la classification et la détection des changements.

Ces avancées dans les extensions HMM et les techniques de fusion contribuent de manière significative à améliorer la classification d'images dans diverses applications, en relevant des défis tels que les modèles de Markov flous et les modèles graphiques[29].

### **II.6. Modèle probabiliste et classification bayésienne**

Les modèles probabilistes sont essentiels dans la classification d'images, car ils offrent un cadre solide pour appréhender l'incertitude et la variabilité des données d'imagerie complexe. Parmi cette diversité de modèles, les modèles de mélange gaussiens (GMMS) et les modèles de Markov cachés (HMMs) se distinguent par leur polyvalence et leur capacité à modéliser des relations statistiques complexes au sein d'images[30].

En abordant la classification comme un problème de réétiquetage dans un cadre probabiliste, ces modèles permettent d'estimer les distributions de probabilité, offrant ainsi une représentation nuancée des relations entre les caractéristiques de l'image.

La classification bayésienne utilisant des HMM joue un rôle clé en exploitant les informations spatiales et les distributions d'étiquettes de classe, facilitant la partition efficace d'images en régions homogènes et améliorant ainsi la précision de la classification.

Les champs aléatoires de Markov cachés (HMRF) constituent une approche courante, où la distribution des étiquettes est influencée par les pixels voisins, ce qui renforce les performances de classification en intégrant des informations spatiales.

Dans un cadre bayésien, chaque pixel est associé à une classe, et la relation entre les données observées et les classes est modélisée par des fonctions de vraisemblance, les distributions antérieures étant construites sous forme de champs aléatoires de Markov pour tenir compte des corrélations spatiales.

Cependant, bien que ces modèles améliorent considérablement la précision de la classification, ils nécessitent souvent des calculs intensifs, en particulier dans des scénarios non convexes, ce qui peut limiter leur utilisation dans des applications en temps réel [31].

Le théorème de Bayes permet de calculer la probabilité a posteriori d'une classe donnée les observations, en utilisant la formule suivante :

$$P(C_i | \mathbf{x}) = P(\mathbf{x})P(\mathbf{x} | C_i)P(C_i) \quad (\text{II.15})$$

Où :

- $P(C_i | \mathbf{x})$  est la probabilité a posteriori de la classe  $C_i$  donnée l'observation  $\mathbf{x}$ ,
- $P(\mathbf{x} | C_i)$  est la vraisemblance de l'observation  $\mathbf{x}$  étant donné la classe  $C_i$ ,
- $P(C_i)$  est la probabilité a priori de la classe  $C_i$ ,
- $P(\mathbf{x})$  est la probabilité marginale de l'observation  $\mathbf{x}$ .

La classification bayésienne cherche à maximiser la probabilité postérieure, en attribuant chaque pixel à la classe ayant le P le plus élevé ( $C_i | \mathbf{x}$ ).

Ce travail de recherche se concentre sur l'application des modèles de Markov cachés à la tâche de classification d'images.

## II.7. Avantages et limites de l'approche HMM pour la classification d'images

### II.7.1. Avantages du HMM

- **Flexibilité :** Les HMMs peuvent modéliser des dépendances complexes entre les pixels d'image, ce qui les rend appropriés pour segmenter d'images avec diverses textures et formes.
- **Fondement statistique :** Les HMMs sont basées sur un cadre statistique solide, permettant une estimation robuste des paramètres du modèle et des intervalles de confiance pour les résultats de classification.
- **Efficacité :** Les HMMs peuvent être entraînées et utilisées efficacement, en particulier pour les grandes images. Algorithmes bien définis : Les algorithmes existants pour la formation et le décodage des HMMs sont bien établis et facilement disponibles, ce qui facilite la mise en œuvre.
- **Bonnes performances pour des types d'images spécifiques :** Les HMMs se sont avérées efficaces pour segmenter d'images avec des structures bien définies, telles que des images médicales ou des images satellites[32].

### II.7.2. Limites de HMM

- **Hypothèses :** Les HMMs font plusieurs hypothèses simplificatrices, telles que la stationnarité et l'indépendance des valeurs de pixels, qui peuvent ne pas être vraies pour tous les types d'images.
- **Complexité informatique :** La formation et le décodage des HMMs peuvent devenir coûteux en calcul pour les modèles complexes ou les images hautes résolution.
- **Sensibilité à l'initialisation :** Les performances des HMMs peuvent être sensibles aux valeurs initiales des paramètres du modèle, nécessitant des stratégies d'initialisation minutieuses.
- **Difficulté à modéliser les dépendances à longue portée :** Les HMMs sont généralement plus efficaces pour capturer les dépendances locales entre les pixels, ce qui limite leur efficacité pour la classification d'images contenant des motifs spatiaux à longue portée.
- **Capacité limitée à gérer le bruit :** Les HMMs peuvent être sensibles au bruit dans les images, ce qui peut entraîner des résultats de classification inexacts[33].

## II.8. Conclusion

Au cours de ce chapitre, nous avons présenté dans un premier temps les différents types de modèles de Markov appliqués en classification d'images tels que : les champs, les chaînes, les arbres, les modèles couplés, les modèles triplets et les modèles flous. Les démarches de classification utilisées par ces modèles se ressemblent, la différence est au niveau du type, du nombre et de la taille des paramètres ainsi que de la manière d'estimation. Par la suite, nous avons présenté les extensions, les avantages et les limites de l'approche HMM pour la classification d'images.

Dans le chapitre III suivant, nous explorons la méthode hybride et le modèle proposé CNN-HMM qui combine les réseaux neuronaux convolutifs (CNN) avec les modèles de Markov cachés (HMM).

**Références du chapitre II**

- [1] T. X. Pham, P. Siarry, and H. Oulhadj, "Segmentation of MR Brain Images Through Hidden Markov Random Field and Hybrid Metaheuristic Algorithm," *IEEE Trans. Image Process.*, vol. 29, no. XX, pp. 6507–6522, 2020.
- [2] Z. Luo *et al.*, "ContextDesc: Local Descriptor Augmentation With Cross-Modality Context," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, vol. 2019-June, pp. 2522–2531.
- [3] W. Pieczynski, "Modèles de Markov en traitements d'images Markov models in image processing," vol. 20, no. 3, pp. 255–278, 2003.
- [4] Q. Lv, S. Zhang, and Y. Wang, "Deep Learning Model of Image Classification Using Machine Learning," *Adv. Multimed.*, vol. 2022, pp. 1–12, Jul. 2022.
- [5] Z. Alom, V. K. Asari, A. Parwani, and T. M. Taha, "Microscopic nuclei classification, segmentation, and detection with improved deep convolutional neural networks (DCNN).," *Diagn. Pathol.*, vol. 17, no. 1, p. 38, Apr. 2022.
- [6] E. Xi, "Image Classification and Recognition Based on Deep Learning and Random Forest Algorithm," *Wirel. Commun. Mob. Comput.*, vol. 2022, pp. 1–9, Jun. 2022.
- [7] J. Sun, "Application of Image Segmentation Algorithm Based on Partial Differential Equation in Legal Case Text Classification," *Adv. Math. Phys.*, vol. 2021, pp. 1–9, Oct. 2021.
- [8] T. L. Athey, D. J. Tward, U. Mueller, J. T. Vogelstein, and M. I. Miller, "Hidden Markov modeling for maximum probability neuron reconstruction.," *Commun. Biol.*, vol. 5, no. 1, p. 388, Apr. 2022.
- [9] H. Li, W. Guo, J. Liu, L. Cui, and D. Xie, "Image Segmentation with Adaptive Spatial Priors from Joint Registration," *SIAM J. Imaging Sci.*, vol. 15, no. 3, pp. 1314–1344, Sep. 2022.
- [10] E. Azeraf, E. Monfrini, and W. Pieczynski, "Equivalence between LC-CRF and HMM, and Discriminative Computing of HMM-Based MPM and MAP," *Algorithms*, vol. 16, no. 3, p. 173, Mar. 2023.
- [11] M. E. Y. Boudaren, A. Hamache, I. Debicha, and H. T. Sadouk, "Dempster-Shafer Parzen-Rosenblatt Hidden Markov Fields for Multichannel Image Segmentation," in *Communications in Computer and Information Science*, vol. 1237 CCIS, 2020, pp. 613–624.
- [12] H. Wang, J. F. Wellmann, Z. Li, X. Wang, and R. Y. Liang, "A Segmentation Approach for

- Stochastic Geological Modeling Using Hidden Markov Random Fields,” *Math. Geosci.*, vol. 49, no. 2, pp. 145–177, Feb. 2017.
- [13] M. Pereyra and S. McLaughlin, “Fast Unsupervised Bayesian Image Segmentation With Adaptive Spatial Regularisation,” *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2577–2587, Jun. 2017.
- [14] R. Paroli and L. Spezia, “Digital Image Segmentation Through Bayesian Hidden Markov Models,” no. Mcmc, 1997.
- [15] K. Tumilaar, Y. Langi, and A. Rindengan, “Hidden Markov Model,” *d’CARTESIAN*, vol. 4, no. 1, p. 86, Feb. 2015.
- [16] F. Mohammadi *et al.*, “A lineage tree-based hidden Markov model quantifies cellular heterogeneity and plasticity,” *Commun. Biol.*, vol. 5, no. 1, p. 1258, Nov. 2022.
- [17] N. Awaya and L. Ma, “Supplementary Materials for ‘ Hidden Markov P ´ olya trees for high-dimensional distributions ,”” pp. 1–39.
- [18] H. Gangloff, J. Courbot, E. Monfrini, and C. Collet, “UNSUPERVISED IMAGE SEGMENTATION WITH SPATIAL TRIPLET MARKOV TREES SAMOVAR , Télécom SudParis , Institut Polytechnique de Paris , Palaiseau , France ICube , Université de Strasbourg , CNRS UMR 7357 , Illkrich , France GEPROVAS , Strasbourg , France,” pp. 1790–1794, 2021.
- [19] N. Awaya and L. Ma, “Hidden Markov Pólya Trees for High-Dimensional Distributions,” *J. Am. Stat. Assoc.*, vol. 119, no. 545, pp. 189–201, Jan. 2024.
- [20] Z. Jiang and A. M. Sainju, “Hidden Markov Contour Tree,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 804–813.
- [21] Z. Xiong, B. Li, and D. Liu, “Map-Matching Using Hidden Markov Model and Path Choice Preferences under Sparse Trajectory,” 2021.
- [22] H. İ. Turhan, D. Acar, and N. B. Ayana, “Saklı Markov Model Tabanlı Değişken Yapılı Çoklu Model Algoritması Hidden Markov Model Based Variable Structured Multiple Model Algorithm,” pp. 4–7, 2020.
- [23] B. Roblès, M. Avila, F. Duculty, P. Vignat, S. Begot, and F. Kratz, “Methods to choose the best Hidden Markov Model topology for improving maintenance policy,” *MOSIM’12 9th Int. Conf. Model. Optim. Simulation, Bordeaux Fr.*, vol. 1, 2012.
- [24] T. Zhang, Y. Xia, and D. D. Feng, “Hidden Markov random field model based brain MR

- image segmentation using clonal selection algorithm and Markov chain Monte Carlo method,” *Biomed. Signal Process. Control*, vol. 12, no. 1, pp. 10–18, Jul. 2014.
- [25] W. Pieczynski, “Pairwise markov chains,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 634–639, May 2003.
- [26] G. S. Multidimensionnels, “Segmentation non supervisée d’images par chaîne de Markov couple.”
- [27] N. E. I. Hamda, A. Hadjali, and M. Lagha, “Multisensor Data Fusion in IoT Environments in Dempster–Shafer Theory Setting: An Improved Evidence Distance-Based Approach,” *Sensors*, vol. 23, no. 11, p. 5141, May 2023.
- [28] K. Rapporteur and K. Examineur, *Centre d’Etudes Doctorales : Sciences et Techniques Formation doctorale : Mathématiques et Physiques Appliquées Remerciements*. 2020.
- [29] R. Langrock, R. King, J. Matthiopoulos, L. Thomas, D. Fortin, and J. M. Morales, “Flexible and practical modeling of animal telemetry data: hidden Markov models and extensions,” *Ecology*, vol. 93, no. 11, pp. 2336–42, Nov. 2012.
- [30] V. Dutoit, M. van der Wilk, A. Artemev, and J. Hensman, “Bayesian Image Classification with Deep Convolutional Gaussian Processes,” *Proc. Mach. Learn. Res.*, vol. 108, pp. 1529–1539, Feb. 2019.
- [31] Z. Liu, C. Ma, W. She, and M. Xie, “Biomedical Image Segmentation Using Denoising Diffusion Probabilistic Models: A Comprehensive Review and Analysis,” *Appl. Sci.*, vol. 14, no. 2, p. 632, Jan. 2024.
- [32] S. Núñez Mejía, “Hidden Markov Models for early detection of cardiovascular diseases,” *Ing. Solidar.*, vol. 20, no. 1, pp. 1–31, Dec. 2023.
- [33] M. Ameur, C. Daoui, and N. Idrissi, “Hierarchical Hidden Markov Models in Image Segmentation,” *Sci. Vis.*, vol. 12, no. 1, pp. 22–47, 2020.

---

## *Chapitre III*

### *Méthode hybride HMM-CNN pour la classification d'images*

---

### III.1 Introduction

La classification d'images est une opération importante et essentielle pour diverses applications de vision par ordinateur. Les méthodes classiques utilisées pour la classification d'images présentent certaines limitations, telles que ; le captage des dépendances spatiales et des variations locales. Avec l'essor des réseaux neuronaux convolutifs (CNN) et des modèles de Markov cachés (HMM), des approches hybrides ont été développées pour améliorer les performances de classification.

Les CNN sont particulièrement efficaces pour extraire des caractéristiques spatiales complexes et généraliser à partir de divers ensembles de données.

En revanche, les HMM excellent dans la modélisation des dépendances séquentielles entre les données. La combinaison de ces deux méthodes, dans un cadre HMM-CNN, permet d'exploiter les forces de chaque technique, aboutissant à une classification plus précise et plus robuste.

Dans ce chapitre, nous discutons au début des défis posés par ces méthodes (CNN, HMM) et de la nécessité de développer des approches hybrides combinant CNN et HMM pour surmonter ces limitations.

Dans ce cadre et pour améliorer la précision et l'efficacité de la classification d'images, nous avons proposé une méthode hybride CNN-HMM qui intègre les avantages des CNN pour l'extraction des caractéristiques de l'image et des HMM pour la modélisation des dépendances spatiales de l'image.

Nous allons explorer la méthode hybride HMM-CNN proposée pour la classification d'images. Nous discuterons des avantages, des limitations ainsi que de l'architecture et des algorithmes utilisés dans cette approche.

### III.2 Méthodes de classification d'images

La classification d'images, un aspect crucial de la vision par ordinateur, a considérablement progressé avec l'essor de l'apprentissage profond, en particulier grâce à l'utilisation de modèles préentraînés de réseau de neurones convolutifs (CNN) [1][2][3]. Les CNN se sont avérés très efficaces dans les tâches de catégorisation d'images, surpassant d'autres modèles comme Auto Encoder, Sparse Coding et Restricted Boltzmann Machine [4].

Le succès des CNN réside dans leur capacité à apprendre des représentations de données complexes à travers plusieurs couches de traitement, ce qui leur permet d'atteindre des performances de pointe dans diverses applications, y compris la classification d'images [5].

Alors que les modèles de Markov cachés (HMM) ont traditionnellement été utilisés pour la segmentation d'images en raison de leur capacité à modéliser des dépendances temporelles complexes et des données séquentielles, la récente prédominance des NC dans ce domaine démontre leur rendement supérieur et leur adoption généralisée dans les tâches de classification d'images [6].

### III.2.1 CNN pour les méthodes de classification d'images

Les réseaux de neurones convolutionnels (CNN) sont utilisés pour extraire des caractéristiques d'images et les utiliser pour la classification. Ils sont efficaces pour la reconnaissance d'objets et l'analyse d'image.

Les FCN sont un type de CNN qui ont une architecture encodeur-décodeur. La partie encodeur se compose de couches convolutionnelles et de mise en commun, tandis que la partie décodeur se compose de couches dé-convolutionnelles.

Les FCN produisent une prédiction dense en pixels de l'image d'entrée, qui peut être seuillée pour générer un masque de segmentation binaire. U-Net est une extension de l'architecture FCN qui inclut des connexions sautées entre l'encodeur et les parties du décodeur.

Ces connexions permettent au décodeur d'accéder aux informations des couches précédentes de l'encodeur, ce qui peut améliorer la précision de la segmentation.

SegNet est une autre architecture encodeur-décodeur qui inclut des indices de mise en commun dans la partie encodeur. Ces indices sont utilisés dans la partie décodeur pour effectuer un suréchantillonnage, ce qui réduit le nombre de paramètres nécessaires par rapport à d'autres architectures.

Ce sont des réseaux convolutifs dilatés. Ils utilisent des couches convolutives avec des taux de dilatation plus rapides pour agrandir le champ réceptif du réseau tout en gardant la taille du filtre petite. Cela peut améliorer la capacité du réseau à capturer l'information et augmenter la précision de la segmentation.

Les réseaux multi-échelles combinent plusieurs CNN avec différents champs réceptifs pour capturer des informations à différentes échelles. Cela peut améliorer la capacité du réseau à capturer des informations locales et globales et améliorer la précision de la segmentation.

Ces méthodes ne sont que quelques exemples d'approches basées sur CNN pour la segmentation d'images, et de nombreuses autres variantes et extensions ont été proposées. Le choix de la méthode dépend de l'application spécifique et des exigences de précision de segmentation, de vitesse et d'utilisation de la mémoire.

### III.2.2 HMM pour les méthodes de classification d'images

Les Modèles de Markov Cachés (HMM) sont utilisés pour modéliser les dépendances temporelles entre les caractéristiques extraites d'images. Ils sont utiles pour la classification des séquences d'images.

Les méthodes de classification d'images basées sur HMM sont particulièrement utiles pour traiter des scènes dynamiques ou des vidéos, où la classification doit être effectuée sur une séquence d'images.

L'idée de base derrière la classification d'images basée sur HMM est de modéliser la séquence d'images comme un processus de Markov, où chaque image est un état et les transitions entre les états sont régies par un ensemble de probabilités. Ces probabilités peuvent être apprises à partir des données d'entraînement à l'aide d'algorithmes tels que le Baum-Welch algorithm.

Dans la classification d'image basée sur HMM, l'image est d'abord représentée à l'aide d'un ensemble de fonctionnalités, telles que la couleur, la texture ou la forme. Le HMM est ensuite utilisé pour modéliser les dépendances temporelles entre les caractéristiques, et la séquence d'images est segmentée en trouvant la séquence d'états la plus probable compte tenu des caractéristiques observées.

Les méthodes de classification d'images basées sur HMM présentent plusieurs avantages, notamment la capacité à gérer des données bruyantes ou incomplètes et la capacité à modéliser des dépendances temporelles complexes. Cependant, ils peuvent être intensifs en calcul et peuvent nécessiter des quantités importantes de données d'entraînement pour atteindre une grande précision.

### III.3 Méthode hybride HMM-CNN

L'approche hybride HMM-CNN combine les forces des réseaux de neurones convolutifs (CNNs) et des modèles de Markov cachés (HMMs) pour améliorer les tâches de classification d'images (figure III.1).

Dans cette approche, les CNN sont utilisés pour l'extraction de caractéristiques et la réduction de la dimensionnalité, tandis que les HMM sont utilisés pour la classification.

Le modèle CNN extrait les cartes d'entités à travers des couches convolutives et de mise en commun, tandis que les HMM sont obtenues à l'aide d'un balayage Peano. L'algorithme d'espérance-maximisation (EM) est ensuite appliqué pour estimer les paramètres des HMM, permettant une classification non supervisée par la méthode du mode bayésien maximum postérieur (MPM).

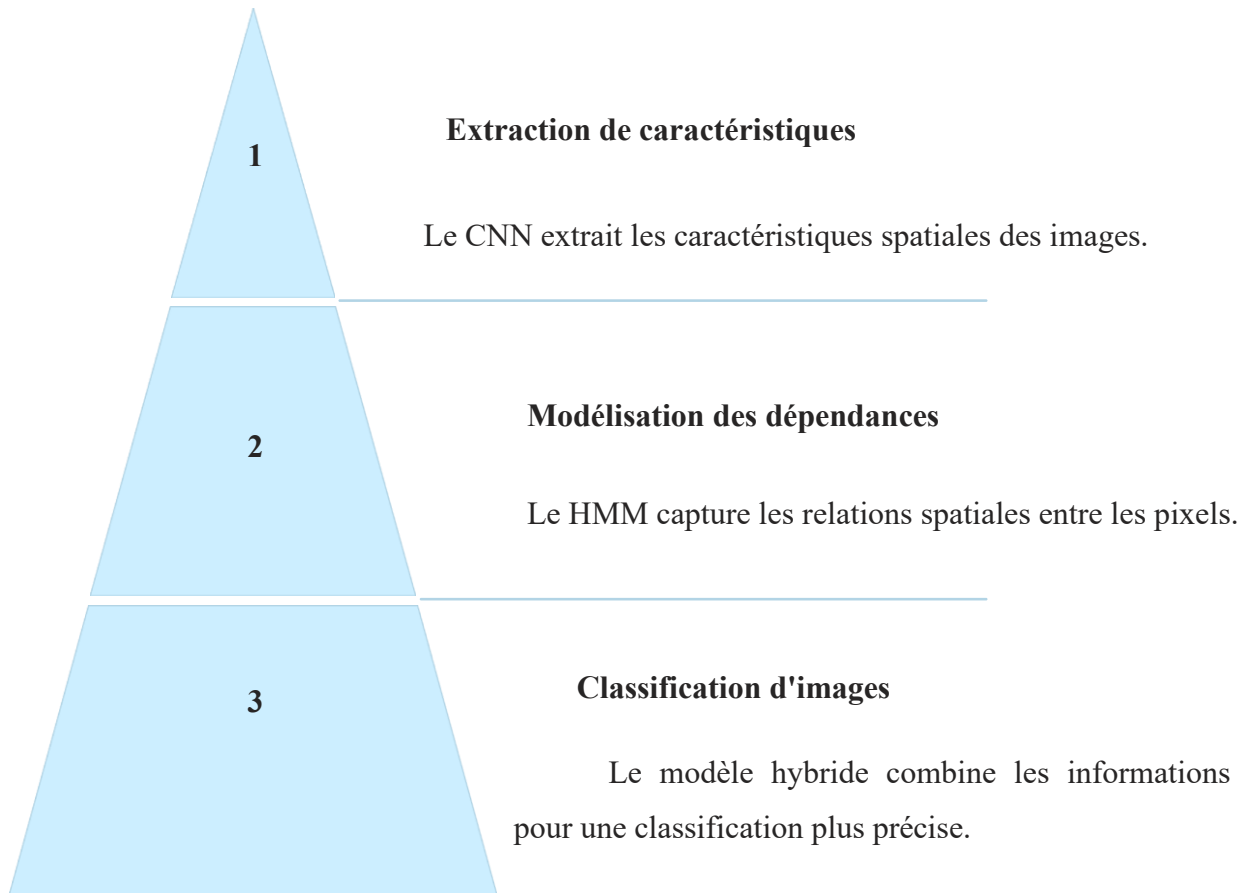


Figure III.1 : Processus d'une approche hybride HMM-CNN

### III.3.1 Avantages de HMM-CNN hybride

La combinaison de modèles de Markov cachés (MMH) et de réseaux de neurones convolutifs (CNN) offre plusieurs avantages dans divers domaines.

Les MMH offrent des capacités de modélisation de séquence, tandis que les CNN excellent à l'extraction d'entités et à la réduction de la dimensionnalité [7][8]. Cette combinaison améliore les capacités discriminantes des CNN avec la modélisation probabiliste des MMH, ce qui améliore les performances dans des tâches telles que la classification d'images et la reconnaissance de la langue des signes [9][1]. L'intégration des MMH avec les CNN permet de gérer efficacement des séquences de longueur variable et de petits corpus, en répondant aux principaux défis techniques de la paralinguistique informatique et de la classification d'images de scènes par télédétection. De plus, les modèles hybrides CNN-HMM ont montré des résultats supérieurs à ceux des modèles CNN autonomes, démontrant l'efficacité de l'exploitation des deux approches pour une précision et une robustesse accrue dans diverses applications.

### III.3.2 Limitations de HMM-CNN hybride

Cependant, l'efficacité de la méthode hybride dépend de divers facteurs tels que :

- La complexité d'images et la qualité des données d'entraînement (complexité accrue due à la combinaison de deux modèles).
- Nécessite plus de ressources en termes de calcul et de mémoire.
- Plus difficile à optimiser et à entraîner le modèle hybride.
- Sensibilité aux données bruyantes ou manquantes.

Des recherches supplémentaires sont nécessaires pour évaluer sa performance dans différents types d'applications.

### III.4 Architecture et algorithmes utilisés

#### III.4.1 Architecture

L'architecture du CNN-HMM est essentiellement une combinaison de deux modèles puissants : les réseaux de neurones convolutifs (CNN) et les modèles de Markov cachés (HMM).

Le CNN est utilisé pour extraire des caractéristiques d'images, tandis que le HMM est utilisé pour modéliser les dépendances spatiales entre les pixels adjacents. Cette approche hybride vise à améliorer la performance de classification d'images en combinant les forces des deux modèles.

Le CNN extrait des caractéristiques pertinentes d'images, tandis que le HMM modélise les relations spatiales entre les pixels adjacents pour une compréhension plus approfondie des données d'image. En combinant ces deux modèles de manière hybride, on obtient des résultats qui sont utiles dans de nombreux contextes et qui apportent des insights précieux dans le domaine de la classification d'images.

Cette combinaison permet de capturer les caractéristiques hiérarchiques et spatiales extraites par le CNN ainsi que les dépendances temporelles modélisées par le HMM.

Le modèle est composé de 18 couches, dont des couches de convolution, de pooling, de normalisation, de dropout, de flattening, et de densité. Chaque couche est décrite avec ses paramètres correspondants dans le tableau :

Tableau III .1. Exemple représentatif d'une architecture CNN

Couche (Type)	Forme de sortie
Conv2d (Conv2D)	(None, 32, 32, 32)
Batch_Normalization	(BatchNo (None, 32, 32, 32))
Conv2d_1 (Conv2D)	(None, 32, 32, 32)
Batch_Normalization_1	(Batch (None, 32, 32, 32))
Max_Pooling2d	(None, 16, 16, 32)
Dropout (Dropout)	(None, 16, 16, 32)
Conv2d_2 (Conv2D)	(None, 16, 16, 64)
Batch_Normalization_2	(Batch (None, 16, 16, 64))
Conv2d_3 (Conv2D)	(None, 16, 16, 64)
Batch_Normalization_3	(Batch (None, 16, 16, 64))
Max_Pooling2d_1	(None, 8, 8, 64)
Dropout_1 (Dropout)	(None, 8, 8, 64)
Conv2d_4 (Conv2D)	(None, 8, 8, 128)
Batch_Normalization_4	(Batch (None, 8, 8, 128))
Conv2d_5 (Conv2D)	(None, 8, 8, 128)
Batch_Normalization_5	(Batch (None, 8, 8, 128))
Max_Pooling2d_2	(None, 4, 4, 128)
Dropout_2 (Dropout)	(None, 4, 4, 128)
Flatten (Flatten)	(None, 2048)
Dense (Dense)	(None, 128)
Dropout_3 (Dropout)	(None, 128)
Dense_1 (Dense)	(None, 10)
Total Des Paramètres	-
Paramètres Entraînables	-
Paramètres Non Entraînables	-

### III.4.2 Algorithme

La figure III.2 illustre une représentation de l'algorithme global de l'approche hybride CNN-HMM proposée pour la classification d'images.

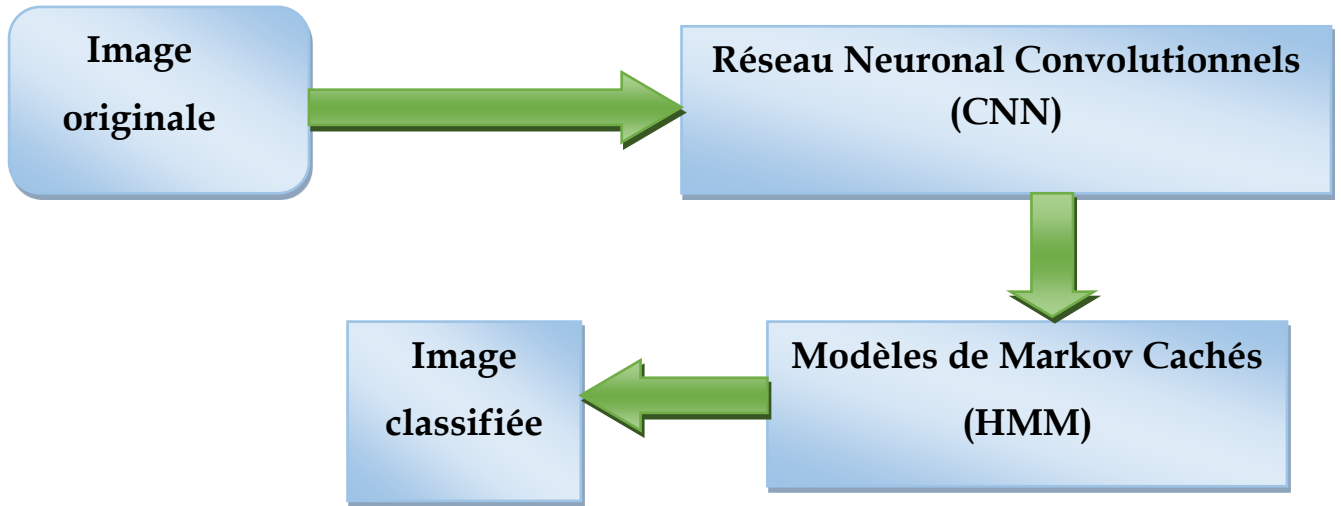


Figure III.2 : Algorithme global de l'approche hybride CNN-HMM proposée [10].

Dans l'ensemble, ce diagramme montre comment l'information circule à travers différentes étapes d'extraction de caractéristiques (CNN) et de modélisation temporelle (HMM) pour améliorer la précision de la classification d'images en exploitant à la fois les capacités d'extraction des caractéristiques spatiales des CNN et les dépendances temporelles modélisées par les HMM.

Dans la partie suivante, nous allons expliquer en détail les différentes étapes de l'algorithme de la méthode hybride CNN-HMM proposée :

- **Image originale**

Le processus débute par l'acquisition ou le chargement de l'image originale en entrée dans le système.

- **Réseau de neurones convolutifs**

En parallèle, les caractéristiques extraites sont également traitées par un CNN. Les CNN sont très efficaces pour l'analyse d'images car ils peuvent apprendre automatiquement et de manière adaptative les hiérarchies spatiales des caractéristiques, des bords de bas niveau aux structures complexes. Le CNN améliore l'ensemble de fonctionnalités en fournissant une représentation hiérarchique profonde de l'image.

### - Modèle de Markov caché (HMM)

Les caractéristiques extraites sont ensuite introduites dans un HMM. Les HMM sont utilisées pour modéliser les dépendances temporelles et contextuelles dans les données. Dans le contexte de la classification d'images, les HMM peuvent capturer les motifs séquentiels et les relations spatiales entre les caractéristiques extraites.

### - Classification d'images

Les fonctionnalités raffinées et mises à jour, éclairées par le traitement HMM et CNN, sont ensuite utilisées pour classer l'image. La combinaison de HMM et CNN permet une analyse plus complète, en tirant parti à la fois des dépendances séquentielles et des représentations profondes des fonctionnalités.

## III.5 Méthode proposée pour la classification d'images

La méthode proposée consiste à préparer les images, extraire les caractéristiques avec un CNN pré-entraîné, mapper les états cachés et les observations pour l'apprentissage des HMM, et classifier les images en utilisant le modèle hybride HMM-CNN.

Dans cette section, nous détaillerons les différentes étapes de la méthode hybride CNN-HMM, incluant l'extraction des caractéristiques par les CNN et l'entraînement des HMM pour modéliser les dépendances spatiales entre les pixels. Nous expliquerons également comment ces deux approches sont combinées pour améliorer la performance de la classification et de la segmentation d'images.

La figure III.3 illustre le processus étape par étape de l'approche hybride proposée, qui combine les CNN et les MMH pour la classification d'images.

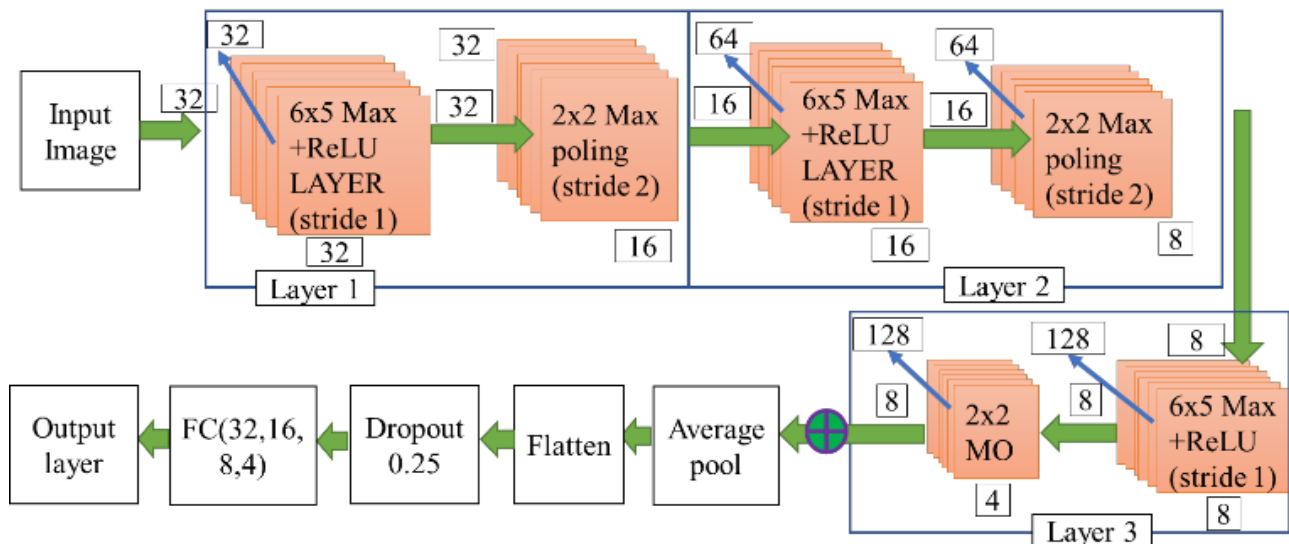


Figure III.3 : Diagramme de la méthode proposée [10].

**Étape 1 : Préparation de l'image**

La première étape consiste à prétraiter les images d'entrée, cela inclut des tâches telles que le redimensionnement d'images à une taille cohérente et la normalisation des valeurs de pixels. Ces étapes de prétraitement garantissent que toutes les images ont le même format et la même gamme de valeurs, ce qui est important pour une extraction efficace des caractéristiques.

**Étape 2 : Extraction des caractéristiques CNN**

Dans cette étape, un modèle CNN pré-entraîné (dans la figure 1) est utilisé pour extraire des caractéristiques de haut niveau d'images prétraitées. Le modèle CNN comporte plusieurs couches convolutives, des couches de regroupement (pooling) et des couches entièrement connectées qui lui permettent d'apprendre les caractéristiques spatiales pertinentes à partir d'images d'entrée.

**Étape 3 : Représentation des caractéristiques**

Les caractéristiques extraites par le CNN sont ensuite aplaties ou regroupées dans une représentation vectorielle de dimension fixe pour chaque image. Cette représentation capture des informations importantes sur les caractéristiques visuelles de l'image et sert d'entrée pour un traitement ultérieur.

**Étape 4 : Mappage des états et observation HMM**

Dans cette étape, un ensemble d'états cachés est défini dans le modèle HMM, représentant différentes classes cibles pour la classification d'images (par exemple, avion, automobile, oiseau). Chaque vecteur caractéristique extrait de l'étape 3 est mappé à un symbole d'observation qui représente des observations discrètes à utiliser dans le HMM.

**Étape 5 : Entraînement HMM**

L'étape suivante consiste à entraîner (former) le HMM en utilisant des données d'entraînement étiquetées. Le processus d'entraînement implique l'estimation des probabilités de transition entre les états cachés en fonction des vecteurs de caractéristiques observés. Cela permet de capturer les dépendances temporelles entre les pixels adjacents au sein de chaque classe.

**Étape 6 : Classification à l'aide du modèle hybride**

Pour chaque image d'un ensemble de tests ou de nouvelles données non vues après la fin de l'entraînement, le modèle CNN préentraîné extrait sa représentation vectorielle de caractéristiques (comme fait précédemment à l'étape 3). Puis ce vecteur de caractéristiques est passé à travers le HMM entraîné en utilisant l'algorithme de Viterbi, qui trouve la séquence la plus probable.

**Étape 7 : Évaluation**

Calculez des métriques d'évaluation telles que, la précision, le rappel et le score F1 pour évaluer la performance du modèle hybride HMM-CNN sur l'ensemble de test.

**Étape 8 : Ajustement et optimisation**

Ajustez les hyperparamètres du CNN et du HMM, tels que le taux d'apprentissage, le nombre de couches, les tailles de filtre et les probabilités de transition.

Affinez et ajustez itérativement le modèle hybride pour obtenir une meilleure précision et la généralisation.

**III.6 Performance du modèle**

Une évaluation empirique du modèle hybride HMM-CNN sur des ensembles de données de référence montre une amélioration de la précision de la classification d'images par rapport aux CNN traditionnels. Les résultats des tests, y compris la perte, l'exactitude, la précision et le rappel, démontrent l'efficacité de l'approche hybride dans le domaine de la classification et la segmentation d'images.

Il convient de noter que la mise en œuvre d'une approche hybride HMM-CNN pour la classification d'images peut être complexe et que les performances ne peuvent pas toujours dépasser celles de l'utilisation des CNN seuls. Les CNN sont généralement le choix privilégié pour les tâches de classification d'images en raison de leur capacité à apprendre directement des motifs spatiaux complexes à partir d'images. L'approche hybride vise à intégrer les dépendances temporelles dans le processus de classification, en tirant parti des capacités de modélisation séquentielle des HMM.

**III.7 Conclusion**

L'objectif principal de ce chapitre était d'étudier l'efficacité d'un modèle hybride combinant des CNNs et des HMMs pour la classification d'images.

L'objectif était d'exploiter les capacités d'extraction de caractéristiques spatiales des CNN et la modélisation temporelle.

Dans ce cadre, nous avons présenté au début une discussion sur les différentes méthodes et approches les plus utilisées dans le domaine de la classification d'images, telles que, CNN et HMM. Les réseaux de neurones convolutifs (CNN) en raison de leur capacité à extraire des caractéristiques spatiales complexes, ainsi que des modèles de Markov cachés (HMM) utilisés pour la représentation de séquences temporelles et d'interrelations contextuelles.

Par la suite, une approche hybride HMM-CNN qui explore l'intégration de convolutional neural networks (CNN) et de hidden markov models (HMM), ainsi qu'une solution hybride prometteuse qui combine ces deux approches dans la classification d'images. Malgré une complexité et des exigences de calcul accrues, cette méthodologie offre une précision et une adaptabilité accrues.

Le chapitre a ensuite décrit le cadre et les procédures de calcul utilisés pour fusionner HMM et CNN, aboutissant à la proposition d'une technique spécifique de catégorisation d'images basée sur cette architecture intégrée. L'évaluation des performances du modèle qui a suivi a révélé une amélioration substantielle par rapport aux techniques conventionnelles traditionnelles.

En résumé, ce chapitre fournit un examen complet et une méthode hybride CNN-HMM inventive qui combine les avantages des CNN et des HMM pour faciliter une classification d'images plus efficace.

Dans le chapitre suivant, nous présentons une discussion sur les résultats obtenus de la méthode CNN-HMM proposée.

**Références du chapitre III**

- [1] S. Goumiri, D. Benboudjema, and W. Pieczynski, “A new hybrid model of convolutional neural networks and hidden Markov chains for image classification,” *Neural Comput. Appl.*, vol. 35, no. 24, pp. 17987–18002, Aug. 2023.
- [2] B. Olimov, B. Subramanian, R. A. A. Ugli, J.-S. Kim, and J. Kim, “Consecutive multiscale feature learning-based image classification model,” *Sci. Rep.*, vol. 13, no. 1, p. 3595, Mar. 2023.
- [3] S. Wang, “Research on Computer Vision Image Classification,” *Acad. J. Comput. Inf. Sci.*, vol. 6, no. 6, pp. 26–30, 2023.
- [4] J. Luo and D. Hu, “An Image Classification Method Based on Adaptive Attention Mechanism and Feature Extraction Network,” *Comput. Intell. Neurosci.*, vol. 2023, pp. 1–11, Feb. 2023.
- [5] “Stochastic Study on CNN Approach for Classifying Images,” *Int. J. FOOD Nutr. Sci.*, vol. 11, no. 4, pp. 135–143, Jan. 2023.
- [6] X. Cao, F. Zhou, L. Xu, D. Meng, Z. Xu, and J. Paisley, “Hyperspectral Image Classification With Markov Random Fields and a Convolutional Neural Network,” *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2354–2367, May 2018.
- [7] M. Vetráb and G. Gosztolya, “Using Hybrid HMM/DNN Embedding Extractor Models in Computational Paralinguistic Tasks,” *Sensors (Basel)*, vol. 23, no. 11, p. 5208, May 2023.
- [8] O. Koller, S. Zargaran, H. Ney, and R. Bowden, “Deep Sign: Enabling Robust Statistical Continuous Sign Language Recognition via Hybrid CNN-HMMs,” *Int. J. Comput. Vis.*, vol. 126, no. 12, pp. 1311–1325, Dec. 2018.
- [9] A. O. Tur and H. Y. Keles, “Evaluation of hidden Markov models using deep CNN features in isolated sign recognition,” *Multimed. Tools Appl.*, vol. 80, no. 13, pp. 19137–19155, May 2021.
- [10] A. Djalab, L. Lalaoui, A. Bisker, and A. Hadibi, “Enhancing Image Classification Through a Hybrid Approach: Integrating Convolutional Neural Networks with Hidden Markov Mod,” *Trait. du Signal*, vol. 41, no. 1, pp. 383–390, Feb. 2024.

---

## *Chapitre IV*

### *Implémentation et évaluation de CNN-HMM pour la classification d'images*

---

### IV.1 Introduction

L'objectif principal de cette recherche était d'étudier l'efficacité d'un modèle hybride combinant des CNNs et des HMMs pour la classification d'images.

L'objectif était de tirer parti des capacités d'extraction de caractéristiques spatiales des CNN et des forces de modélisation temporelle des MMH pour obtenir une compréhension plus complète des données d'image.

Dans ce chapitre, nous présenterons les résultats des tests réalisés en utilisant la méthode hybride CNN-HMM pour la classification d'images sur le Jeu de Données CIFAR-10.

Nous discuterons des métriques d'évaluation telles que l'exactitude, la précision, le rappel et le score F1 pour évaluer la performance du modèle hybride sur l'ensemble de test.

### IV.2 Développement et application sur le Jeu de Données CIFAR-10

L'ensemble de données CIFAR-10 est une référence largement utilisée dans l'évaluation des algorithmes et des modèles de classification d'images, contribuant de manière significative aux progrès de la vision par ordinateur et de la recherche en apprentissage profond.

Les chercheurs utilisent cet ensemble de données pour évaluer la précision et les capacités de généralisation des modèles d'apprentissage automatique lors de la classification d'images dans dix catégories différentes. En formant et en évaluant des modèles de pointe comme les réseaux de neurones convolutifs (CNN) sur CIFAR-10, des niveaux de précision élevés ont été atteints dans les tâches de classification d'images.

La popularité de l'ensemble de données provient de son efficacité à fournir une plate-forme normalisée pour comparer les performances de divers algorithmes, aidant à l'élaboration de modèles robustes et efficaces pour les tâches de classification d'images [1][2].

Dans cette section, je décrirai l'architecture du modèle CNN. Je l'ai développé pour la classification d'images et démontré son application sur le jeu de données CIFAR-10.

Le modèle est implémenté en utilisant l'API Keras, qui est construite sur TensorFlow. L'architecture du modèle se compose de plusieurs couches convolutionnelles, suivies de couches de normalisation par lots et de couches de max-pooling, qui aident à améliorer la stabilité du processus d'entraînement et à réduire le surapprentissage.

Le dropout est également implémenté pour prévenir davantage le surapprentissage. La couche entièrement connectée à la fin du modèle comporte un grand nombre de neurones, ce qui augmente la capacité du modèle et lui permet d'apprendre des caractéristiques plus complexes. Le modèle est entraîné en utilisant l'optimiseur Adam et la fonction de perte catégorielle d'entropie croisée.

En entraînant le modèle sur le jeu de données CIFAR-10 pendant plusieurs époques, le modèle apprend à classifier précisément les images en 10 classes différentes. Globalement, la combinaison de l'architecture CNN, de la normalisation par lots, du dropout et des techniques d'optimisation rend le modèle très efficace pour les tâches de classification d'images.

### IV.3 Logiciels et outils

#### IV.3.1 Python

Python est un langage de programmation de haut niveau, interprété, connu pour sa simplicité, sa lisibilité et sa polyvalence. Il a été créé par Guido van Rossum et publié pour la première fois en 1991.

La philosophie de conception de Python met l'accent sur la lisibilité du code, ce qui permet aux développeurs d'exprimer des concepts clairement et de manière concise.

L'une des caractéristiques déterminantes de Python est sa syntaxe claire et intuitive, qui permet aux programmeurs d'écrire du code de manière plus naturelle et expressive [3].

Cette lisibilité améliore non seulement la productivité des développeurs, mais facilite également la collaboration entre les équipes. Python fournit une vaste bibliothèque standard qui offre une large gamme de modules et de fonctions pour diverses tâches, telles que la gestion de fichiers, la mise en réseau, le développement Web, la manipulation de données, et bien plus encore. De plus, Python possède un écosystème florissant de bibliothèques et de frameworks tiers qui étendent ses capacités, permettant aux développeurs d'aborder des domaines spécialisés tels que l'apprentissage automatique, le calcul scientifique et l'analyse de données [4].

La nature interprétée de Python permet un développement et un prototypage rapides. Il prend en charge plusieurs paradigmes de programmation, y compris la programmation procédurale, la programmation orientée objet et la programmation fonctionnelle, offrant aux développeurs la flexibilité de choisir l'approche qui convient le mieux à leurs besoins.

La force de Python réside dans sa compatibilité multiplateforme, permettant au code de s'exécuter de manière transparente sur différents systèmes d'exploitation.

Cette portabilité fait de Python un choix populaire pour le développement d'applications qui doivent fonctionner sur différentes plates-formes et environnements.

Python possède une communauté importante et solidaire qui contribue à sa croissance et à son développement. La communauté fournit une documentation exhaustive, des tutoriels et des forums où les programmeurs peuvent demander de l'aide, partager des connaissances et collaborer sur des projets open source [5].

En résumé, Python est un langage de programmation polyvalent qui privilégie la simplicité, la lisibilité et la facilité d'utilisation. Sa vaste bibliothèque standard, son écosystème diversifié, sa compatibilité multiplateforme et sa communauté solidaire en font un choix populaire pour un large éventail d'applications et de niveaux de compétence.

### IV.3.2 Architecture du modèle CNNs

La figure IV.1, illustre l'architecture d'un modèle de réseau de neurones convolutifs (CNN) utilisé pour classer les images du jeu de données CIFAR-10.

Ce jeu de données est largement utilisé pour tester et évaluer les algorithmes de classification d'images.



Figure IV.1 : Jeu de données CIFAR-10.

L'architecture du modèle CNN se compose de plusieurs couches, chacune jouant un rôle spécifique dans le traitement d'images d'entrée.

- **Convolutionnelle (Conv2D)** : Applique des filtres pour extraire les caractéristiques d'images.
- **Normalisation par lots (Batch Normalization)** : Normalise les sorties des couches convolutionnelles pour accélérer l'entraînement.
- **Max-Pooling (MaxPooling2D)** : Réduit la dimensionnalité des données en ne conservant que les valeurs maximales dans les régions de l'image.
- **Abandon (Dropout)** : Evite le surapprentissage en désactivant aléatoirement des neurones pendant l'entraînement.
- **Aplatissement (Flatten)** : Convertit les données 2D en un vecteur 1D avant l'entrée dans les couches entièrement connectées.
- **Dense (Dense)** : Couches entièrement connectées qui effectuent la classification finale d'images.

Le tableau IV.1 présente un résumé complet de l'architecture du modèle, fournissant des informations détaillées sur chaque couche et ses paramètres correspondants. Veuillez-vous référer au tableau pour une représentation visuelle de l'architecture du modèle.

Tableau IV.1 : Architecture du modèle utilisé pour mon test

Couche (Type)	Forme de sortie	Nombre de paramètres
Conv2d (Conv2D)	Aucun, 32, 32, 32	896
Normalisation de lot	(Aucun, 32, 32, 32)	128
Conv2d_1 (Conv2D)	(Aucun, 32, 32, 32)	9248
Normalisation de lot_1	(Aucun, 32, 32, 32)	128
Maximalisation de la mise en commun	(Aucun, 16, 16, 32)	0
Abandon (Dropout)	(Aucun, 16, 16, 32)	0
Conv2d_2 (Conv2D)	(Aucun, 16, 16, 64)	18496
Normalisation de lot_2	(Aucun, 16, 16, 64)	256
Conv2d_3 (Conv2D)	(Aucun, 16, 16, 64)	36928
Normalisation de lot_3	(Aucun, 16, 16, 64)	256
Maximalisation de la mise en commun_1	(Aucun, 8, 8, 64)	0
Abandon_1 (Dropout)	(Aucun, 8, 8, 64)	0
Conv2d_4 (Conv2D)	(Aucun, 8, 8, 128)	73856
Normalisation de lot_4	(Aucun, 8, 8, 128)	512
Conv2d_5 (Conv2D)	(Aucun, 8, 8, 128)	147584
Normalisation de lot_5	(Aucun, 8, 8, 128)	512
Maximalisation de la mise en commun_2	(Aucun, 4, 4, 128)	0

Abandon_2 (Dropout)	(Aucun, 4, 4, 128)	0
Aplatissement (Flatten)	(Aucun, 2048)	0
Dense	(Aucun, 128)	262272
Abandon_3 (Dropout)	(Aucun, 128)	0
Dense_1	(Aucun, 10)	1290
Total paramètres	552362	
Paramètres entraînaibles	551466	
Paramètres non entraînaibles	896	

Le modèle étudié se compose de 18 couches au total, incluant :

- 6 couches convolutionnelles
- 3 couches de max-pooling,
- 6 couches de normalisation par lots,
- 3 couches de dropout,
- 1 couche de flattening et 2 couches denses.

Chaque couche a des paramètres spécifiques (nombre de filtres, taille de filtre, etc.). Par exemple, la première couche convolutionnelle (conv2d) a une sortie de forme (32, 32, 32) avec 896 paramètres, et ainsi de suite pour chaque couche.

L'image d'entrée, qui a une taille de 32x32 pixels, passe à travers ces couches de manière séquentielle pour être classée dans l'une des 10 classes du jeu de données CIFAR-10.

Le processus commence par l'extraction des caractéristiques à travers les couches convolutionnelles, suivie de la réduction de dimensionnalité par max-pooling, la régularisation par dropout, et se termine par la classification à l'aide des couches denses.

### IV.3.3 Hardware

Le matériel utilisé pour exécuter les modèles Convolutional Neural Network (CNN) et Hidden Markov Model (HMM) a un impact significatif sur les performances.

Les unités de traitement graphique (GPU) sont couramment utilisées pour accélérer les processus de formation et d'inférence en raison de leur architecture adaptée aux opérations CNN, améliorant ainsi l'efficacité globale du système. Bien que les GPU soient connus pour leur grande précision.

Le Tableau IV.2 donne des détails sur le matériel utilisé pour mettre en œuvre l'ensemble du projet.

Tableau IV.2 : Matériel utilisé pour exécuter les tests

<b>CPU</b>	Intel(R) Core (TM) i3-10100F CPU @ 3.60GHz 3.60 GHz
<b>GPU</b>	Nvidia GeForce GTX 1050 Ti
<b>Memory (RAM)</b>	16.0 GB
<b>OS</b>	Windows 11 64-bit

### IV.4 Jeu de données (Data set)

L'ensemble de données CIFAR-10 est une ressource essentielle dans l'évaluation des algorithmes et des modèles de classification d'images, jouant un rôle crucial et une référence importante dans l'avancement de la vision par ordinateur et de la recherche en apprentissage profond.

Les chercheurs utilisent cet ensemble de données pour évaluer la précision et les capacités de généralisation des modèles d'apprentissage automatique dans différentes catégories d'images, ce qui contribue de manière significative au développement de modèles de pointe comme les CNN.

Des études ont montré que la performance de ces modèles peut varier d'une classe à l'autre en raison de caractéristiques partagées entre des catégories similaires.

L'importance de l'ensemble de données réside dans sa capacité à former et à évaluer efficacement les modèles, à améliorer les tâches de classification d'images et à favoriser les progrès dans le domaine de la vision par ordinateur et de l'apprentissage profond.

Pour lancer mes tests, j'ai commencé par visualiser un échantillon sélectionné au hasard dans l'ensemble de données. Veuillez consulter la figure IV.2 pour une illustration de cet échantillon aléatoire. Dans ce code, nous allons construire un modèle CNN qui peut classer d'images de divers objets. Nous avons 10 classes d'images, telles que : avion, automobile, oiseau, chat, cerf, chien, grenouille, cheval, bateau et camion.



Figure IV.2 : Echantillon aléatoire de l'ensemble de données.

#### IV.4.1 Section d'entraînement

Lors de l'exécution du code, certains résultats ont été atteints. La figure IV.3 présente un résumé de la répartition des classes dans l'ensemble d'entraînement (training set).

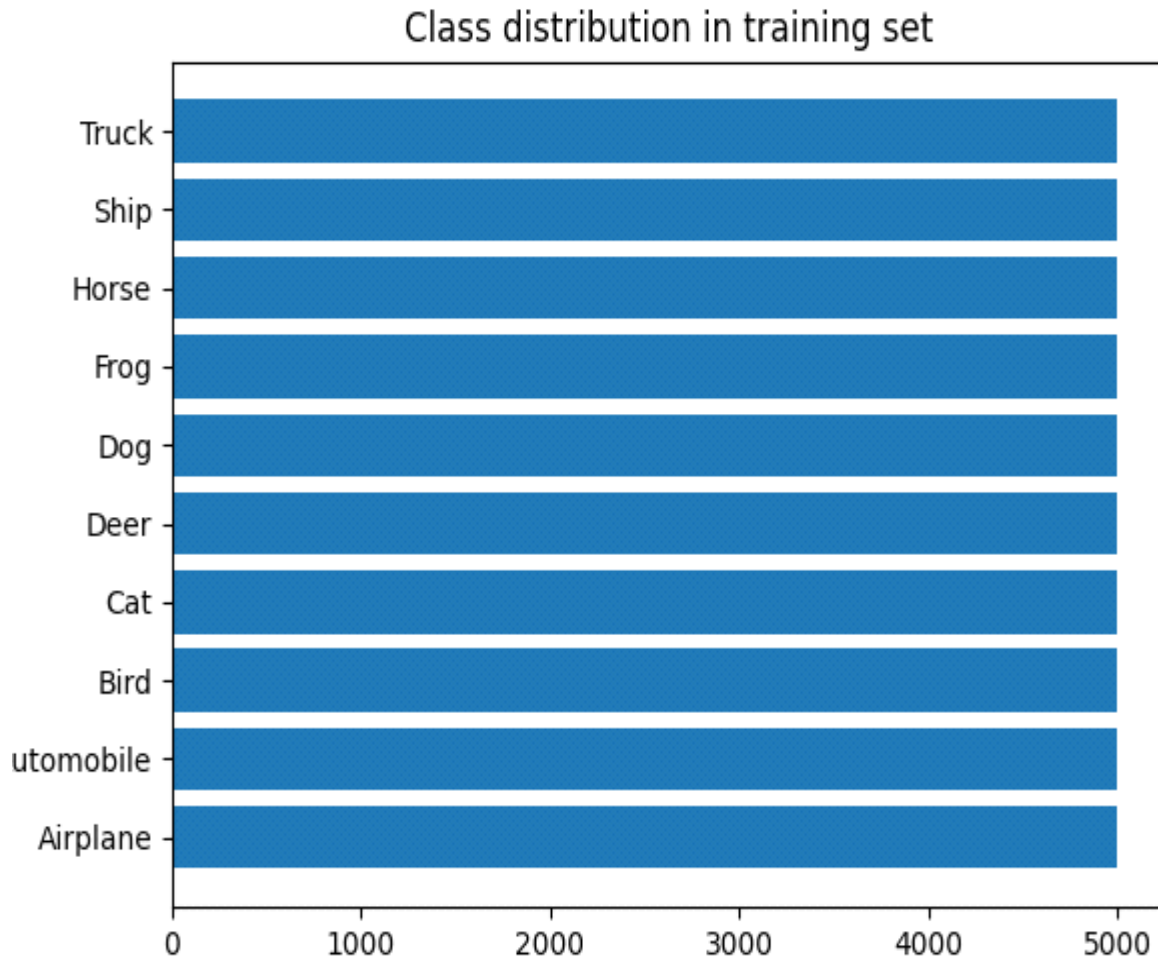


Figure IV.3. Répartition des classes dans l'ensemble d'entraînement (training set)

Les lignes fournies représentent l'évolution de l'entraînement et de la validation d'un modèle sur 50 époques. Tout au long du processus d'entraînement, les performances du modèle sont évaluées à l'aide de diverses métriques telles que la perte, la précision et le rappel.

Au début, lors de la première époque, le modèle a atteint une précision relativement faible de 0.4095 sur le jeu de données d'entraînement, indiquant qu'il avait initialement du mal à faire des prédictions précises. Cependant, la précision notable de 0.6303 suggère que lorsque le modèle faisait des prédictions, elles étaient relativement fiables. Le rappel, qui mesure la capacité du modèle à identifier correctement les instances positives, était de 0.1953, indiquant un besoin d'amélioration.

Au fur et à mesure de l'entraînement, les performances du modèle se sont progressivement améliorées. À la 50ème époque, le modèle a atteint une précision significativement plus élevée de 0.8755 sur le jeu de données d'entraînement, indiquant qu'il a appris à faire des prédictions plus

précises au fil du temps. Tant la précision (0.9091) que le rappel (0.8459) étaient relativement élevés, suggérant que le modèle identifiait et classait efficacement les instances positives.

Dans l'ensemble, les performances du modèle ont progressé positivement tout au long du processus d'entraînement, démontrant une capacité croissante à faire des prédictions précises et à classer les instances positives.

### IV.4.2 Section ensemble de test

Tandis que la figure IV.4 illustre la répartition des classes.

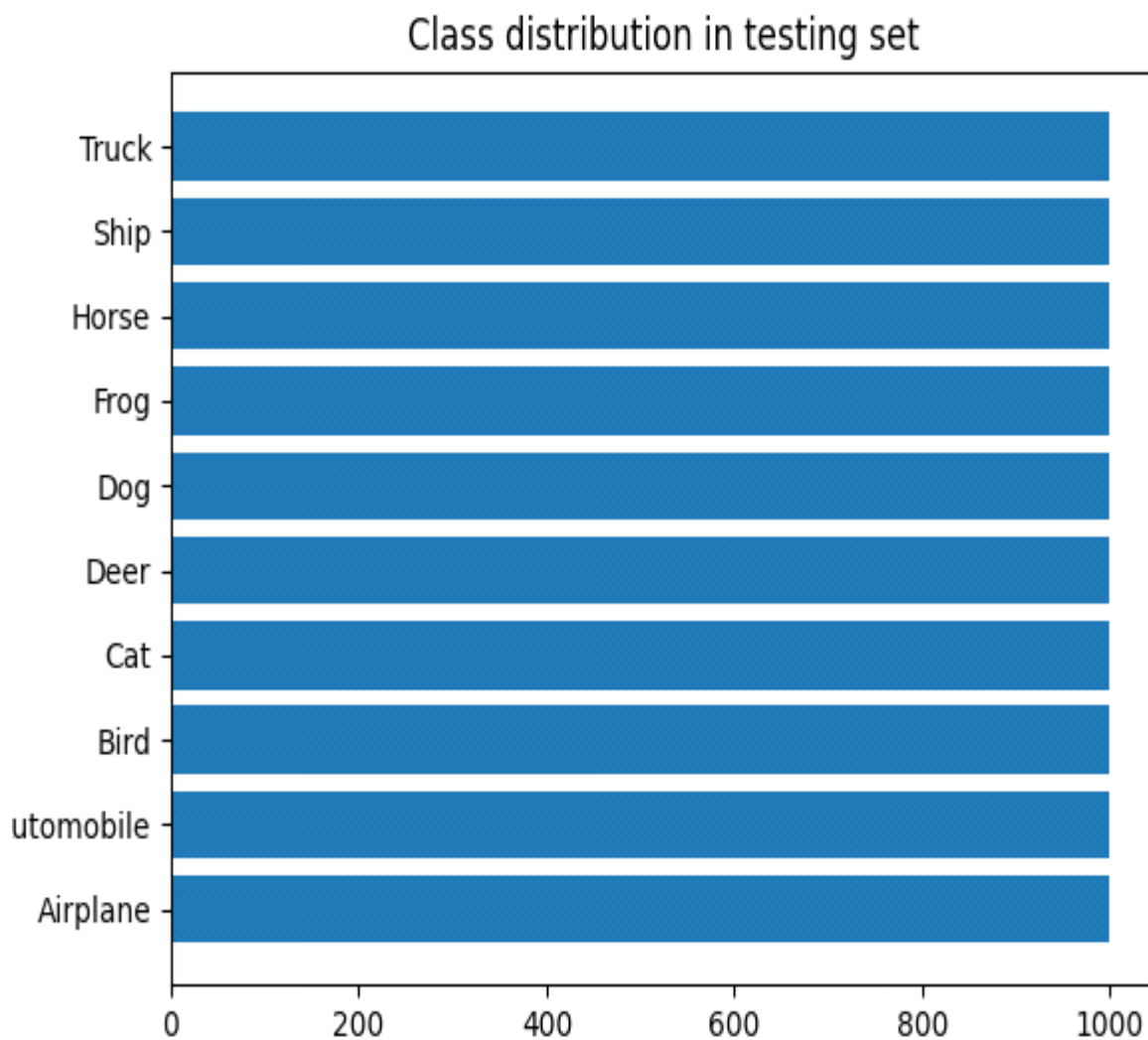


Figure IV.4 : Répartition des classes dans le jeu de test.

La Table IV.3 présente l'évolution des métriques principales (perte, précision, rappel, etc.) pour les ensembles d'entraînement et de validation sur plusieurs époques lors de l'entraînement du modèle CNN sur le jeu de données CIFAR-10.

Tableau IV .3 : Valeurs des critères d'exécution des tests

Époque	Époque 8/50	Époque 9/50	Époque 10/50	Époque 50/50
	1562/1562 - 156s 100ms/étape	1562/1562 148s 95ms/étape	1562/1562 144s 92ms/étape	1562/1562 154s 98ms/étape
Perte	0.7307	0.6964	0.6964	0.3600
Accuracy	0.7524	0.7649	0.7649	0.8764
Précision	0.8366	0.8439	0.8486	0.9083
Rappel	0.6732	0.6912	0.7023	0.8473
Perte de val	0.5864	0.5819	0.5712	0.4052
Accuracy de val	0.8050	0.8060	0.8085	0.8687
Précision de val	0.8758	0.8628	0.8674	0.8990
Rappel de val	0.7417	0.7486	0.7563	0.8432

Le modèle apprend efficacement avec une perte réduite (0,73 → 0,36) et une précision accrue (75,24 % → 87,64 %) sur l'ensemble d'entraînement.

Les performances de validation montrent une bonne généralisation (précision de 80,50 % → 86,87 %), bien qu'un léger écart suggère un début de surapprentissage, nécessitant des ajustements pour optimiser davantage.

Totalement : Les performances montrent un apprentissage efficace et une généralisation correcte.

La différence légère entre les métriques d'entraînement et de validation (écart de précision et de perte) peut signaler un début de surapprentissage.

Des ajustements, comme la régularisation ou l'augmentation des données, pourraient améliorer encore les résultats.

### IV.4.3 Accuracy et perte du modèle

La figure IV.5 suggère que le modèle apprend et acquiert plus d'informations au fil du temps. Par conséquent, la perte d'entraînement diminue à un rythme plus rapide que la perte de validation, ce qui indique une amélioration des performances sur les données d'entraînement. L'écart croissant entre les courbes d'entraînement et de validation peut indiquer un sur-apprentissage potentiel, nécessitant des techniques pour améliorer la généralisation.

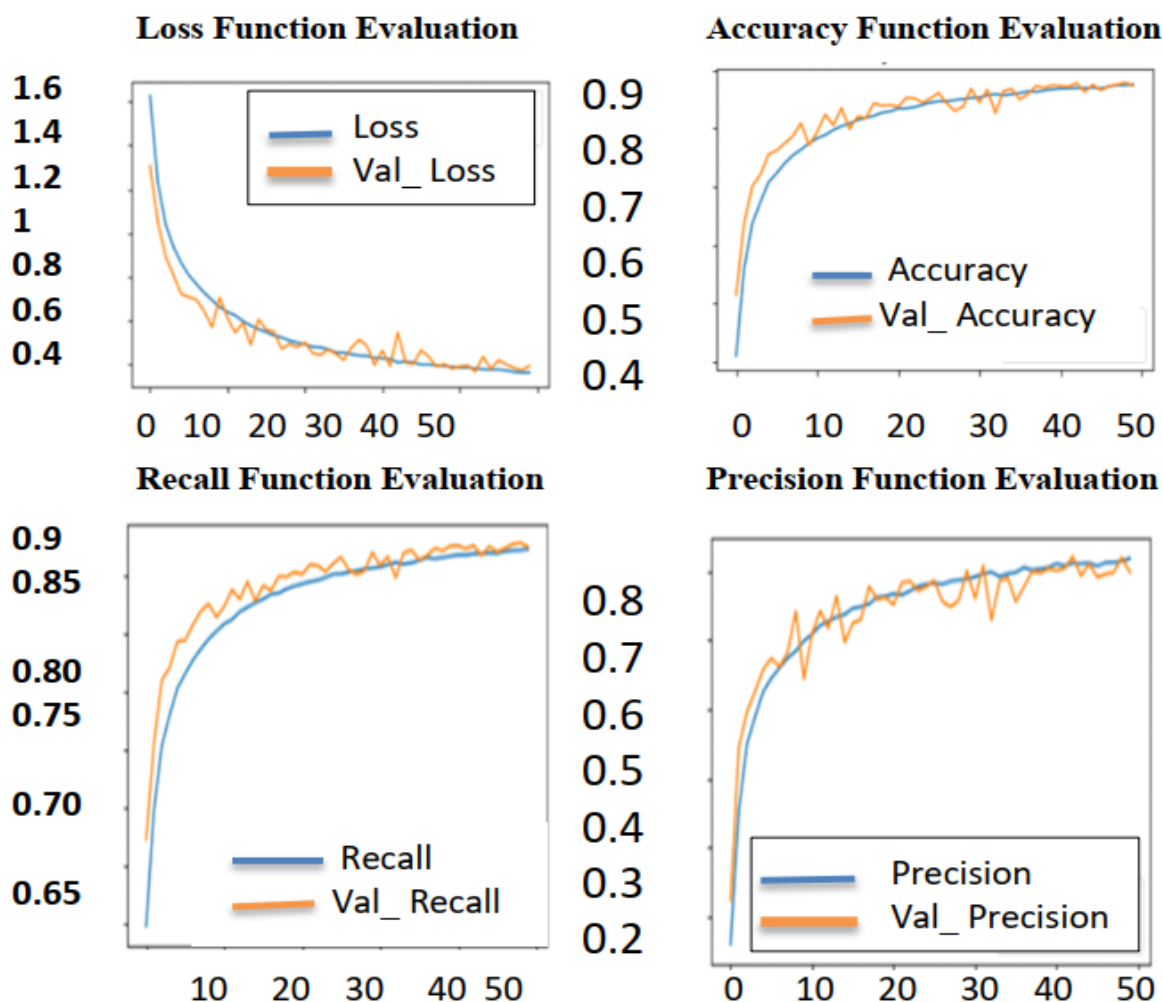


Figure IV.5 : Accuracy du test et perte du modèle

La perte d'entraînement diminue de 0,73 à 0,36, tandis que celle de validation passe de 0,58 à 0,40, indiquant une amélioration constante et une bonne convergence. Parallèlement, l'accuracy d'entraînement atteint 87,64 %, et celle de validation, 86,87 %, avec un écart minimal, ce qui reflète une excellente généralisation.

Le modèle montre une capacité d'apprentissage robuste sans surajustement notable, bien qu'une régularisation légère et une exploration des performances au-delà de 50 époques pourraient renforcer sa fiabilité.

La figure IV.6, illustre la performance du modèle dans le classement d'images à travers plusieurs catégories, telles que les oiseaux, les chats, les camions et les chiens. Le modèle montre une bonne précision pour les classes distinctes comme les oiseaux et les camions, mais certaines catégories, comme les chiens et les cerfs, présentent des erreurs de classification notables dues à des similarités visuelles.

La matrice reflète une performance solide, avec des opportunités pour affiner la précision sur les classes les plus confondues.

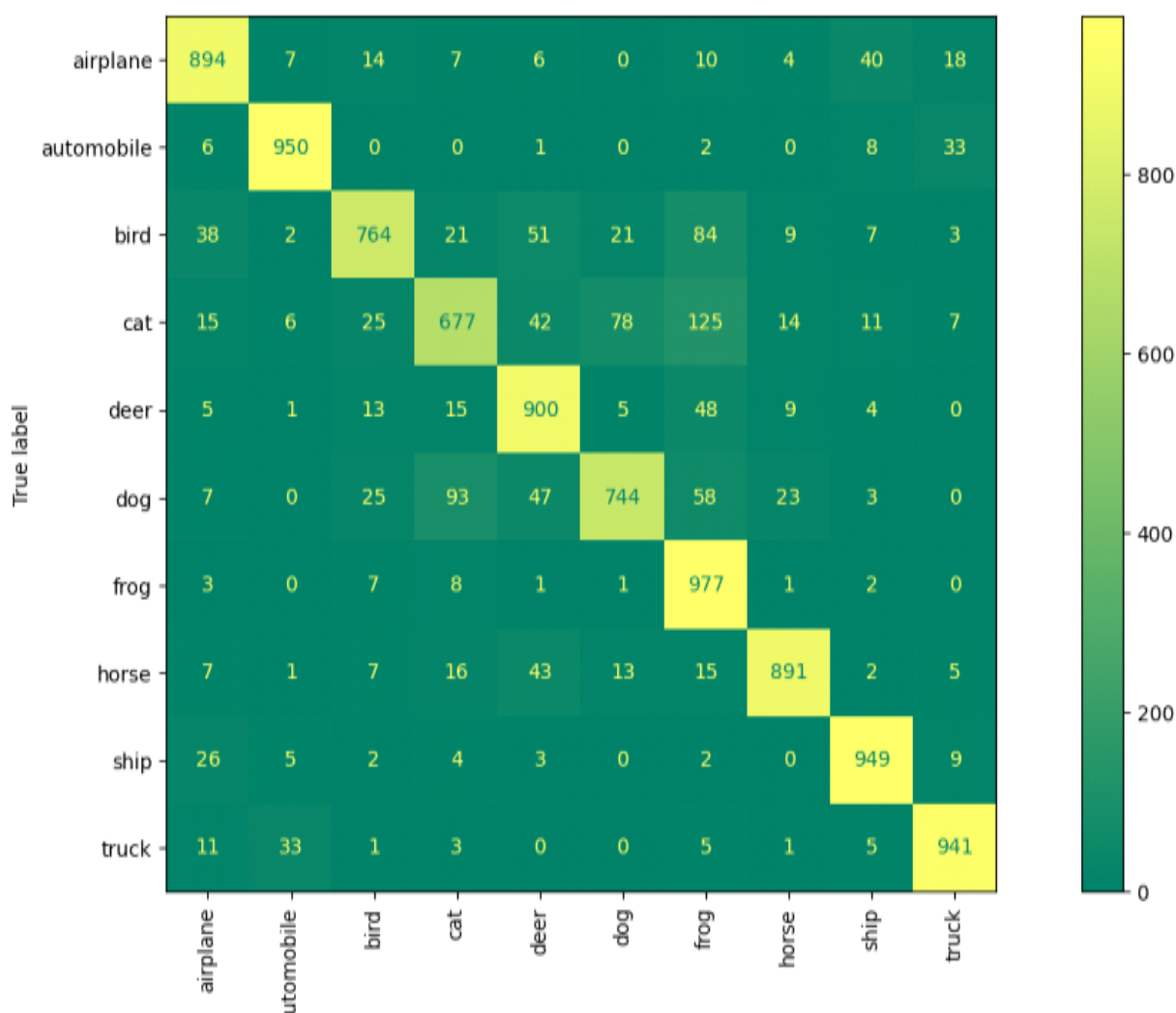


Figure IV.6 : Matrice de confusion du modèle

La matrice de confusion est un outil précieux pour évaluer la performance d'un modèle car elle offre des insights sur les vrais positifs, les vrais négatifs, les faux positifs et les faux négatifs.

### IV.4.4 Taux d'erreur

La figure IV.7 illustre le taux d'erreur d'images classées correctement et incorrectement par le modèle entraîné. À partir de cette observation, on peut en déduire que le modèle a atteint un niveau de précision satisfaisant, car le taux de mauvaise classification est inférieur au quart de l'ensemble des données.

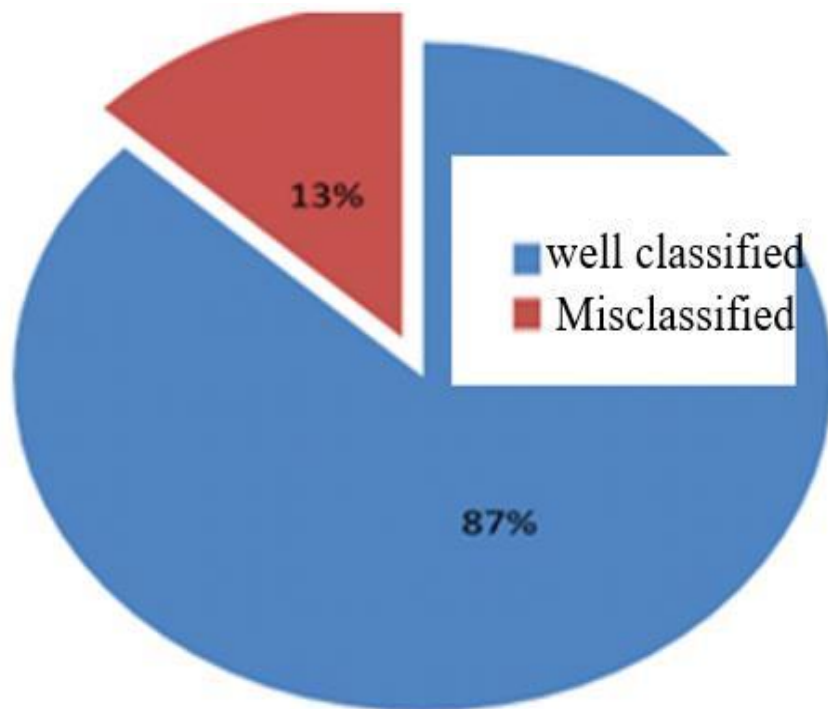


Figure IV.7 : Taux d'erreur du modèle

Cette analyse montre que le modèle a atteint une précision de test de 87,09%, indiquant une bonne capacité à classer les images avec une faible proportion d'erreurs. Le taux d'erreur est inférieur à 25% de l'ensemble des données, ce qui démontre un niveau de précision satisfaisant dans les classifications.

### IV. 5 Section de Test

Basé sur la figure IV.8 de la section "Test", l'image examinée a été efficacement reconnue et son emplacement a été correctement identifié. Bien que nous ayons appliqué un filtre pour voir s'il pouvait l'identifier, il l'a identifié comme un cerf et a obtenu un rang de 4 dans l'évaluation.

Néanmoins, il reste une opportunité d'améliorer encore les résultats et d'obtenir des performances encore meilleures.

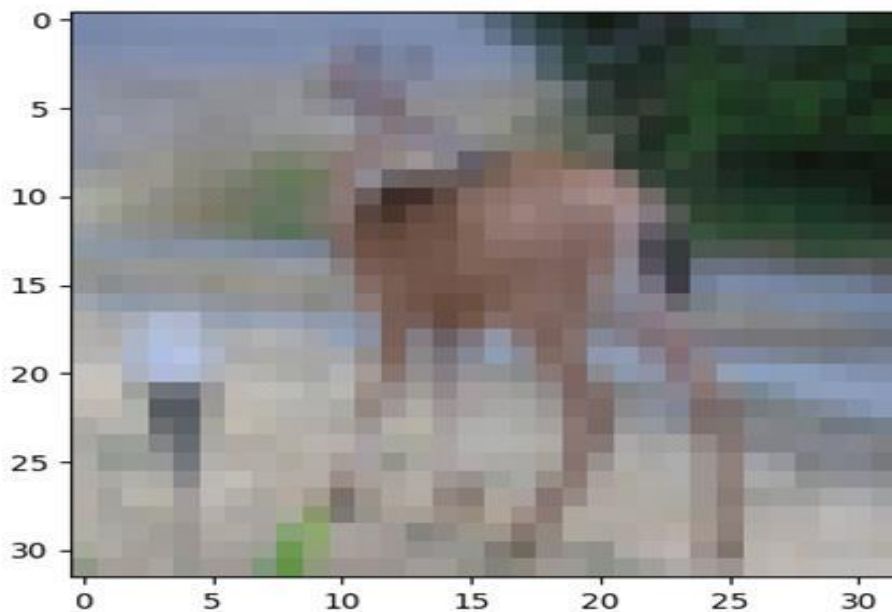


Figure IV.8 : Résultats de la section de test.

Avant de conclure les tests, le code fourni génère un ensemble d'images avec leurs étiquettes correspondantes dans une grille à des fins de visualisation.

Chaque image est sélectionnée au hasard à partir de l'ensemble `X_Test`, et l'étiquette prédite est assignée comme titre à chaque image.

La grille résultante d'images et d'étiquettes est présentée dans le document accompagnant.

La figure IV.9 montre les résultats du modèle sur d'images de test auxquelles du bruit a été ajouté. Bien que le modèle conserve une certaine capacité à classer correctement les images avec du bruit, ses performances globales diminuent par rapport aux tests effectués sur d'images sans bruit.

La précision baisse et les erreurs de classification augmentent, ce qui indique que le bruit perturbe les prédictions. Cependant, le modèle reste efficace dans de nombreux cas. Des améliorations, comme une meilleure gestion du bruit ou un prétraitement des données, pourraient aider à améliorer les résultats dans des conditions de bruit plus importantes.

La figure IV.10 présente une sélection aléatoire d'images de test, choisies à partir de l'ensemble `X_Test`. Chaque image est accompagnée de l'étiquette prédite par le modèle, et ces résultats sont organisés dans une grille pour une meilleure visualisation. L'objectif de cette section est de montrer comment le modèle classifie différentes images, mettant en évidence la diversité des données et les prédictions associées.

Ces images testées permettent de mieux évaluer la performance du modèle en termes de précision, et de visualiser comment il se comporte face à une variété de classes. À la fin des tests, le modèle a atteint une précision de test de 87,09 %, confirmant sa capacité à effectuer des classifications précises sur cet ensemble de données



Figure IV.9 : Résultats de la section de test avec du bruit.



Figure IV.10 : Sélection aléatoire d'images de test.

### IV.6 Évaluation métrique détaillée du modèle CNN-HMM sur un ensemble de données d'images

Dans la figure IV.11, l'agencement de ces métriques pour chaque classe est représenté, offrant une représentation visuelle claire de la performance du modèle en termes de classifications correctes et incorrectes.

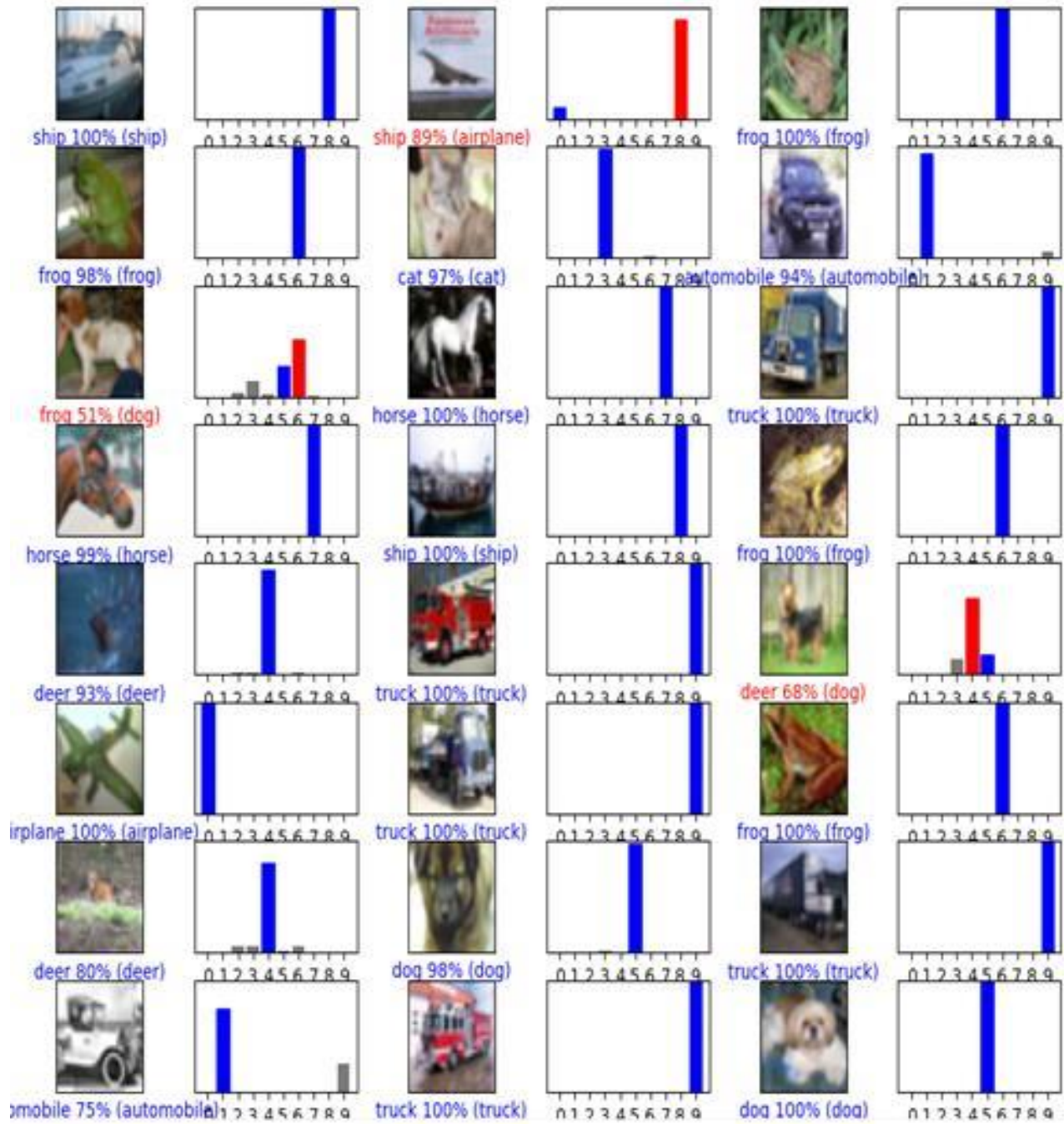
La figure IV.11 fournit un échantillon d'images, montrant les étiquettes vraies attendues et les étiquettes prédites par le modèle. Les prédictions correctes sont représentées en bleu, tandis que les prédictions incorrectes sont en rouge.

Cette figure représente d'une façon générale les résultats quantitatifs de la performance de classification de la méthode hybride proposée par rapport à d'autres méthodes de pointe.

Les résultats présentés dans de ces études suggèrent que le modèle hybride CNN-HMM peut être une approche efficace pour la classification d'images, en tirant parti des forces des CNN et des HMM.

Le CNN peut extraire des caractéristiques de l'image pertinentes pour la tâche de classification, tandis que le HMM peut modéliser les relations spatiales entre les pixels adjacents et identifier les frontières entre différentes régions de l'image.

Cependant, l'efficacité du modèle hybride dépend de divers facteurs, y compris la complexité de l'image, le nombre de classes et la qualité des données d'entraînement.



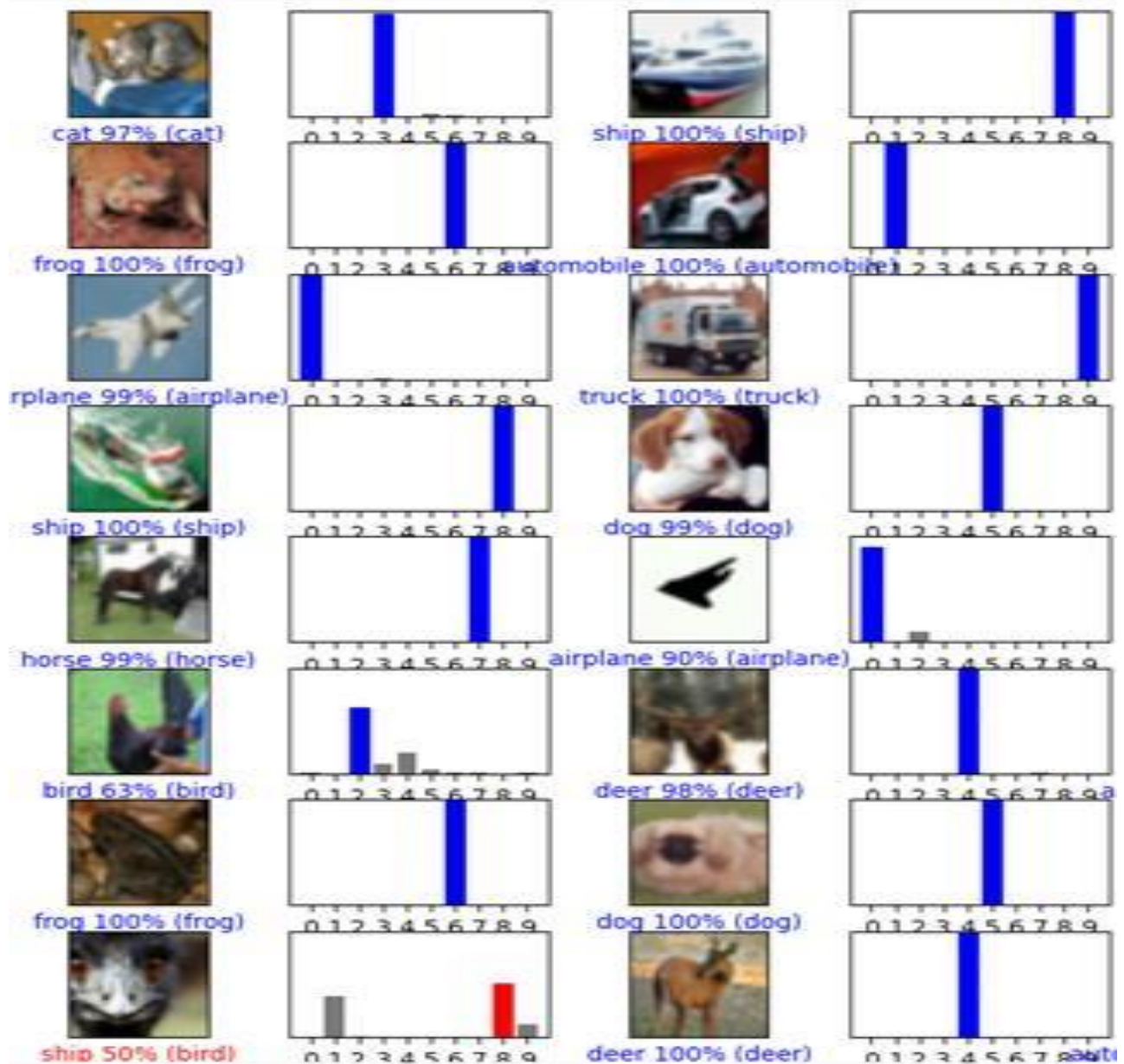


Figure IV.11 : Prédictions correctes et incorrectes mises en évidence

#### IV.7 Efficacité du modèle hybride CNN-HMM pour la classification d'images

Les résultats présentés dans ces études suggèrent que le modèle hybride CNN-HMM peut être une approche efficace pour la classification d'images, en tirant parti des forces des CNN et des HMM.

Le CNN peut extraire des caractéristiques de l'image pertinentes pour la tâche de classification, tandis que le HMM peut modéliser les relations spatiales entre les pixels adjacents et identifier les frontières entre différentes régions de l'image.

Cependant, l'efficacité du modèle hybride dépend de divers facteurs, y compris la complexité de l'image, le nombre de classes et la qualité des données d'entraînement.

Des recherches supplémentaires sont nécessaires pour évaluer la performance du modèle hybride CNN-HMM pour différents types d'images et d'applications.

### IV. 8 Etude comparative entre le modèle proposé et d'autres modèles de classification

Le tableau IV-4 illustre une comparaison détaillée entre nos résultats et d'autres études. La comparaison a été effectuée en fonction de plusieurs facteurs, notamment : précision, Accuracy, Recall, Score F1.

Tableau IV-4 : Comparaison de nos résultats avec d'autres études utilisant CIFAR-10 et divers types d'images.

Méthode	Base de données	Accuracy (%)	Recall (%)	Precision (%)	F1-score (%)
<b>Cybernetic Approaches 2023[6]</b>	CIFAR-10	82	79	81	79
<b>K-NN 2016 [7]</b>	ImageNet (extraits)	41.78	–	–	–
<b>CNN 2019 [8]</b>	CIFAR-10	85.0	84.0	85.0	84.0
<b>Travail proposé (2024)</b>	<b>CIFAR-10</b>	<b>89.09</b>	<b>84.73</b>	<b>90.83</b>	<b>87.67</b>

Le travail proposé affiche des performances nettement supérieures à celles des méthodes précédentes sur les trois métriques clés :

- **Accuracy (89.09 %) :** Le modèle proposé dépasse largement toutes les autres méthodes, la plus proche étant le CNN 2019 avec 85 %. Cela démontre une meilleure capacité du modèle à prédire correctement les classes globalement sur CIFAR-10.
- **Recall (84.73 %) :** Le recall élevé montre une excellente capacité du modèle à détecter les vraies classes, minimisant ainsi les faux négatifs. Cette valeur est comparable à celle du CNN 2019 (84 %), indiquant une bonne sensibilité maintenue.

- **Precision (90.83 %)** : Le modèle obtient la meilleure précision du tableau, ce qui signifie qu'il génère peu de faux positifs. C'est une amélioration significative par rapport aux Cybernetic Approaches (81 %) et au CNN 2019 (85 %).
- **Score F1 (87.67 %)** : Le Score F1 du modèle proposé est le plus élevé parmi toutes les méthodes comparées, reflétant un équilibre optimal entre précision et rappel, et attestant d'une performance globale robuste.

Ces résultats démontrent la supériorité et l'efficacité du modèle proposé pour la classification d'images sur CIFAR-10. Contrairement aux approches plus anciennes comme K-NN ou les méthodes cybernétiques, ce modèle exploite mieux les caractéristiques visuelles complexes, ce qui se traduit par une meilleure généralisation et une réduction significative des erreurs de classification.

### IV. 9 Conclusion

Les résultats obtenus dans cette étude indiquent que le modèle hybride CNN-HMM proposé parvient à obtenir une précision plus élevée de classification d'images, environ 87,09% sur le Jeu de Données CIFAR-10. Cela démontre la capacité du modèle à apprendre et à généraliser à de nouvelles images.

Le modèle a présenté une perte d'entraînement décroissante à un rythme plus rapide que la perte de validation, suggérant une meilleure performance sur les données d'entraînement.

**Références du chapitre IV**

- [1] C. Jiang and G. Goldsztein, “Convolutional Neural Network Approach to Classifying the CIFAR-10 Dataset,” *J. Student Res.*, vol. 12, no. 2, pp. 1–7, May 2023.
- [2] A. Djalab, L. Lalaoui, A. Bisker, and A. Hadibi, “Enhancing Image Classification Through a Hybrid Approach: Integrating Convolutional Neural Networks with Hidden Markov Mod,” *Trait. du Signal*, vol. 41, no. 1, pp. 383–390, Feb. 2024.
- [3] J. O. Healthcare Engineering, “Retracted: The Value of Python Programming in General Education and Comprehensive Quality Improvement of Medical Students Based on a Retrospective Cohort Study.,” *J. Healthc. Eng.*, vol. 2023, p. 9869424, Oct. 2023.
- [4] G. B. Nanlohy, G. G. Yosia, C. Salim, and M. Mariyanto, “PyTherNal: A python program for analyzing curie temperature from thermomagnetic data,” *J. Phys. Conf. Ser.*, vol. 2309, no. 1, p. 012035, Jul. 2022.
- [5] T. Blanke, G. Colavizza, and Z. van Hout, “An open educational resource to introduce data analysis in Python for the Humanities,” *Educ. Inf.*, vol. 39, no. 2, pp. 105–119, Jun. 2023.
- [6] S. B. Vinay and S. Balasubramanian, "A Comparative Study of Convolutional Neural Networks and Cybernetic Approaches on CIFAR-10 Dataset," *Int. J. Mach. Learn. Cybern.*, vol. 1, no. 1, pp. 1-13, 2023.
- [7] Y. Abouelnaga, O. S. Ali, H. Rady, and M. Moustafa, “CIFAR-10: KNN-based ensemble of classifiers,” *arXiv preprint arXiv:1611.04905*, 2016.
- [8] K. A. Simonyan and A. Zisserman, “Cifar-10 Classification using Deep Convolutional Neural Network,” in *2019 International Conference on Communication and Signal Processing (ICCSP)*, Melmaruvathur, India, 2019, pp. 0397–0401, doi: 10.1109/ICCSP.2019.8745428.

---

## *Conclusion générale*

---

## Conclusion Générale

Le contexte général de cette étude était la classification et la segmentation des images. Les travaux présentés dans cette thèse ont comme objectif est proposée une approche hybride CNN-HMM qui combine les réseaux de neurones convolutionnels (CNN) et les modèles de Markov cachés (HMM) pour améliorer l'efficacité de la classification d'images. Cette méthode vise à intégrer les forces des CNN pour l'extraction des caractéristiques spatiales des images avec celles des HMM pour modéliser les données séquentielles, afin d'améliorer les performances de classification sur des ensembles de données de référence.

Dans un premier temps, une définition du réseau de neurones artificiels et le deep learning a été présentée, en mettant l'accent sur les réseaux neuronaux convolutifs (CNN). Par la suite nous discutons sur le contexte historique, l'architecture et les algorithmes d'entraînement des réseaux de neurones convolutifs, ainsi que leurs diverses applications. Ensuite, Nous avons expliqué leurs structures et composants distincts, en soulignant leur efficacité en matière d'analyse et de classification d'images et en illustrant leur utilisation généralisée dans différents domaines de l'intelligence artificielle.

Nous avons donné en deuxième temps un aperçu sur modèles de Markov Cachée utilisés en classification des images, commencé par les modèles classiques, tel que les champs, les chaînes, les arbres, Ensuite, les modèles de Markov plus récents et avancés, notamment les modèles couples, les modèles triplets et les modèles flous.

Par la suite, nous avons proposé un modèle hybride CNN-HMM qui combine les forces des CNN et des HMM pour améliorer la classification d'images. Cette approche permet d'exploiter à la fois l'extraction spatiale efficace réalisée par les CNN ainsi que la modélisation temporelle fournie par les HMM.

Nous avons décrit en détail l'architecture du modèle hybride CNN-HMM, ainsi que son algorithme associé. Avons montré comment combiner ces deux techniques complémentaires pour obtenir une meilleure performance en termes de précision et d'efficacité lorsqu'il s'agit de classer ou segmenter des images.

Le modèle hybride HMM-CNN proposé tire parti des forces des deux modèles pour améliorer la précision et les performances des tâches d'analyse d'images. Nous fournissons une description détaillée de l'architecture du modèle hybride et démontrons son efficacité par des expériences sur des ensembles de données d'images de référence.

Nous avons présenté nos résultats détaillés ainsi qu'une discussion approfondie sur ces résultats. Nous avons évalué notre modèle hybride sur plusieurs jeux de données standard utilisés dans le domaine de la vision par ordinateur.

Nos résultats montrent que le modèle hybride HMM-CNN surpasse les modèles CNN traditionnels en termes de précision et d'efficacité, ce qui en fait une approche prometteuse pour les tâches d'analyse d'images.

En résumé, l'intégration des CNN et des HMM dans une approche hybride offre une nouvelle perspective pour la classification des images, en combinant les avantages des deux méthodes pour une performance améliorée.

La thèse fournit une analyse approfondie des performances du modèle hybride HMM-CNN pour la classification d'images, en mettant en lumière les avantages de combiner les capacités d'extraction de caractéristiques des CNN avec la modélisation séquentielle des HMM.