

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique



Université M'Hamed Bougara de Boumerdès
Faculté des Hydrocarbures et de la Chimie



Département Transport et Equipements des Hydrocarbures

Mémoire de fin d'études

En vue de l'obtention du diplôme de Master en Mécanique des chantiers pétroliers.

Présenté par :

- BOULAGHLEME Abdenour
- REZAIMIA Oussama

Application de l'intelligence artificielle au contrôle de la qualité des surfaces métalliques

En présence de :

Mr. BENBRIK Abderrahmane	Président, FHC
Mr. SAHNOUN Khaled	Examineur, FHC
Mr. HARHOUT Riad	Examineur, FHC
Mr. SABRI Khier	Encadreur, FHC

Boumerdès 2024.

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Remerciement

Tout d'abord, merci à Dieu, par la grâce duquel toutes les bonnes actions sont accomplies.

Ensuite, nous remercions nos chers parents pour leurs sacrifices, leur assistance morale et matérielle et leur soutien matériel, ainsi que nos chers frères et tous nos amis.

Nous tenons à exprimer nos sincères remerciements à tous les professeurs qui nous ont aidés tout au long de nos études universitaires, en particulier notre encadreur Mr. SABRI Khier qui nous a aidés tout au long de la rédaction de notre thèse, nous tenons à le remercier pour ses conseils continus, sa patience, ses précieux conseils et ses encouragements, et nous saluons ses connaissances et son savoir.

Sans oublier nos camarades de promotion du MACP19, nous avons passé avec eux cinq belles années qui se sont écoulées comme des heures.

Enfin, nous tenons à remercier toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce travail.

ملخص

في مجال التصنيع، تعتبر جودة اسطح القطع المراد إنتاجها ذات أهمية كبيرة ومن الضروري التحكم فيها أثناء عملية التصنيع. على الرغم من التحديات التي يفرضها الفحص البصري، خاصةً مع زيادة سرعة الإنتاج، إلا أنه من الصعب تحديد العيوب في القطع حتى بعد تصنيعها، مما يؤدي إلى انخفاض الجودة وزيادة النفقات. يشتمل العمل المقدم في هذه الأطروحة على تطبيق خوارزميات الذكاء الاصطناعي (AI) لتحديد العيوب السطحية في الصفائح المعدنية وتحديد مواقعها تلقائيًا وتحسين نتائج التصنيف. لتحقيق ذلك، يتم تدريب الخوارزميات على قاعدة بيانات لصور العيوب المختلفة التي يمكن أن تظهر على الصفائح المعدنية. ويسهل هذا النهج تطوير أداة حقيقية لمراقبة جودة الصفائح المعدنية وتحسين الصيانة والتحكم في العمليات.

الكلمات المفتاحية: صفائح معدنية، جودة السطح، عيوب السطح، الذكاء الاصطناعي.

Résumé

Dans l'industrie de fabrication, la qualité de la surface des pièces à produire est d'une grande importance et il est essentiel de la contrôler au cours des processus de production. Malgré les défis posés par l'inspection visuelle, notamment avec l'augmentation de la vitesse de production, il est difficile d'identifier les défauts dans les produits même après leur fabrication, ce qui entraîne une réduction de la qualité et une augmentation des dépenses.

Le travail que nous avons réalisé dans ce mémoire consiste à appliquer des algorithmes d'intelligence artificielle (IA) pour identifier et repérer automatiquement les défauts de surface des tôles et pour optimiser les résultats de la classification. Pour ce faire, ces algorithmes sont entraînés sur une base de données d'images des différents défauts qui peuvent apparaître sur la tôle. Nous contribuons ainsi à fournir un véritable outil de contrôle de qualité des tôles, d'optimisation de la maintenance et du contrôle des procédés.

Mots clés : Tôles d'acier, Qualité de la surface, Défauts de surface, Intelligence artificielle.

Abstract

In the manufacturing industry, the surface quality of the parts to be produced is of great importance and it is essential to control it during the production process. Despite the challenges posed by visual inspection, particularly with the increase in production speed, it is difficult to identify defects in products even after they have been manufactured, leading to a reduction in quality and an increase in expenditure.

The work presented in this thesis comprises the application of artificial intelligence (AI) algorithms to automatically identify and locate surface defects in sheet metal and to optimize the classification results. To achieve this, the algorithms are trained on a database of images of the various defects that can appear on sheet metal. This approach facilitates the development of a genuine tool for sheet metal quality control, maintenance optimization and process control.

Keywords: Steel sheet, Surface quality, Surface defects, Artificial intelligence.

Table des matières

Introduction générale.....	1
Chapitre 1 : Procédés de fabrication des tôles d'acier	
Introduction	2
1. Procédés de laminage.....	2
1.1. Matière première des laminoirs	2
1.2. Le laminage	3
1.2.1. Laminage à chaud.....	4
1.2.2. Laminage à froid	6
1.3. Production des laminoirs	6
1.3.1. Les Produits Plats	6
1.3.2. Les produits longs	6
2. Qualité des produits laminés	7
2.1. Objectifs.....	7
2.2. Défauts de fabrication.....	8
3. Vision artificielle pour l'inspection des surfaces des produits laminés	10
3.1. Système de vision	10
3.2. Caractéristiques et conditions de travail	11
3.2.1. Nature des produits objets d'inspection	11
3.2.2. Eclairage.....	12
3.2.3. Caméra et capteurs optiques.....	12
Conclusion.....	13
Chapitre 2 : Apprentissage profond et vision artificielle	
Introduction	14
1. L'intelligence Artificielle.....	14
2. Apprentissage profond (Deep Learning).....	15
2.1. Principe de fonctionnement	16
3. Les réseaux de neurones artificiels	16
3.1. La structure d'un neurone artificiel	16
3.2. Sur-apprentissage (Overfitting)	20
3.3. Les paramètres de l'apprentissage	20
3.3.1. La régularisation.....	20
3.3.2. Techniques d'optimisation	22

3.3.3.	Les techniques d'augmentation d'image.....	23
3.3.4.	Normalisation	24
3.3.5.	Indicateurs de performance en classification	24
4.	Architecture d'un réseau de neurones artificiel	25
4.1.	Fonctionnement des réseaux de neurones artificiel.....	26
4.2.	Les différents types de réseaux de neurones artificiels	26
4.2.1.	Les réseaux de neurones à propagation avant (feed-forward).....	26
4.2.2.	Les réseaux de neurones récurrents.....	26
4.2.3.	Les réseaux de neurones convolutifs.....	27
5.	Principe de fonctionnement des réseaux de neurones convolutifs.....	27
5.1.	Les images numériques	28
5.1.1.	Les différents types de format d'image numérique	29
5.1.2.	Les caractéristiques d'images numériques	29
5.2.	Schéma conceptuel d'un réseau de neurones convolutifs	30
5.2.1.	Couche de convolution (CONV).....	31
5.2.2.	Couche Pooling	34
5.2.3.	Couche de correction Relu	35
5.2.4.	Couche entièrement connectée (Fully connected Neural Network).....	35
5.2.5.	La couche de perte (LOSS)	36
5.3.	Les différentes architectures des réseaux de neurone convolutif	36
6.	Le modèle VGG16.....	37
6.1.	Apprentissage par transfert et VGG16	38
6.2.	Les avantages de l'utilisation de l'apprentissage par transfert avec le VGG16	38
6.3.	Les inconvénients de l'utilisation de l'apprentissage par transfert avec le VGG16... ..	39
6.4.	VGG16 et ses limites	40
6.5.	Moyens de surmonter les limites de la VGG16.....	40
7.	Le modèle YOLO	41
7.1.	L'algorithme de modèle YOLO	42
	Conclusion.....	44
	Chapitre 3 : Classification et localisation des défauts de tôles	
	Introduction	45
1.	Python	45
2.	Bibliothèques utilisées	45

3. Configuration Utilisé dans l'implémentation.....	46
4. Base de données d'images	46
5. Visualiser les prédictions avec GradCAM.....	47
6. Architectures des modelés	48
6.1. Le réseau de neurones convolutifs simple (CNN).....	48
6.2. Architecture du modèle VGG16.....	50
7. Résultats obtenus et discussion	52
8. Comparaison des résultats.....	60
9. Application du modèle YOLO	61
Conclusion.....	63
Conclusion générale	64
Bibliographie	65

List des figures

FIGURE I.1. SCHEMA SIMPLIFIE DU PROCESSUS DE FABRICATION DE L'ACIER.....	3
FIGURE I.2. TRANSFORMATION MECANIQUE EN LAMINAGE.....	4
FIGURE I.3. INSTALLATION DE LAMINAGE A CHAUD (800-1200°C).	5
FIGURE I.4. PRODUITS LAMINES.	7
FIGURE I.5. SCHEMA SIMPLIFIE D'UN SYSTEME DE VISION.....	11
FIGURE I.6. PRINCIPE DE LA VISUALISATION DE TOUTE LA LARGEUR DES DEUX FACES PAR CAMERAS HAUTE RESOLUTION EN LAMINAGE DE PRODUITS PLATS.....	11
FIGURE I.7. CONFIGURATIONS COURAMMENT UTILISEES POUR L'ECLAIRAGE EN SYSTEME DE VISION INDUSTRIELLE.....	12
FIGURE I.8. CAMERAS D'INSPECTION.	12
FIGURE II.1. LA RELATION ENTRE IA, ML ET LE DL.....	15
FIGURE II.2. STRUCTURE D'UN NEURONE ARTIFICIEL.	17
FIGURE II.3. FONCTIONS D'ACTIVATION.	18
FIGURE II.4. RESEAU DE NEURONES AVEC DROPOUT (A DROITE) ET SANS DROPOUT (A GAUCHE).	21
FIGURE II.5. L'ARRET PRECOCE (EARLY STOPPING)	22
FIGURE II.6. LE GRADIENT DESCENDANT EN BATCH ET LE GRADIENT DESCENDANT EN MINI- BATCH ET DESCENTE DE GRADIENT STOCHASTIQUE.....	22
FIGURE II.7. ARCHITECTURE D'UN RESEAU DE NEURONES.....	25
FIGURE II.8. LES RESEAUX DE NEURONES A PROPAGATION AVANT.	26
FIGURE II.9. (À GAUCHE) UN RNN (A DROITE) SA VERSION DEROULE SOURCE.....	27
FIGURE II.10. À GAUCHE : UNE CONVOLUTION, OU G EST DECALE ET INVERSE ; A DROITE : UNE ILLUSTRATION ETAPE PAR ETAPE D'UNE OPERATION DE CONVOLUTION.....	28
FIGURE II.11. LES DIMENSIONS D'IMAGES.	29
FIGURE II.12. ARCHITECTURE GENERALE D'UN RESEAU DE NEURONES CONVOLUTIF (CNN/CONVNET).	31
FIGURE II.13. COUCHE POOLING.....	34
FIGURE II.14. FONCTION D'ACTIVATION RELU	35
FIGURE II.15. UNE COUCHE FULLY-CONNECTED.	36
FIGURE II.16. L'ARCHITECTURE DE VGG16	38
FIGURE II.17. L'ARCHITECTURE DE VGG16.....	39
FIGURE II.18. ARCHITECTURE DE MODEL YOLO.....	41
FIGURE II.19. DIVISER L'IMAGE.....	42

FIGURE II.20. LE VECTEUR PREDIT DANS LE CAS D'UNE SEULE BOITE.....	43
FIGURE II.21. LE RESULTAT DE SUPPRESSION NON MAXIMALE.....	43
FIGURE II.22. SORTIE DE DETECTION.	44
FIGURE III.1. CARACTERISTIQUES TECHNIQUES DU MATERIEL UTILISE DANS LES CALCULS.....	46
FIGURE III.2. DEFAUTS DE SURFACE TYPQUES DES TOLES D'ACIER.	47
FIGURE III.3. ARCHITECTURE DU MODELE 1 (CNN SIMPLE).	49
FIGURE III.4. ARCHITECTURE DU MODELE VGG16.	51
FIGURE III.5. COURBES DE PRECISION ET DE PERTE ASSOCIEES AU MODELE 01.....	52
FIGURE III.6. TAUX D'ERREUR DU MODELE 01.	53
FIGURE III.7. MATRICE DE CONFUSION POUR LE MODELE 01.	57
FIGURE III.8. IMAGES PREDITES ET LEURS LIBELLES.	54
FIGURE III.9. VISUALISATION GRADCAM DU MODELE 1.	55
FIGURE III.10. PRECISION ET PERTE DU MODELE 02.....	56
FIGURE III.11. TAUX D'ERREUR DU MODELE 02.	56
FIGURE III.12. MATRICE DE CONFUSION POUR LE MODELE 02.	57
FIGURE III.13. EXTRAIT DES IMAGES PREDITES AVEC LEURS LABELS RESPECTIFS DU MODELE 2.	58
FIGURE III.14. ILLUSTRATION GRADCAM ASSOCIEE AU MODELE 2.	59
FIGURE III.15. RESULTATS DE DETECTION PAR YOLOV9.....	61
FIGURE III.16. MATRICE DE CONFUSION DU MODELE YOLO.....	62

List des tableaux

TABLEAU III.1. CONFIGURATION DU MODELE 1.	49
TABLEAU III.2. CONFIGURATION DU MODELE 2 (LE VGG16).	51
TABLEAU III.3. RESULTATS COMPARES ENTRE MODELE 01 ET MODELE 02.	60

Acronymes

IA : Intelligence artificielle.

ML : Machine learning (l'apprentissage automatique).

DL : Deep learning (Apprentissage profond).

Tanh : Tangente hyperbolique.

ReLU : Rectified Linear Unit.

PReLU : Paramétrique Rectified Linear Unit.

SGD : Stochastique Gradient Descente.

MB-SGD : Mini batch- Stochastique Gradient Descente.

RMSProp : RootMean Square Propagation (propagation quadratique moyenne).

Adam : Adaptative du moment.

CNN : Convolutional neural network (réseaux de neurones convolutifs).

RNN : Recurrent Neural Network (Les réseaux de neurones récurrents).

RVB: rouge, vert, bleu.

VGG: Visual Geometry Group.

YOLO: You Only Look Once.

CPU: Central Processing Unit.

GPU: Graphics Processing Unit.

RAM: Random Access Memory.

OS: Operating System.

GradCAM: Gradient-weighted Class Activation Mapping.

Introduction générale

Le secteur de la fabrication mécanique joue un rôle central dans l'économie mondiale, en fournissant des produits essentiels à une multitude de secteurs, notamment l'automobile, l'aérospatiale et la construction. La qualité des tôles d'acier est d'une importance capitale pour la sécurité et la performance des produits finis. Les défauts de surface, tels que les rayures, les bosses, les fissures et les inclusions, peuvent altérer de manière significative les propriétés mécaniques de la tôle et affecter son intégrité structurelle.

L'approche conventionnelle de l'inspection des tôles consiste à utiliser des méthodes manuelles ou semi-automatiques, souvent basées sur l'inspection visuelle par des opérateurs qualifiés. Bien que ces méthodes soient fiables, elles présentent des limites en termes de rapidité, de précision et de répétabilité. En outre, elles peuvent être fastidieuses et sujettes à l'erreur humaine, ce qui peut entraîner des coûts importants liés aux rejets et aux défaillances potentielles. En matière de surveillance et de diagnostic des équipements, l'intelligence artificielle (IA) est apparue comme une solution prometteuse pour relever les défis susmentionnés.

L'objectif de ce mémoire est de présenter l'implémentation d'un modèle basé sur l'IA pour le contrôle de la surface des tôles en acier laminées.

Avec l'avènement des réseaux de neurones convolutifs (CNN), le domaine du traitement d'images s'est transformé, ouvrant de nouvelles voies pour le contrôle industriel des surfaces de tôles. Dans ce contexte, notre objectif principal est d'identifier et de classifier de multiples défauts sur des surfaces de tôles en acier. Cette approche utilise des techniques d'apprentissage profond pour développer un système de reconnaissance et d'identification des défauts. Cette technologie présente de nombreux avantages. Elle permet de réduire le temps de détection des pannes, d'améliorer la sécurité, de réduire les coûts, d'améliorer la qualité des pièces et d'optimiser les processus de maintenance.

Ce mémoire est divisé en trois (03) chapitres :

Au premier chapitre, nous présentons les procédés de fabrication des tôles d'acier et les défauts qui peuvent en résulter. Les théories de l'apprentissage profond et les techniques du traitement de l'image font l'objet du deuxième chapitre. Dans le troisième chapitre, les algorithmes d'apprentissage profond tels que le modèle CNN simple, le VGG16, la visualisation GradCAM, ainsi que le modèle YOLOv9 sont appliqués et permettent de classifier et de repérer les défauts présents sur la surface des tôles d'acier.

CHAPITER 01

Procédés de fabrication des tôles d'acier

Introduction

Dans ce chapitre, nous présentons les divers aspects des opérations de laminage, depuis le processus de traitement initial des matières premières jusqu'à la qualité des produits finis, en mettant l'accent sur les avancées technologiques telles que la vision industrielle pour l'inspection des surfaces. Nous entamons en introduisant les différentes étapes du laminage. Ensuite, nous abordons la diversité des plats et des longs produits par les laminoirs. Nous discutons aussi les imperfections de fabrication et les techniques employées pour les repérer. Dans un second temps, nous expliquons l'influence de la vision par ordinateur sur l'inspection de la surface des produits laminés, en soulignant les systèmes spécifiques employés pour cette opération. Par ailleurs, nous abordons les spécificités et les conditions de fonctionnement des systèmes d'inspection par vision industrielle, en mettant en évidence les éléments essentiels tels que l'éclairage, les caméras et les capteurs optiques.

Ce chapitre présente donc une vision globale des processus de laminage, de la fabrication à la qualité des produits finis, en soulignant les avancées technologiques qui influencent ce secteur industriel majeur.

1. Procédés de laminage

1.1. Matière première des laminoirs

Le processus d'affinage de la fonte vise à diminuer la quantité de carbone et à éliminer les impuretés. Il facilite l'acquisition de produits provenant des aciéries qui, au cours de la chaîne de production, sont transportés vers les laminoirs. Le schéma I.1 présente une vue d'ensemble des principales étapes du processus sidérurgique. L'acier liquide, qui est continuellement coulé à une température de 1500°C, commence à se former dans une lingotière qui agit comme un échangeur thermique dont les parois sont refroidies à l'eau afin de réduire la température superficielle. Grâce à ce refroidissement, une enveloppe d'acier solidifiée se forme, ce qui lui confère une résistance adéquate pour empêcher le produit moulé et encore fragile de se briser à sa sortie de la lingotière. La lingotière utilisée est spécifiquement conçue pour le type de produit à fabriquer (Billette ou Brame). On obtient ces produits à l'aide d'un ensemble de systèmes automatisés qui les extraient de la lingotière, comme les rouleaux motorisés, un second système de refroidissement pour finir leur solidification graduelle, et l'oxycoupage qui procède à la découpe de tronçons qui sont ensuite stockés en tant que produits semi-finis. Par la suite, les procédés de laminage sont utilisés pour transformer ces

Chapitre 01 : Procédés de fabrication des tôles d'acier.

brames/billetés, ce qui donne lieu à des bandes de tôle noires et assez épaisses, voire moins épaisses et revêtues.

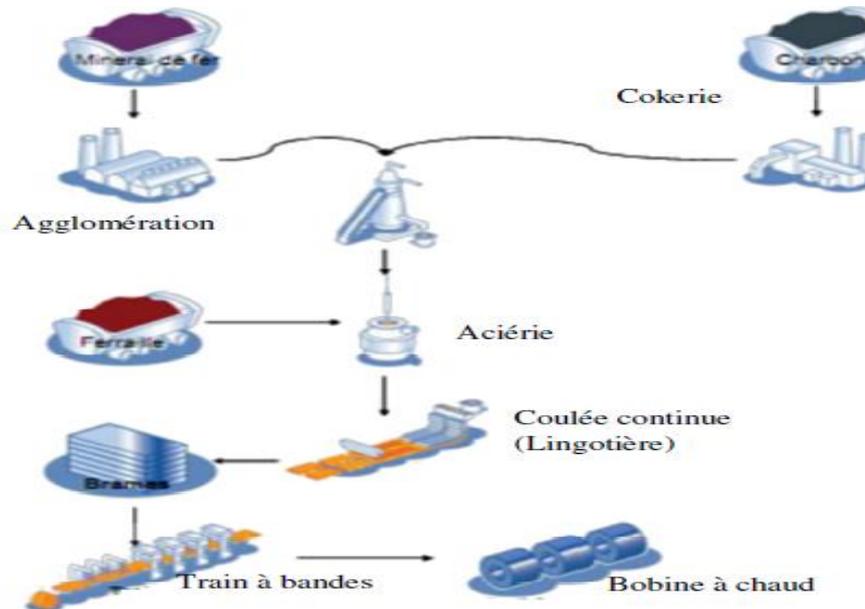


Figure I.1. Schéma simplifié du processus de fabrication de l'acier.

1.2. Le laminage

Le laminage vise principalement à déformer le matériau en utilisant une compression exercée par des cylindres tournant en sens inverse, comme illustré dans la Figure I.2. Le produit brut de l'aciérie, appelé brame brute, est considéré comme un produit intermédiaire car sa composition chimique est bien définie, mais il demeure indifférencié par rapport au produit final en termes de géométrie, de structure interne, de caractéristiques mécaniques et d'état de surface. Au fil des traitements suivants, tels que le laminage à chaud, le laminage à froid et le parachèvement, ces caractéristiques lui sont attribuées.

Ainsi, la transformation du demi-produit en produit fini est généralement réalisée selon deux processus :

- Le laminage à chaud ou d'ébauche "hot rolling" utilisant les produits de l'aciérie.
- Le laminage à froid ou de finition "cold rolling" utilisant les produits du laminage à chaud.

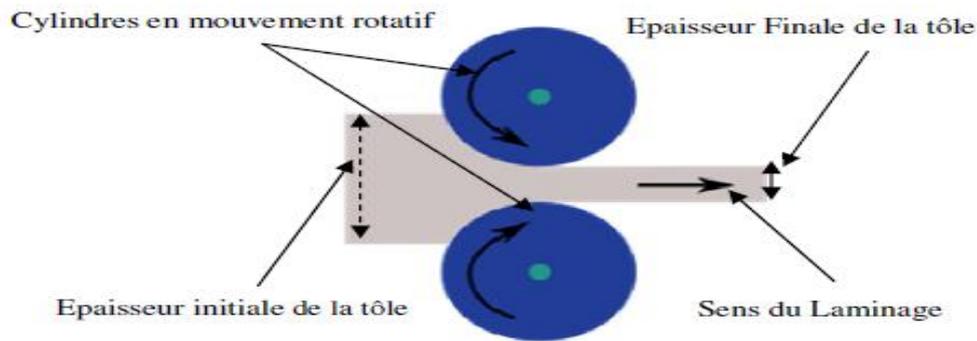


Figure I.2. Transformation mécanique en laminage.

Le premier est spécialement conçu pour travailler sur des pièces de grande taille. Les efforts déployés sont excessivement importants et exigent donc un produit relativement flexible en entrée. On parvient à cet état en augmentant sa ductilité par le chauffage préalable.

Sur le plan métallurgique, le laminage à chaud entraîne une modification de la structure cristalline primaire du lingot, un affinage du grain, une homogénéisation de la composition et une amélioration des caractéristiques, en particulier dans le sens longitudinal, en raison de l'orientation des structures cristallines.

Le produit qui sort du laminage à chaud est généralement d'une finition standard et d'un noir.

Pour certains produits, comme les tôles fortes, les bandes à chaud et d'autres produits longs, il suffit d'effectuer un laminage à chaud, suivi d'opérations de finition (traitements thermiques, dressage, décapage, revêtements et usinages). En ce qui concerne le passage au laminage à froid, il est généralement utilisé pour produire des tôles minces, c'est-à-dire là où des tolérances serrées (quelques micromètres), des propriétés mécaniques élevées et un état de surface satisfaisant sont recherchés.

La différence entre le laminage à chaud et à froid réside dans le fait que le premier entraîne un laminage plus épais, tandis que le second entraîne un laminage plus mince. Cette distinction engendre une autre question relative aux zones de température, qui sont habituellement compris entre 800 et 1200 °C lors du laminage à chaud et inférieur à 200 °C lors du laminage à froid ^[1].

1.2.1. Laminage à chaud

Le laminage à chaud, qui intervient immédiatement après la coulée continue, est une étape essentielle du processus sidérurgique. La transformation thermomécanique implique la combinaison d'un traitement thermique qui permet de réchauffer la brame jusqu'à la température nécessaire pour augmenter la ductilité de l'alliage et prévenir les cassures, ainsi

Chapitre 01 : Procédés de fabrication des tôles d'acier.

que du laminage, qui consiste à déformer le métal en réduisant son épaisseur puis à le refroidir jusqu'à la température ambiante par refroidissement [2].

Pour obtenir un laminé de l'épaisseur souhaitée, on établit un schéma de laminage qui détermine le nombre de passes et le niveau de réduction d'épaisseur par passe, en prenant en compte plusieurs paramètres électriques (couple, force, vitesse, etc.).

Pour réaliser ce schéma, il est essentiel d'avoir un Laminoir (comme le montre la Figure I.3) ou un train de laminage, ainsi que des installations auxiliaires, généralement constituées de :

- Parc à demi-produits avec installation de nettoyage.
- Four de réchauffage à plusieurs zones.
- Une cage Quarto dégrossisseuse avec un moteur d'entraînement par cylindre de travail.
- Un train finisseur composé de quatre à neuf cages avec un moteur d'entraînement par cage.
- Table de refroidissement accéléré.
- Equipement de pilotage et de contrôle.
- Convoyeur de bobines, équipement de cerclage, pesage.
- Equipement de cisailage et d'oxycoupage.

On distingue au niveau des cages de laminage deux types de cylindres :

- Les cylindres de travail : Dont la fonction est la réduction de l'épaisseur du métal qui s'engage par frottement entre deux ou plusieurs cylindres tournant autour de leurs axes.
- Les cylindres de soutien : De diamètre plus important, ils servent d'appui pour éviter le cambrage des cylindres de travail. On compte sur installation un cylindre de soutien pour une paire de cylindres de travail.

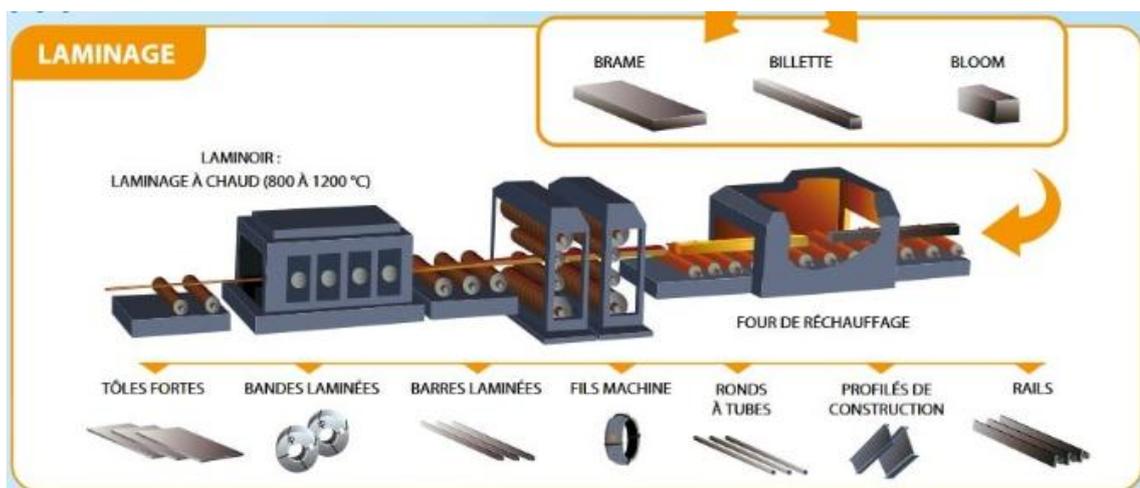


Figure I.3. Installation de laminage à chaud (800-1200°C).

1.2.2. Laminage à froid

Le laminage à froid est un processus essentiel pour la fabrication du produit fini, car il est impossible d'obtenir les caractéristiques dimensionnelles, structurales et esthétiques uniquement grâce au laminage à chaud, car la bande se refroidit rapidement et s'oxyde sous l'effet de la température. Le laminage à froid, en utilisant les différentes nuances d'aciers provenant du procédé à chaud, offre une grande précision :

- L'atteinte des caractéristiques dimensionnelles (épaisseur finale d'utilisation, largeur, planéité et tolérances).
- L'obtention de l'homogénéité de la structure et la fixation des caractéristiques mécaniques (résistance, ...).
- L'atteinte de la qualité de surface requise (aptitude aux traitements de surface, brillance, rugosité).

1.3. Production des laminoirs

Selon le procédé de laminage appliqué et le semi-produit en entrée, on peut obtenir ce qui est communément appelé produit plat ou produit long et cela repose essentiellement sur leurs caractéristiques géométriques obtenues, comme montré sur la Figure I.4 ^[3].

1.3.1. Les Produits Plats

L'épaisseur est petite devant la largeur, elle-même très inférieure à la longueur et on distingue:

- Plaques ou tôles fortes : Produits de grande largeur et d'épaisseur supérieure à 8-12 mm.
- Tôles, dont l'épaisseur est comprise entre 3 et 12 mm ;
- Tôles minces, au-dessous de 3 mm Produits du laminage à chaud ou à froid.

1.3.2. Les produits longs

Les dimensions de l'épaisseur et la largeur sont du même ordre de grandeur, en général, et sont faibles comparés à la longueur.

- Barres (Barres laminées, Barres forgées, Barres creuses) ;
- Profilés, Profilés spéciaux, Profilés soudés, Poutrelles ;
- Produits crénelés ou nervurés pour béton armé ;
- Cornières, Tés, Palplanches, Tubes sans soudure, Tubes soudés, Profils creux ;
- Matériel de voies ferrées, Fil machine.

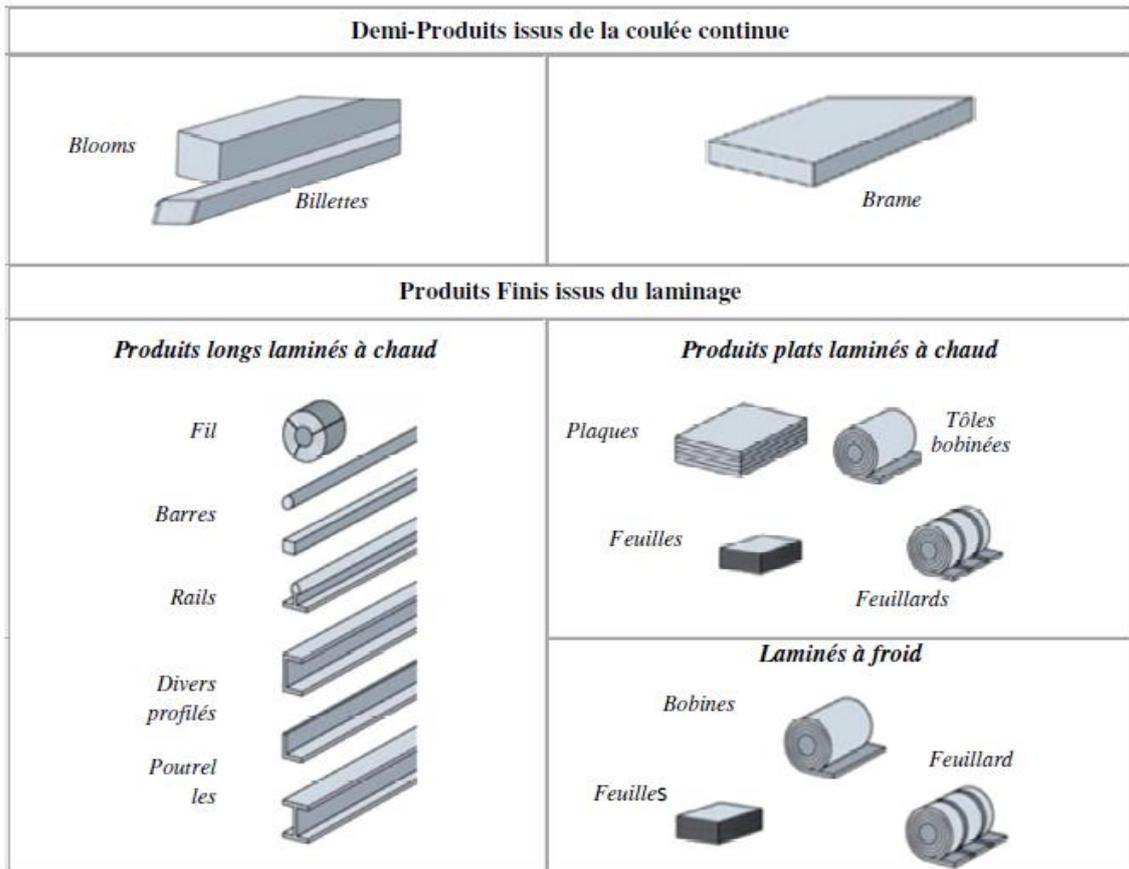


Figure I.4. Produits laminés.

2. Qualité des produits laminés

2.1. Objectifs

L'usage très diversifié de l'acier dans l'industrialisation massive des produits de consommation, dans la construction, le transport et pour les engins de manutention, pour ne citer que ces domaines, conjugués aux exigences des consommateurs ont largement impulsé à la fois le développement des processus de fabrication de l'acier et l'amélioration de sa qualité en tout point de vue.

Ainsi, pour les propriétés recherchées, nous trouvons des défis lancés par cette industrie et ayant connu du succès, par rapport à :

1. La qualité géométrique (tolérances d'épaisseur, largeur, planéité à la découpe, rectangulaire du profil, ...) dépendant directement des systèmes automatisés de mesures, de formage et de découpe.
2. L'aspect de surface correspondant au soucis fonctionnel et esthétique exigés par le consommateur, et pour lesquels des objectifs sont préalablement fixés tels que le nombre (minimum) de défauts de surface permis par bobine laminée. Ces défauts doivent d'ores et

déjà être évités à chaque étape de la chaîne de fabrication : coulée continue, train à chaud, à froid, recuit et revêtement.

3. La santé interne, notamment pour les produits les plus minces destinés aux biens de consommation où un seuil des plus bas en matière d'inclusions conditionne l'acceptation du produit. L'étape critique, à ce niveau, est celle de la coulée continue de l'acier qui présente des risques de piégeage d'éventuelles impuretés dans l'acier liquide.
4. La pureté analytique par rapport à la nuance souhaitée, qui définit les caractéristiques mécaniques de l'acier laminé.

Néanmoins, malgré les avancées technologiques enregistrées et la maîtrise des processus, les objectifs d'amélioration de la qualité et d'optimisation restent des sujets majeurs d'études et de progrès, car bons nombres de défauts demeurent inévitables bien que minimisés. Bien des produits se trouvent déclassés ou même rebutés pour des raisons de non-conformité au cahier des charges et occasionnent des pertes et autres perturbations au niveau des chaînes de production.

2.2 Défauts de fabrication

Le nombre conséquent d'opérations effectués sur le produit en acier, allant de la phase de fusion jusqu'à l'obtention du produit fini en passant par la solidification, les refroidissements, les déformations, les traitement thermiques et la manutention, peuvent être à l'origine d'une multitude de défauts qui peuvent être aussi bien internes qu'externes pour les premières étapes et plutôt superficiels vers la fin du processus.

Parmi les défauts les plus importants :

Poinçonnage (Punching) : Dans la ligne de production de la bande, la bande d'acier doit être poinçonnée selon les spécifications du produit ; une défaillance mécanique peut entraîner un poinçonnage non désiré, entraînant des défauts de poinçonnage.

Ligne de soudure (Welding line) : Lorsque la bande est changée, il est nécessaire de souder les deux bobines de la bande, ce qui produit une ligne de soudure. À proprement parler, ce n'est pas un défaut, mais il doit être détecté et suivi automatiquement pour être contourné dans les coupes ultérieures.

Écart en croissant (Crescent gap) : Dans la production de bande d'acier, la découpe entraîne parfois des défauts, comme une demi-circulaire.

Tache d'eau (Water spot) : Une tache d'eau est produite par séchage en production. Sous différents produits et processus, les exigences pour ce défaut sont différentes. Cependant,

Chapitre 01 : Procédés de fabrication des tôles d'acier.

parce que les taches d'eau sont généralement de faible contraste, et sont similaires à d'autres défauts tels que les taches d'huile, elles sont généralement détectées par erreur.

Tache d'huile (Oil spot) : Une tache d'huile est généralement causée par la contamination du lubrifiant mécanique, ce qui affectera l'apparence du produit.

Tache de soie (Silk spot) : Une plaque locale ou continue en forme d'onde sur une surface de bande qui peut apparaître sur les surfaces supérieure et inférieure, et la densité est inégale dans le sens de la longueur de la bande entière. Généralement, la principale raison réside dans la température inégale du rouleau et la pression inégale.

Inclusion (Inclusion) : L'inclusion est un défaut typique des défauts de surface métallique, se manifestant généralement par de petits points, une forme d'écaille de poisson, une forme de bande, une distribution irrégulière de blocs dans la bande de la surface supérieure et inférieure (globale ou locale), et est souvent accompagnée de surfaces rugueuses. Certaines inclusions sont lâches et faciles à tomber et certaines sont pressées dans la plaque.

Fossé roulé (Rolled pit) : Les fossés roulés sont des bosses ou des fossés périodiques sur la surface d'une plaque d'acier qui sont ponctuels, en écailles ou en forme de bande. Ils sont distribués sur toute la longueur ou la section de la bande, principalement causés par des dommages aux cylindres de travail ou aux cylindres de tension.

Plis (Crease) : Un pli est un pli transversal vertical, avec un espacement régulier ou irrégulier à travers la bande, ou sur le bord de la bande. La raison principale est le rendement local le long de la direction de déplacement de la bande dans le processus de déroulage.

Craquelures (Crazing) : La craquelure est le phénomène qui produit des fissures à la surface d'un matériau.

Surface piquée (Pitted surface) : Les piqûres sont une forme de corrosion qui se concentre sur une très petite gamme de surfaces métalliques et pénètre à l'intérieur du métal. Les piqûres sont généralement de faible diamètre mais profondes.

Rayures (Scratches) : Une rayure est une marque d'abrasion sur une surface.

Écaille roulée (Rolled in scale) : Un défaut de calamine se produit lorsque la calamine est roulée dans le métal au cours du processus de laminage.

Plis de taille (Waistfolding) : Il y a des plis évidents dans les parties défectueuses, un peu plus populaires, un peu comme des rides, indiquant que la déformation locale du défaut est trop importante. La raison en est le faible carbone ^[4].

3. Vision artificielle pour l'inspection des surfaces des produits laminés

3.1. Système de vision

Un système de vision industriel est une application qui associe du matériel et du logiciel afin d'obtenir et de traiter des images, ainsi que de fournir des instructions opérationnelles pour accomplir d'autres tâches au niveau du processus ou pour prendre des décisions. Le système conçu offre de multiples bénéfices, ce qui en fait une solution efficace pour diverses applications. Ces dernières peuvent inclure différentes tâches, telles que l'aide dans la production, le guidage de robot, le contrôle des assemblages de pièces, ou encore l'inspection pour identifier les formes et évaluer la conformité dimensionnelle, ainsi que pour détecter et classer les défauts^[6].

La possibilité d'avoir accès à cette solution bénéfique, le système de vision, a renforcé les chercheurs et les industriels dans leur décision, les a incités à innover dans ce domaine et à multiplier les objectifs lors de sa production. On retrouve parmi ces objectifs ceux qui sont stratégiques tels que :

- L'amélioration de la qualité et l'augmentation de la productivité.
- La réduction des coûts et la minimisation des temps d'arrêts ; Et d'autres objectifs, plus techniques, qui permettent au système d'être performant dans la prise en charge des différentes tâches. Ces objectifs peuvent être :
- L'identification des défauts avec le minimum d'erreurs de classification.
- La capacité de prise en charge des considérations du temps réel.
- Les possibilités d'archivage des défauts, voire d'extension des bases de données.

Un système de vision est généralement constitué d'une ou plusieurs caméras positionnées face à la zone à examiner, d'un module d'acquisition pour enregistrer les images, d'un éclairage pour assurer une meilleure qualité d'image et de calculateur(s) pour le traitement et l'analyse de l'information (figure I.5).

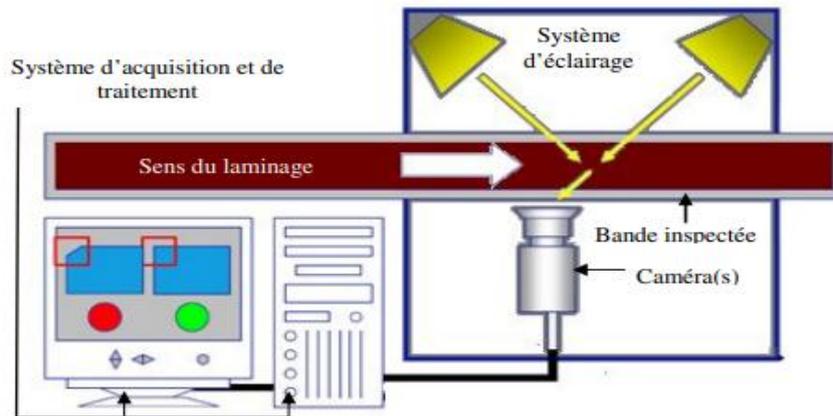


Figure I.5. Schéma simplifié d'un système de vision.

3.2. Caractéristiques et conditions de travail

3.2.1. Nature des produits objets d'inspection

Les caractéristiques du produit à inspecter sont intégrées au cahier des charges pour la mise au point d'un système de vision. En plus du design de la surface du produit, qui influence la réflexion de la lumière, sa taille est une autre considération lors de la sélection du type et du nombre de caméras requis. Dans certaines situations d'inspection pour détecter les imperfections de surface, comme celles concernant les produits longs, le système peut être équipé de trois à cinq caméras afin de couvrir l'ensemble de la surface du produit rond.

Tandis que dans les processus de laminage des produits plats (tôles d'acier), et avec des caméras dont la taille du capteur actif peut couvrir toute la largeur du produit, un minimum de deux voire trois systèmes de caméras (figure I.6) sont nécessaires pour visionner les deux faces de la bande ainsi que ses bords ^[3].

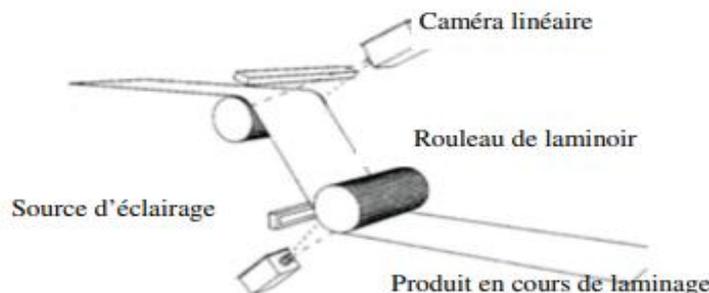


Figure I.6. Principe de la visualisation de toute la largeur des deux faces par caméras haute résolution en laminage de produits plats.

3.2.2. Eclairage

Dans les applications de vision industrielle, l'éclairage de la zone d'intérêt permet de mettre en valeur les détails de la surface inspectée avec le meilleur contraste.

Il existe de nombreuses configurations dans ce domaine, et il est possible de choisir, dans certains cas, une technique d'éclairage adaptée qui pourrait améliorer une image en atténuant certaines caractéristiques ou en accentuant d'autres.

Ces derniers n'ont aucune limitation en termes d'encombrement ou de distance. La figure I.7 montre quelques variantes de positionnement de la source d'éclairage par rapport aux deux autres éléments caméra et objet à inspecter [5].

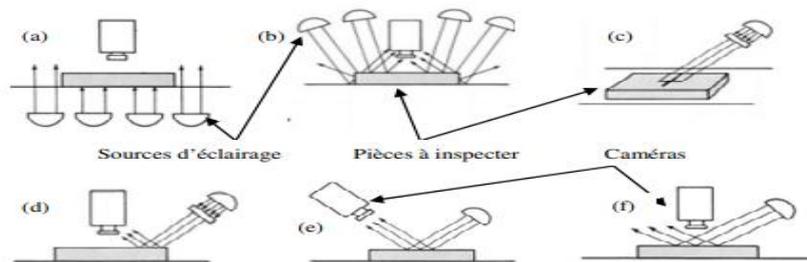


Figure I.7. Configurations couramment utilisées pour l'éclairage en système de vision industrielle.

(a): Rétro éclairage « Backlight » pour le contraste pièce-environnement, (b) : Eclairage frontal pour l'extraction de caractéristiques à la surface, (c) : Eclairage structuré pour information dimensionnelle et calcul de volume, (d, e, f) : Eclairage directionnels pour la recherche de caractéristiques tridimensionnelles.

3.2.3. Caméra et capteurs optiques

L'introduction des semi-conducteurs dans la technologie des caméras a permis non seulement une plus grande intégration et miniaturisation des circuits, mais aussi la réalisation de capteur plus robustes vis-à-vis du bruit et des fluctuations de l'éclairage, et la réduction de leur temps de réponse pour pouvoir capter des objets défilant à grande vitesse. Généralement, Les caméras CMOS sont d'une plus grande intégration avec des circuits aux dimensions inférieures, mais au détriment des aspects qualité de l'image, et à l'opposé, de meilleures performances en qualité d'image et en flexibilité aux dépens de l'aspect miniaturisation sont obtenues avec les caméras CCD.



Figure I.8. Caméras d'inspection.

Conclusion

Ce chapitre présente une vision d'ensemble des opérations de laminage, mettant en évidence à la fois les difficultés liées à la fabrication de produits laminés et les avancées technologiques qui améliorent la qualité et influencent l'avenir de l'industrie. Il est crucial de continuer à optimiser les processus de production et d'intégrer des technologies avancées comme la vision industrielle afin de satisfaire les demandes croissantes du marché et d'assurer la compétitivité à long terme des entreprises actives dans ce domaine.

CHAPTER 02

Apprentissage profond et vision artificielle

Introduction

Dans ce chapitre, nous considérons en détail les progrès de l'intelligence artificielle et du deep learning, des domaines qui transforment notre manière de traiter les données et d'interagir avec les machines. Dans un premier temps, nous avons considéré les principes essentiels de l'apprentissage profond en soulignant ses principes fondamentaux et les paramètres essentiels tels que la régularisation et la normalisation.

Par la suite, nous avons exploré le domaine des réseaux de neurones artificiels, en examinant leur structure, leur fonctionnement et les diverses architectures, en mettant l'accent sur les réseaux de neurones convolutifs qui sont les plus dominants dans le domaine de la vision numérique.

En bref, ce chapitre présente une analyse approfondie des avancées récentes en intelligence artificielle et en deep learning, fournissant un aperçu des principes essentiels et des architectures essentielles qui influencent l'avenir de la technologie et de l'interaction entre les êtres humains et les machines.

1. L'intelligence Artificielle

L'intelligence artificielle (IA) désigne la capacité d'un système informatique à exécuter des tâches qui nécessitent généralement une intelligence humaine. Cette discipline de l'informatique vise à créer des programmes et des machines capables de percevoir leur environnement, d'apprendre, de raisonner et de prendre des décisions en fonction des données disponibles. L'IA comprend diverses techniques telles que l'apprentissage profond, les réseaux neuronaux, le traitement du langage naturel et la vision par ordinateur. L'objectif principal de l'IA est de développer des systèmes capables de résoudre des problèmes complexes et de réaliser des tâches de manière autonome, ce qui peut entraîner des applications dans de nombreux domaines tels que la mécanique, la médecine, la robotique, la finance, et bien d'autres.

2. Apprentissage profond (Deep Learning)

L'apprentissage profond est un domaine de recherche récent dans le domaine de l'apprentissage automatique (ML) qui s'inspire de la structure et du fonctionnement du cerveau. Les algorithmes utilisés dans l'apprentissage profond peuvent apprendre des niveaux de représentation multiples pour modéliser des relations complexes entre les données. Ces dernières années, l'apprentissage profond a attiré une grande attention en raison de ses performances exceptionnelles, en particulier dans la classification d'images, grâce à l'utilisation d'algorithmes basés sur des réseaux de neurones qui imitent le cerveau.

L'apprentissage profond est une forme d'intelligence artificielle dérivée de l'apprentissage automatique, où la machine est capable d'apprendre et de s'améliorer par elle-même en fonction du volume de données disponibles. Les progrès réalisés dans le domaine du deep learning ont été rendus possibles grâce à l'augmentation de la puissance de calcul des ordinateurs et au développement de vastes bases de données.

L'apprentissage profond repose sur des réseaux neuronaux artificiels composés de milliers d'unités, appelées neurones, qui effectuent des opérations simples. Les résultats d'une couche de neurones sont ensuite utilisés comme entrées pour les calculs de la couche suivante, et ainsi de suite. L'illustration suivante montre la relation entre les trois concepts : intelligence artificielle (AI), apprentissage automatique (ML) et apprentissage en profondeur (DL).

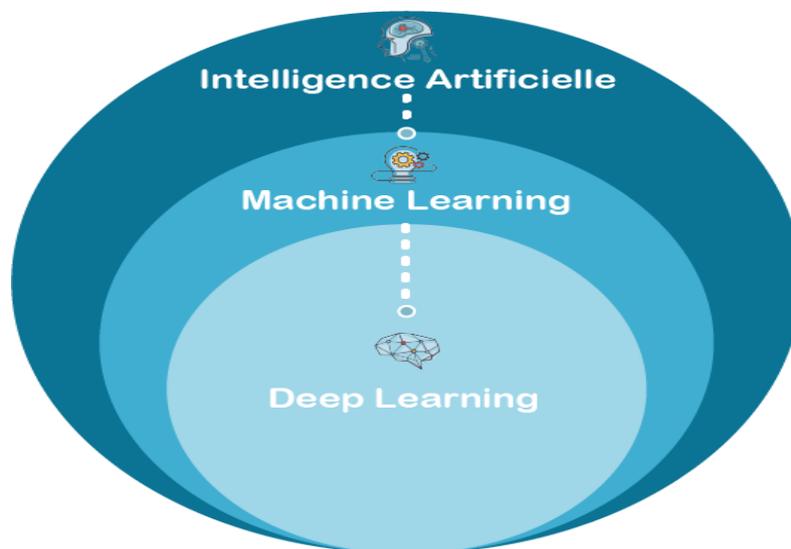


Figure II.1. La relation entre IA, ML et le DL

2.1. Principe de fonctionnement

Comme à l'intérieur du cerveau humain, les signaux voyagent entre les neurones du cerveau artificiel. Le secret de cette prouesse repose en grande partie sur les algorithmes. Dans le cas de la reconnaissance visuelle, pour être performant, l'algorithme d'apprentissage profond doit être capable d'identifier toutes les formes existantes et dans tous les angles. Ainsi, il sera capable de détecter une voiture sur la route au milieu du paysage. Ceci n'est possible que si la machine a suivi un entraînement poussé. Et ceci passe par la visualisation de milliers de photographies sur lesquelles apparaît une voiture, de toutes les formes et dans tous les angles possibles. Lorsque l'image nouvelle apparaît, elle est envoyée au réseau de neurones qui se charge de les analyser et de déterminer si l'objet au milieu du cliché est bel et bien une voiture. La machine a gagné son pari ? Elle garde sa bonne réponse au chaud, car elle l'aidera à résoudre d'autres situations similaires le jour où elle devra reconnaître une autre voiture.

3. Les réseaux de neurones artificiels

Les réseaux de neurones sont des constructions abstraites simulant l'activité d'un réseau de neurones biologique simplifiés. Ils sont utilisés dans apprentissage profond pour construire un modèle à partir de données existantes dans le but d'effectuer des prédictions sur de nouvelles données soit à l'aide de régression, dans le cas continu, ou de classification, dans le cas discret ^[10].

3.1. La structure d'un neurone artificiel

La figure II.2 montre la structure d'un neurone artificiel. Chaque neurone artificiel est un processeur élémentaire. Il reçoit un nombre variable d'entrées en provenance de neurones en amonts et à chacune de ces entrées est associé un poids \mathbf{W} abréviation de weight qui représente la force de la connexion. Chaque processeur élémentaire est doté d'une sortie unique, qui se ramifie ensuite pour alimenter un nombre variable de neurones avals. A chaque connexion est associée un poids.

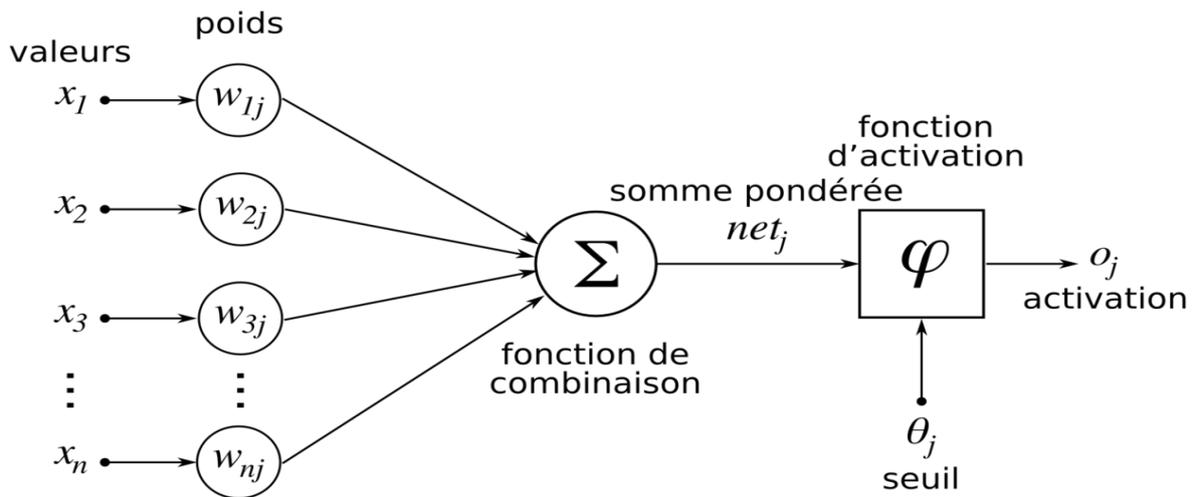


Figure II.2. Structure d'un neurone artificiel.

- Un neurone possède des entrées.
- Chaque entrée possède un poids.
- La sortie est une fonction du poids et des entrées :

$$Y = W_1 * X_1 + W_2 * X_2 + \dots \dots \dots (2.1)$$

a) Fonctions d'activation (ou fonction de transfert)

Les fonctions d'activation sont des éléments clés des réseaux de neurones artificiels, utilisés dans le domaine de l'apprentissage automatique et de l'intelligence artificielle. Elles sont appliquées à la sortie de chaque neurone dans une couche du réseau pour introduire de la non-linéarité, permettant ainsi au réseau de modéliser des relations complexes dans les données. Voici quelques-unes des fonctions d'activation les plus couramment utilisées [10] :

- **Sigmoïde**

La fonction sigmoïde (ou fonction logistique) est définie comme :

$$f(x) = \frac{1}{1 + e^{-x}} \dots \dots \dots (2.2)$$

Elle produit une sortie dans l'intervalle (0, 1), ce qui la rend utile pour les tâches de classification binaire.

- **Tangente hyperbolique (tanh)**

Cette fonction est similaire à la sigmoïde mais elle produit des valeurs dans l'intervalle (-1, 1). Elle est souvent utilisée dans les réseaux de neurones pour modéliser des relations non linéaires.

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \dots \dots \dots (2.3)$$

- **ReLU (Rectified Linear Unit)**

La fonction ReLU est définie comme :

$$f(x) = \begin{cases} 0 & \text{si } x < 0 \\ x & \text{si } x \geq 0 \end{cases} \dots\dots\dots (2.4)$$

Elle est devenue très populaire ces dernières années en raison de sa simplicité et de son efficacité. Elle introduit une non-linéarité en activant simplement les valeurs positives, tout en mettant à zéro les valeurs négatives.

- **ReLU paramétrique (PReLU)**

C'est une extension de la fonction ReLU qui permet d'avoir un coefficient de pente pour les valeurs négatives. Ce coefficient est appris pendant l'entraînement.

- **Softmax**

Cette fonction est couramment utilisée dans la couche de sortie des réseaux de neurones pour la classification multi classe. Elle prend un vecteur de scores réels en entrée et produit une distribution de probabilité sur plusieurs classes en utilisant la formule :

$$\text{Softmax}(x_i) = \frac{e^{x_i}}{\sum_i^n e^{x_i}} \dots\dots\dots (2.5)$$

Ces fonctions sont essentielles pour introduire de la non-linéarité dans les réseaux de neurones, ce qui leur permet de modéliser des relations complexes entre les données et d'effectuer des tâches d'apprentissage automatique et d'intelligence artificielle.

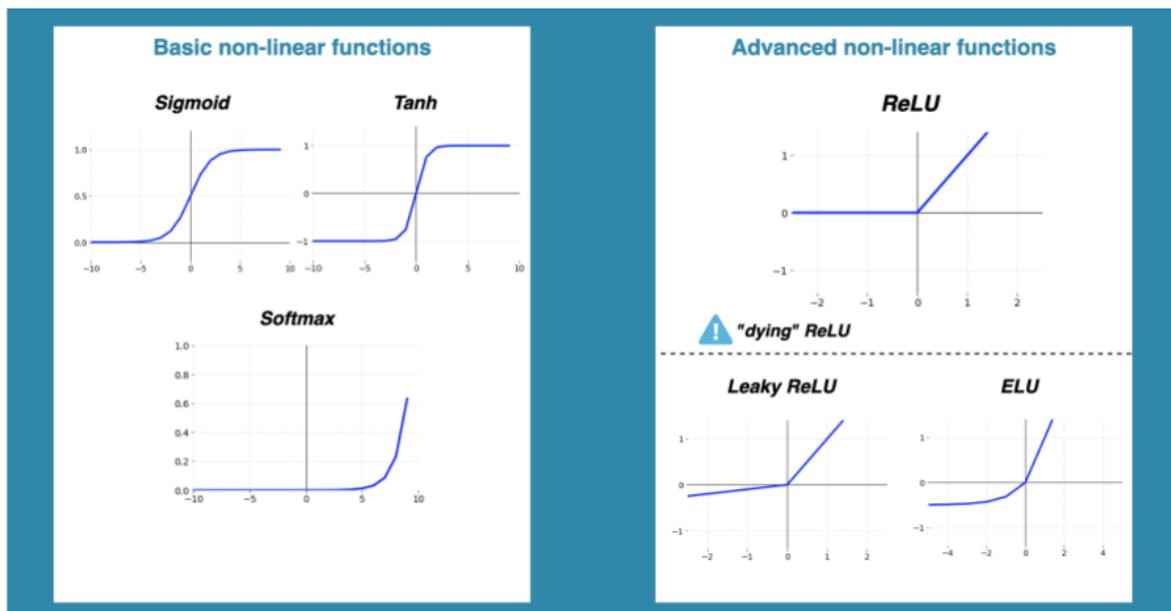


Figure II.3. Fonctions d'activation.

b) Calcul de l'erreur

Dans les réseaux de neurones artificiels, l'erreur que l'on désire minimiser est donnée par :

$$E_k = Y_i - \hat{Y}_i \dots \dots \dots (2.6)$$

Cette erreur peut être positive ($Y_i > \hat{Y}_i$) ou négative ($Y_i < \hat{Y}_i$).

- E_k est l'erreur de sortie.

- Y_i est la sortie désirée.

- \hat{Y}_i est la sortie réelle du réseau.

c) Fonction de coût

Pour réduire l'erreur, nous utilisons la fonction de coût (cost fonction). Autrement dit, nous allons améliorer \hat{Y}_i , pour réaliser cette amélioration, nous modifions les valeurs des poids :

$$J(w_1, w_2, b) = \frac{1}{n} \sum_{i=0}^n Y_i - \hat{Y}_i \dots \dots \dots (2.7)$$

Le but ici est de réduire cette fonction, et pour l'obtenir, nous utilisons un algorithme de descente du gradient.

d) Ajustement des poids (weights)

Descente de gradient signifie que l'on cherche à réduire l'erreur dans la direction de l'erreur, en descendant le long du gradient. Si l'on considère les entrées x_i du réseau associé respectivement aux poids W_i , alors :

$$W_{i+1} \leftarrow W_i - \alpha \frac{dJ}{dW} \dots \dots \dots (2.8)$$

- α est le taux d'apprentissage.

- W : le poids.

- $\frac{dJ}{dW}$: Descente de gradient.

3.2. Sur-apprentissage (Overfitting)

Le sur-apprentissage parfois appelé sur-ajustement est un problème courant dans l'apprentissage profond. Il se produit lorsqu'un modèle apprend à bien fonctionner sur les données d'apprentissage, mais ne parvient pas à se généraliser à de nouvelles données inédites. Par conséquent, le modèle ne fonctionne pas bien sur de nouvelles données qu'il n'a pas vues pendant la formation.

Le sur apprentissage se produit souvent lorsque le modèle est trop complexe pour la quantité de données disponibles, ce qui lui permet de mémoriser les données d'apprentissage au lieu de comprendre les modèles généraux qui s'appliquent aux nouvelles données.

Pour éviter le sur apprentissage, différentes techniques peuvent être utilisées, telles que la validation croisée, la régularisation, l'augmentation des données, la réduction de la complexité du modèle, etc. L'objectif de ces méthodes est de créer un modèle qui se généralise mieux à des données inédites, plutôt que de simplement mémoriser les données d'apprentissage.

3.3. Les paramètres de l'apprentissage

3.3.1. La régularisation

La régularisation fait référence à un ensemble de techniques différentes qui réduisent la complexité d'un modèle de réseau de neurones pendant l'entraînement et empêchent ainsi le sur ajustement. Il existe deux techniques de régularisation très populaires et efficaces appelées L1 et L2 [7].

A. Régularisation L1

Ce type de régularisation est également connu sous le nom de régularisation Lasso. Il ajoute à la fonction de coût un terme proportionnel à la valeur absolue des coefficients de poids :

$$J(w_1, w_2, b) = \frac{1}{n} \sum_{i=0}^n (Y_i - \hat{Y}_i)^2 + \lambda \times (|w_1| + |w_2|) \dots \dots \dots (2.11)$$

Il tend à réduire certains des coefficients de poids à zéro. La somme des termes est multipliée par lambda, qui contrôle le degré de régularisation. Si lambda est trop élevé, le modèle sera simple et le risque d'underfitting apparaît.

B. Régularisation L2

La régularisation L2, ou régularisation Ridge, ajoute un terme à la fonction de coût qui est proportionnel au carré des coefficients de poids :

$$J(w_1, w_2, b) = \frac{1}{n} \sum_{i=0}^n (Y_i - \hat{Y}_i)^2 + \lambda \times (|w_1|^2 + |w_2|^2) \dots \dots \dots (2.12)$$

Ce terme tend à réduire tous les coefficients de poids, mais contrairement à la régularisation L1, il ne met aucun coefficient de poids à zéro.

C. Autres techniques de régularisation

Outre les régularisations L1 et L2, d'autres techniques de régularisation peuvent être utilisées dans les modèles d'apprentissage profond. Parmi ces techniques, nous trouvons l'abandon (dropout) et l'arrêt précoce (early stopping).

- **Abandon (Dropout)**

La régularisation par abandon est appliquée pendant l'entraînement des réseaux neuronaux profonds. Elle élimine aléatoirement un certain nombre de neurones du réseau à chaque itération de l'entraînement, ce qui oblige le réseau à s'appuyer sur plusieurs sous-ensembles de neurones et réduit donc overfitting.

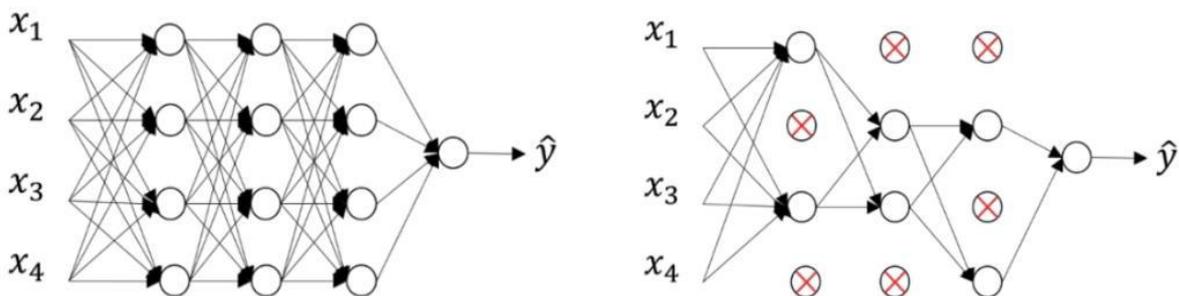


Figure II.4. Réseau de neurones avec dropout (à droite) et sans dropout (à gauche).

- **L'arrêt précoce (Early stopping)**

L'arrêt précoce est une autre technique utilisée pour éviter le sur ajustement (overfitting) dans les modèles d'apprentissage profond. Cette technique consiste à arrêter l'entraînement du modèle lorsque les performances sur l'ensemble de validation commencent à se détériorer. Cela permet d'éviter que le modèle sur apprenne les données d'apprentissage et ne se généralise pas bien sur des données qui n'ont pas été vues auparavant.

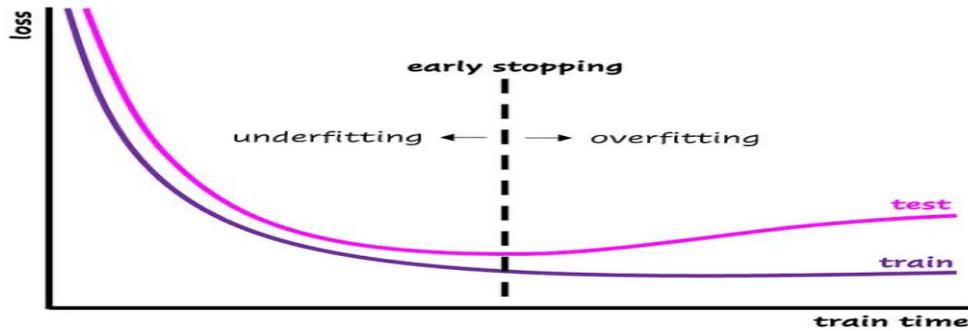


Figure II.5. L'arrêt précoce (Early stopping)

3.3.2. Techniques d'optimisation

A. Mini Batch - Descente de gradient stochastique

MB-SGD est une extension de l'algorithme SGD (Stochastique Gradient Descente) : utilisée pour la minimisation d'une fonction objectif qui est écrite comme une somme de fonctions différentiables (Figure II.6). Il surmonte la complexité chronophage de SGD en prenant un lot de points sous-ensemble de points de l'ensemble de données pour calculer le dérivé. Ces derniers temps, les algorithmes d'optimisation adaptative gagnent en popularité en raison de leur capacité à converger rapidement. Ces algorithmes utilisent des statistiques d'itérations précédentes pour accélérer le processus de convergence^[8].

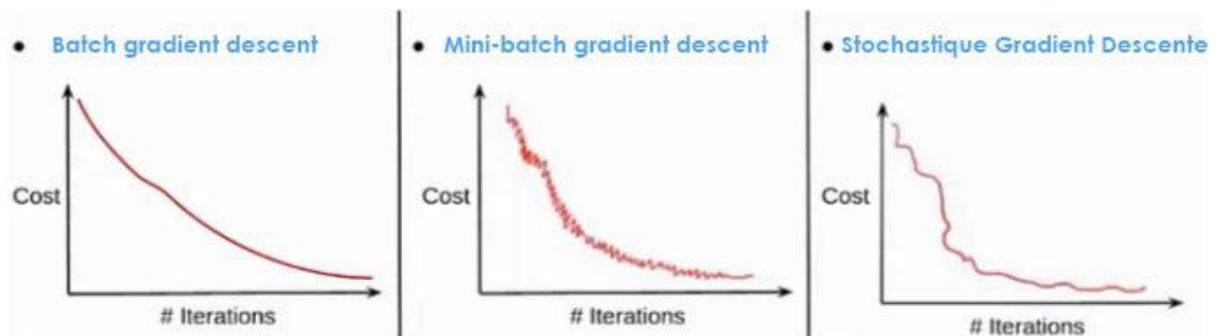


Figure II.6. Le gradient descendant en batch et le gradient descendant en mini-batch et Descente de gradient stochastique.

B. Optimiseur de Momentum

Il s'agit d'un algorithme d'optimisation adaptative qui utilise de manière exponentielle des gradients moyens pondérés sur les itérations précédentes pour stabiliser la convergence, ce qui se traduit par une optimisation plus rapide. Cela se fait en ajoutant une fraction (gamma) aux valeurs de l'itération précédente. Essentiellement, le terme d'élan augmente lorsque les points de gradient sont dans les mêmes directions et diminue lorsque les gradients

fluctuent. En conséquence, la valeur de la fonction de perte converge plus rapidement que prévu.

C. RMSProp (Root Mean Squared Prop)

L'idée clé de la Root Mean Square Propagation (propagation quadratique moyenne) est que le gradient est normalisé par sa racine carrée moyenne.

Dans l'équation ci-dessous, l'élévation au carré du gradient indique que chaque élément du vecteur est élevé au carré individuellement.

$$v_{t+1} = \alpha v_t + (1 - \alpha) \nabla f_i(w_t)^2 \dots \dots \dots (2.13)$$

$$w_{t+1} = w_t - \gamma \frac{\nabla f_i(w_t)}{\sqrt{v_{t+1} + \epsilon}} \dots \dots \dots (2.14)$$

Où γ est le taux d'apprentissage global, ϵ est une valeur de l'ordre de 10^{-7} ou 10^{-8} afin d'éviter les erreurs dû à la division par zéro, et v_{t+1} est l'estimation du 2ème moment.

D. Estimation adaptative du moment (Adam)

C'est une combinaison de RMSProp et Momentum. Cette méthode calcule le taux d'apprentissage adaptatif pour chaque paramètre. En plus de stocker la moyenne décroissante précédente des gradients carrés, il contient également la moyenne du gradient passé similaire à Momentum. Ainsi, Adam se comporte comme une balle lourde avec frottement qui préfère les minima plats en surface d'erreur.

$$m_{t+1} = \beta m_t + (1 - \beta) \nabla f_i(w_t) \dots \dots \dots (2.15)$$

$$v_{t+1} = \alpha v_t + (1 - \alpha) \nabla f_i(w_t)^2 \dots \dots \dots (2.16)$$

$$w_{t+1} = w_t - \gamma \frac{m_t}{\sqrt{v_{t+1} + \epsilon}} \dots \dots \dots (2.17)$$

Où m_{t+1} est la moyenne mobile exponentielle du momentum ^[8].

3.3.3. Les techniques d'augmentation d'image

Les techniques d'augmentation des données sont utilisées pour générer des échantillons de formation artificiels et étendre l'ensemble de données de formation. Elles ont été largement utilisées pour améliorer la performance du modèle et réduire overfitting, en particulier notamment pour l'apprentissage avec un petit ensemble de données ou un ensemble de données déséquilibré. Les techniques d'augmentation d'image couramment utilisées

techniques d'augmentation d'image couramment utilisées incluent le recadrage, le retournement, la rotation de l'image.

3.3.4. Normalisation

La normalisation d'un vecteur (par exemple, une colonne dans un jeu de données) consiste à diviser les données de la norme vectorielle. Typiquement, nous l'utilisons pour obtenir la distance euclidienne du vecteur égale à une certaine valeur prédéterminée ^[8].

3.3.5. Indicateurs de performance en classification

- **Matrice de confusion**

Une matrice de confusion est un tableau qui résume le nombre de prédictions correctes et incorrectes faites par un modèle de classification sur un ensemble de données de test. Elle est généralement représentée sous la forme d'une matrice carrée dont les lignes et les colonnes représentent respectivement les étiquettes de classe prédites et les vraies étiquettes de classe. Les entrées de la matrice indiquent le nombre d'échantillons de test qui appartiennent à une certaine classe, et combien d'entre eux ont été classés correctement ou incorrectement par le modèle. Une matrice de confusion peut fournir une analyse détaillée des performances du modèle, y compris des mesures telles que l'exactitude, la précision, le rappel et le score F1 pour chaque classe. Elle peut être utilisée pour identifier les domaines spécifiques dans lesquels le modèle commet des erreurs et pour diagnostiquer les problèmes liés aux prédictions du modèle ^[9].

- **Rapport de classification**

Un rapport de classification est un résumé des principales mesures de performance d'un modèle de classification, notamment la précision, le rappel et le score F1, ainsi que la précision globale du modèle. Il fournit un aperçu concis des performances du modèle, généralement ventilé par classe, et peut être utilisé pour évaluer rapidement les forces et les faiblesses du modèle. Le rapport est souvent présenté sous la forme d'un tableau, chaque ligne représentant une classe et les colonnes indiquant diverses mesures de performance. Le rapport peut également inclure d'autres mesures telles que le soutien (le nombre d'échantillons de test appartenant à une classe particulière) et les moyennes macros et micro des mesures de performance pour toutes les classes.

Précision : La fraction de vrais positifs (TP, prédictions correctes) par rapport au nombre total de résultats pertinents, c'est-à-dire la somme des TP et des faux positifs (FP). Pour les

problèmes de classification multi-classes, la moyenne de P est calculée entre les classes. La formule de la précision est la suivante :

$$\text{Précision} = \frac{TP}{TP+FP} \dots\dots\dots(2.18)$$

Rappel : La fraction de TP par rapport à la quantité totale de TP et de faux négatifs (FN). Pour les problèmes de classification multi-classes, la moyenne de R est calculée pour toutes les classes. La formule suivante est celle du rappel :

$$\text{Rappel} = \frac{TP}{TP+FN} \dots\dots\dots(2.19)$$

F1 score : La moyenne harmonique de la précision et du rappel. Pour les problèmes de classification multi-classes, la moyenne de F1 est calculée pour toutes les classes. La formule suivante est celle du score F1 :

$$\text{F1 score} = 2 * \frac{TP*FP}{TP+FP} \dots\dots\dots(2.20)$$

4. Architecture d'un réseau de neurones artificiel

En général, un réseau de neurones est composé d'un ensemble de couches successives, où chaque couche reçoit ses entrées à partir des sorties de la couche précédente. Cela signifie que l'ensemble du réseau est entièrement connecté. Chaque couche est constituée de neurones qui ne sont pas connectés entre eux, mais qui reçoivent des informations numériques des neurones voisins. L'ensemble des couches comprend une couche d'entrée qui lit les valeurs d'entrée, une couche de sortie qui fournit les résultats du système, et entre ces deux se trouvent une ou plusieurs couches cachées qui participent au processus de transfert des informations.

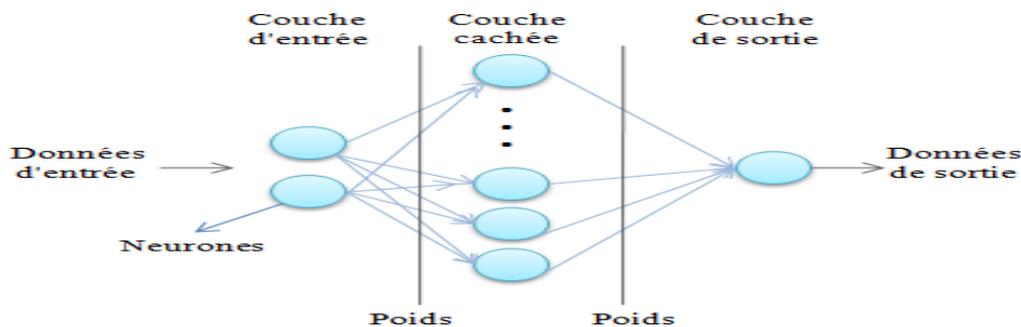


Figure II.7. Architecture d'un réseau de neurones.

4.1. Fonctionnement des réseaux de neurones artificiel

Le concept de réseau de neurones repose sur trois étapes principales :

1. Pour chaque neurone dans une couche, multiplier la valeur d'entrée par le poids.
2. Ensuite, pour chaque couche, additionner toutes les pondérations des neurones et ajouter un biais.
3. Enfin, appliquer la fonction d'activation sur cette valeur pour calculer une nouvelle sortie.

4.2. Les différents types de réseaux de neurones artificiels

Il existe différents types de réseaux de neurones, et ils sont classés en fonction du nombre de nœuds cachés du modèle ou encore du nombre d'entrées et de sorties de chaque nœud.

La propagation des informations entre les différents neurones peut varier et dépend du type de réseaux de neurones^[11].

Il existe plusieurs types de réseaux de neurones :

4.2.1. Les réseaux de neurones à propagation avant (feed-forward)

C'est la variante la plus simple, où l'information se déplace uniquement dans une direction. Elle traverse directement les nœuds de traitement (couches cachées) à partir de l'entrée, puis se dirige vers les sorties. Dans ce type de réseau, il n'y a pas de cycles ou de boucles.

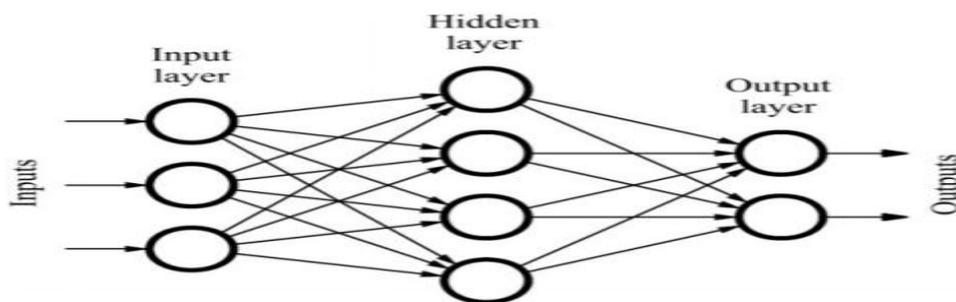


Figure II.8. Les réseaux de neurones à propagation avant.

4.2.2. Les réseaux de neurones récurrents

Ce mode d'apprentissage est un peu plus complexe. Ces réseaux comportent au moins un cycle dans leur structure, ce qui leur permet de sauvegarder les résultats produits par les nœuds de traitement et de les utiliser pour nourrir le modèle. Les réseaux de neurones récurrents sont utilisés dans des domaines tels que la reconnaissance automatique de formes et la traduction automatique de la parole.

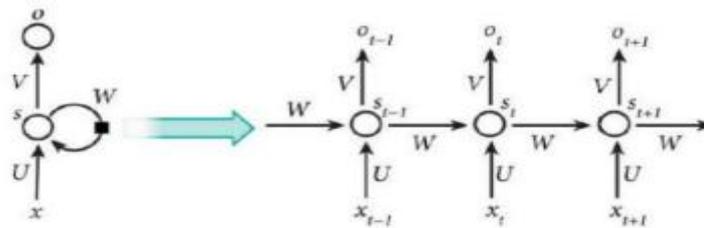


Figure II.9. (À gauche) Un RNN (à droite) sa version déroulé source.

Le schéma ci-dessus montre un RNN déroulé. En déroulant, nous signifions simplement qu'on montre le réseau pour la séquence complète.

Par exemple, si la séquence qui nous intéresse est une phrase de 5 mots, le réseau serait déroulé en un réseau de neurones de 5 couches, une couche pour chaque mot.

Les formules qui régissent les calculs dans un RNN sont les suivantes :

- X_t est l'entrée au moment t .
- U et W sont les paramètres que le réseau va apprendre des données d'apprentissage.
- S_t est l'état caché au moment t , C 'est la « mémoire » du réseau. S_t est calculé en fonction de l'état Caché précédent et de l'entrée à l'étape actuelle :

$$(W * S_{t-1} + X_t * U) = S_t \dots \dots \dots (2.21)$$

4.2.3. Les réseaux de neurones convolutifs

Inspirés par le fonctionnement du cortex visuel des animaux, ces réseaux traitent les données en les faisant passer à travers plusieurs couches, de la couche d'entrée à la couche de sortie. Les réseaux de neurones convolutifs sont utilisés dans des domaines tels que la reconnaissance faciale, la reconnaissance d'images et de vidéos, ainsi que le traitement du langage naturel.

5. Principe de fonctionnement des réseaux de neurones convolutifs

Dans le domaine de l'apprentissage profond, les réseaux de neurones convolutifs (CNN/ConvNet) sont une classe de réseaux de neurones profonds largement utilisés pour l'analyse d'images visuelles. Contrairement à ce que l'on pourrait penser généralement lorsqu'on évoque les réseaux de neurones, les CNN ne se contentent pas d'utiliser des multiplicateurs matriciels. Ils font appel à une technique spéciale appelée "bypass" ou "skip connection".

La convolution, en mathématiques, est une opération arithmétique qui permet de combiner deux fonctions pour en produire une troisième, exprimant ainsi la manière dont la forme de l'une est modifiée par l'autre. Dans le contexte des CNN, la convolution est utilisée pour extraire des caractéristiques des images en appliquant des filtres spécifiques sur des régions locales. Cela permet aux CNN de capturer des motifs et des informations significatives au sein des images.

$$S(t) = (f \times g) \times (t) \div \int_{-\infty}^{+\infty} f(r)g(t-r) dr \dots \dots \dots (2.22)$$

f et g sont deux fonctions ayant un paramètre commun, t.

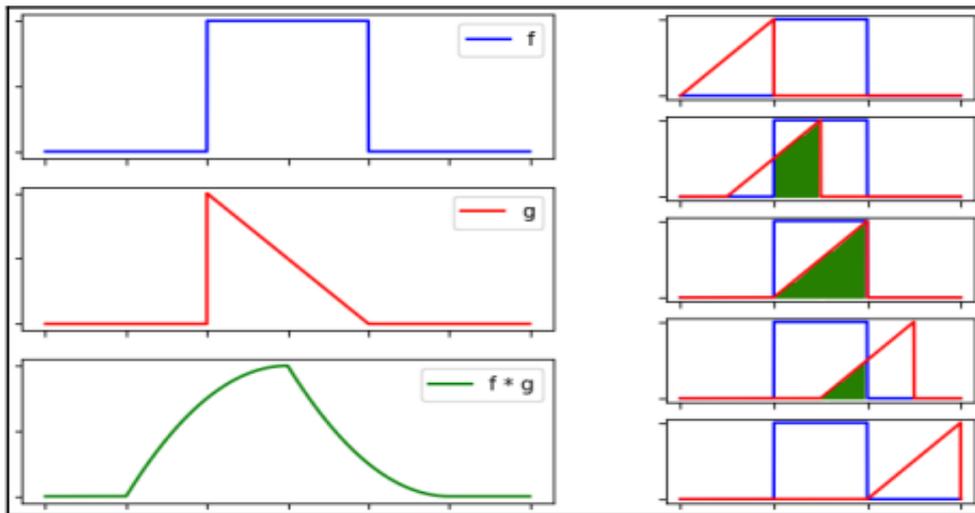


Figure II.10. À gauche : une convolution, où g est décalé et inversé ; à droite : une illustration étape par étape d'une opération de convolution.

En résumé, les réseaux de neurones convolutifs (CNN/ConvNet) sont une classe de réseaux de neurones profonds largement utilisés dans l'analyse d'images. Ils se distinguent par l'utilisation de la convolution, une opération mathématique qui permet de capturer des caractéristiques visuelles importantes. Les CNN utilisent également des connexions spéciales appelées "bypass" ou "skip connection" pour améliorer la performance et la capacité d'apprentissage du réseau.

5.1. Les images numériques

Les images numériques sont constituées d'un ensemble de pixels (Picture éléments), juxtaposés en lignes et en colonnes. Le pixel (qui correspond à un point ou petit carré) est le plus petit élément que l'on peut trouver dans une image. Chaque pixel possède des caractéristiques propres, couleurs, luminosité, brillance, qui permettent de les différencier et de composer les images ^[12].

5.1.1. Les différents types de format d'image numérique

- **Image couleur RVB** : L'œil humain analyse la couleur à l'aide de trois types de cellules photo « les cônes ». Ces cellules sont sensibles aux basses, moyennes, ou hautes fréquences (rouge, vert, bleu). Pour représenter la couleur d'un pixel, il faut donc donner trois nombres, qui correspondent au dosage de trois couleurs de base : Rouge, vert, bleu. On peut ainsi représenter une image couleur par trois matrices chacune correspondant à une couleur de base.

- **Image d'intensités** : C'est une matrice dans laquelle chaque élément est un réel compris entre 0 (noir) et 1 (blanc). On parle aussi d'image en niveaux de gris, car les valeurs comprises entre 0 et 1 représentent les différents niveaux de gris.

- **Image binaire** : Une image binaire est une matrice rectangulaire dans l'élément valent 0 ou 1. Lorsque l'on visualise une telle image, les 0 sont représentés par du noir et les 1 par du blanc.

5.1.2. Les caractéristiques d'images numériques

L'image est un ensemble structuré d'information caractérisé par les paramètres suivants :

a. Les pixels

Une image numérique est constituée d'un ensemble de points appelés pixels pour former une image (Figure II.11). Le pixel représente ainsi le plus petit élément constitutif d'une image numérique. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituantes.

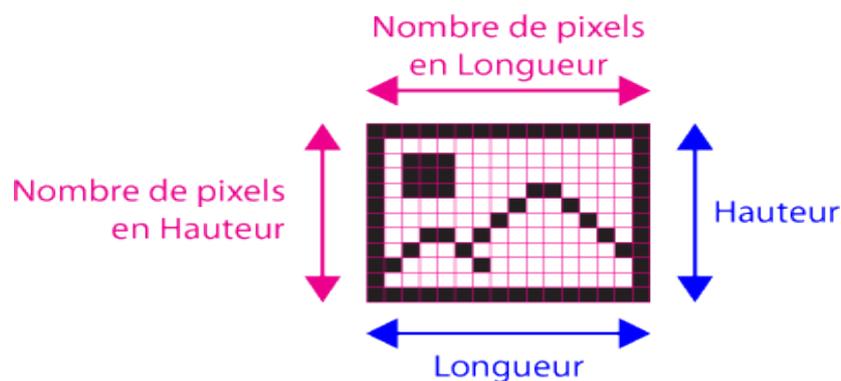


Figure II.11. Les dimensions d'images.

b. Dimension et Résolution

La dimension est la taille de l'image. Elle se présente sous forme d'une matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image. Par contre, la résolution est la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production d'images. Sur les moniteurs d'ordinateur.

La résolution est exprimée en nombre de pixels par unité de mesure (pouce ou centimètre). On utilise aussi le mot résolution pour désigner le nombre total de pixels horizontaux et verticaux sur un moniteur. Plus ce nombre est grand, plus la résolution est meilleure.

c. Niveau de gris

C'est la valeur d'intensité lumineuse d'un pixel. Cette valeur peut aller du noir (0) jusqu'au blanc (255) en passant par les nuances qui sont contenues dans l'intervalle [0, 255]. Elle correspond en fait à la quantité de la lumière réfléchi. Pour 8 bits, on dispose de 256 niveaux de gris dont 40 sont reconnus à l'œil nu. Plus le nombre de bits est grand plus les niveaux sont nombreux et plus la représentation est fidèle^[12].

5.2. Schéma conceptuel d'un réseau de neurones convolutifs

Un réseau de neurones convolutif n'est pas seulement un réseau neuronal profond avec de nombreuses couches cachées. Il s'agit plutôt d'un réseau profond qui simule le fonctionnement du cortex visuel du cerveau pour reconnaître et classifier des images ou des vidéos, et pour découvrir un objet ou même une partie dans une image.

Le concept et le fonctionnement des réseaux de neurones convolutifs est différent des autres réseaux de neurones, en effet un réseau neuronal convolutif comporte deux parties distinctes avec une entrée dans laquelle une image en forme de matrice de pixels bidimensionnelle (avec 2 dimensions, noir et blanc), ou une image couleur avec 3 dimensions (couleurs : rouge, vert et bleu) ou une image multidimensionnelle (image satellitaire).

La première partie d'un réseau de neurones convolutif est la partie convolutif qui sert à extraire les caractéristiques de l'image. Ensuite, l'image passe par le fichier de séquence de filtre, ou le noyau d'enroulement, ce qui conduit à la création d'une nouvelle image appelée cartes de convolution. Généralement, les filtres intermédiaires réduisent la résolution de l'image.

Ensuite, les cartes des caractéristiques sont aplaties dans un vecteur de caractéristiques pour former les données d'entrée de la partie de couche entièrement connectée. Le rôle principal de cette couche (complètement connectée) est de combiner les caractéristiques contenues dans le vecteur de son entrée pour la classification des images ^[13].

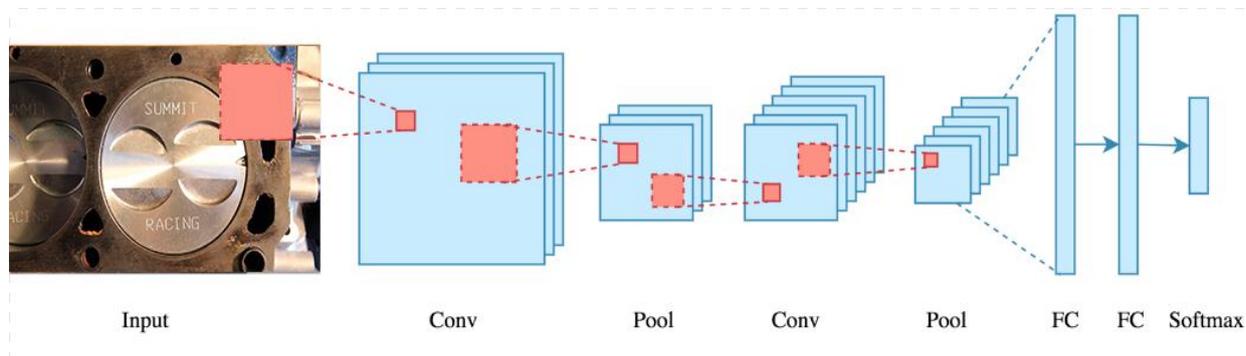


Figure II.12. Architecture générale d'un réseau de neurones convolutif (CNN/ConvNet).

5.2.1. Couche de convolution (CONV)

La couche de convolution est un élément essentiel des réseaux neuronaux convolutifs (CNN) et est généralement la première couche de ces réseaux.

Son objectif est de détecter la présence d'un ensemble de caractéristiques dans les images en entrée. Pour cela, elle effectue une opération de filtrage par convolution : elle fait "glisser" une fenêtre représentant la caractéristique souhaitée sur l'image et calcule le produit de convolution entre la fenêtre et chaque région balayée de l'image. Dans ce contexte, une caractéristique est considérée comme un filtre, les deux termes étant équivalents.

La couche de convolution reçoit plusieurs images en entrée et effectue la convolution de chaque image avec chaque filtre. Les filtres correspondent exactement aux caractéristiques que l'on souhaite détecter dans les images.

Pour chaque paire (image, filtre), on obtient une carte d'activation, également appelée carte de caractéristiques, qui indique l'emplacement des caractéristiques dans l'image : plus la valeur est élevée, plus la région correspondante de l'image ressemble à la caractéristique.

Contrairement aux méthodes traditionnelles, les caractéristiques ne sont pas prédéfinies selon un formalisme particulier, mais sont apprises par le réseau pendant la phase d'entraînement. Les noyaux des filtres représentent les poids de la couche de convolution. Ils sont initialisés puis mis à jour à l'aide de la rétro propagation du gradient.

C'est là toute la puissance des réseaux neuronaux convolutifs : ils sont capables de déterminer automatiquement les éléments discriminants d'une image en s'adaptant au

Chapitre 02 : Apprentissage profond et vision artificielle

problème donné. Par exemple, si le problème consiste à distinguer les chats des chiens, les caractéristiques automatiquement apprises peuvent décrire la forme des oreilles ou des pattes [9].

Lors de l'utilisation d'un CNN, il existe trois hyper paramètres importants que nous devons choisir :

- **Profondeur de la couche** : il s'agit du nombre de noyaux de convolution utilisés, ce qui correspond également au nombre de neurones associés à un même champ récepteur. Une profondeur plus élevée permet au réseau d'apprendre des caractéristiques plus complexes.
- **Le pas (Stride)**: il contrôle le chevauchement des champs récepteurs lors de la convolution. Un pas plus petit entraîne un chevauchement plus important et génère un volume de sortie plus grand.
- **La marge à 0 (Rembourage) ou zéro padding** : parfois, il est utile d'ajouter des zéros autour du volume d'entrée. La taille de ce "zéro padding" est le troisième hyper paramètre. Cette marge permet de contrôler la dimension spatiale du volume de sortie. En conservant la même surface que celle du volume d'entrée, elle peut aider à préserver les informations sur les bords de l'image.

En résumé, la couche de convolution est essentielle dans un CNN. Elle effectue la convolution des images avec des filtres pour extraire des caractéristiques importantes. Les hyper paramètres tels que la profondeur, le pas et la marge jouent un rôle crucial dans la manière dont la convolution est appliquée et dans la dimension du volume de sortie [9].

1	1	1	3
4	6	4	8
30	0	1	5
0	2	2	4

En commençant par le premier filtre, l'opération de convolution commence au coin supérieur gauche de la matrice de la même taille que le filtre de convolution. L'opération de convolution est la somme des produits des éléments qui sont situés sur les mêmes positions des deux matrices. Le résultat de 7 dans la matrice de résultat est trouvé comme suit :

$$(1 \times 1) + (1 \times 0) + (4 \times 0) + (6 \times 1) = 7$$

1	1	1	3
4	6	4	8
30	0	1	5
0	2	2	4

$$* \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} =$$

7		

Une autre opération de convolution est menée pour la prochaine sous-matrice :

1	1	1	3
4	6	4	8
30	0	1	5
0	2	2	4

$$* \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} =$$

7	5	

De la même manière, la troisième opération de convolution est menée, comme il est montré dans la figure

1	1	1	3
4	6	4	8
30	0	1	5
0	2	2	4

$$* \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} =$$

7	5	9

Une fois la rangée du haut terminée, la rangée suivante recommence à partir de la gauche.

1	1	1	3
4	6	4	8
30	0	1	5
0	2	2	4

$$* \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} =$$

7	5	9
4		

Le même processus se répète jusqu'à ce que la carte des caractéristiques du filtre donné soit produite, comme le montre la figure

1	1	1	3
4	6	4	8
30	0	1	5
0	2	2	4

$$* \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} =$$

7	5	9
4	7	9
32	2	5

5.2.2. Couche Pooling

Le regroupement de couches est une opération couramment utilisée après une couche convolutive dans un réseau de neurones convolutif (CNN). Cette opération permet de réduire la dimensionnalité des caractéristiques extraites et d'introduire une certaine invariance aux translations dans les données.

Il existe principalement deux types de regroupement couramment utilisés: le regroupement maximum (max pooling) et le regroupement moyen (average pooling).

Dans le regroupement maximum, une fenêtre (généralement de taille 2x2 ou 3x3) glisse sur la carte d'activation générée par la couche convolutive, et la valeur maximale dans chaque fenêtre est sélectionnée pour former une nouvelle carte d'activation réduite. Cela permet de conserver les caractéristiques les plus dominantes tout en réduisant la taille spatiale.

Dans le regroupement moyen, la même fenêtre glisse sur la carte d'activation, mais cette fois-ci, la valeur moyenne des activations dans chaque fenêtre est calculée pour former la nouvelle carte d'activation réduite. Cette opération permet de prendre en compte l'information globale de chaque région et de produire une représentation plus lisse.

Le regroupement de couches permet donc de réduire la dimension spatiale des caractéristiques tout en conservant les informations les plus importantes. Cela aide à réduire le nombre de paramètres et de calculs nécessaires dans le réseau, tout en introduisant une certaine invariance aux translations dans les données.

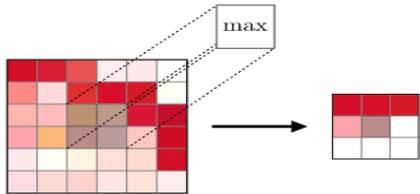
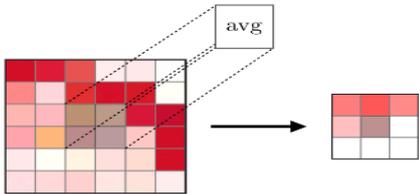
Type	Max pooling	Average pooling
But	Chaque opération de pooling sélectionne la valeur maximale de la surface	Chaque opération de pooling sélectionne la valeur moyenne de la surface
Illustration		
Commentaires	<ul style="list-style-type: none">• Garde les caractéristiques détectées• Plus communément utilisé	<ul style="list-style-type: none">• Sous-échantillonne la <i>feature map</i>• Utilisé dans LeNet

Figure II.13. Couche Pooling.

❖ Types de pooling

a. Max pooling : qui revient à prendre la valeur maximale de la sélection. C'est le type le plus utilisé car il est rapide à calculer (immédiat), et permet de simplifier efficacement l'image.

b. Mean pooling : (ou **average pooling**) : soit la moyenne des pixels de la sélection : on calcule la somme de toutes les valeurs et on divise par le nombre de valeurs. On obtient ainsi une valeur intermédiaire pour représenter ce lot de pixels

c. Sum pooling : c'est la moyenne sans avoir divisé par le nombre de valeurs (on ne calcule que leur somme).

5.2.3. Couche de correction Relu

Il est possible d'améliorer l'efficacité du traitement en intercalant entre les couches de traitement une couche qui va opérer une fonction mathématique (fonction d'activation) sur les signaux de sortie.

La fonction ReLU ($R(x)=\max(0, x)$) Cette fonction force les neurones à retourner des valeurs positives.

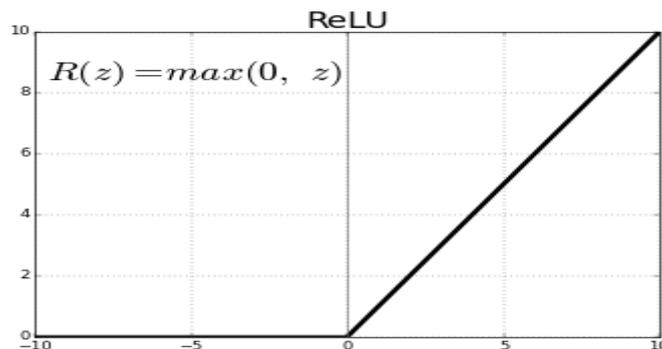


Figure II.14. Fonction d'activation Relu

5.2.4. Couche entièrement connectée (Fully-connected Neural Network)

Après plusieurs couches de convolution et de max-pooling, le raisonnement de haut niveau dans le réseau neuronal se fait via des couches entièrement connectées. Les neurones dans une couche entièrement connectée ont des connexions vers toutes les sorties de la couche précédente (comme on le voit régulièrement dans les réseaux réguliers de neurones). Leurs fonctions d'activations peuvent donc être calculées avec une multiplication matricielle suivie d'un décalage de polarisation ^[13].

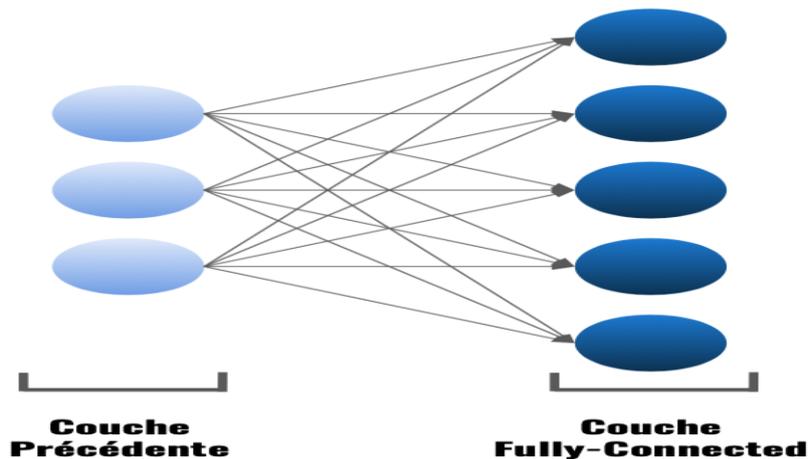


Figure II.15. Une couche fully-connected.

5.2.5. La couche de perte (LOSS)

La couche de perte est la dernière couche d'un réseau neuronal, responsable de calculer l'erreur entre les prédictions du réseau et les valeurs réelles. Lorsqu'il s'agit d'une tâche de classification où la variable aléatoire est discrète, avec des valeurs 0 ou 1 représentant l'appartenance ou non à une classe, la fonction de perte la plus courante et appropriée est l'entropie croisée (cross-entropy en anglais).

L'entropie croisée est une mesure de la différence globale entre deux distributions de probabilité, dans ce cas, celle de la prédiction du modèle et celle des valeurs réelles. Cette fonction est issue du domaine de la théorie de l'information. Formellement, elle est exprimée par l'équation suivante ^[14].

$$loss(x, class) = -\sum_{class=1}^c y_{x, class} \log(p_{x, class}) \dots\dots\dots (2.23)$$

5.3. Les différentes architectures des réseaux de neurone convolutif

Les architectures des réseaux de neurones de convolution (CNN) ont connu une évolution significative depuis les premières conceptions telles que LeNet-5 jusqu'aux modèles état de l'art comme EfficientNet. Ces architectures se caractérisent souvent par des couches de convolution empilées, alternant avec des couches de pooling pour réduire progressivement la dimensionnalité des données. Des innovations telles que les blocs résiduels de ResNet, les modules d'inception de GoogLeNet, et l'utilisation de couches de convolution de taille réduite dans VGGNet et EfficientNet ont permis de construire des réseaux plus profonds et plus efficaces. Ces architectures ont été essentielles pour des avancées majeures dans des domaines comme la reconnaissance d'images, la détection d'objets.

Parmi ces architectures, VGG16 se distingue comme l'une des plus influentes.

6. Le modèle VGG16

VGG16 est un réseau neuronal convolutif (CNN) largement utilisé pour la classification d'images. Il a été présenté pour la première fois en 2014 par le Visual Geometry Group (VGG) de l'Université d'Oxford. VGG16 est un réseau profond, avec 16 couches, qui s'est avéré très efficace pour la classification d'images. Il a obtenu des résultats de pointe sur un certain nombre d'ensembles de données de référence, y compris ImageNet.

VGG16 est un choix populaire pour la classification des images parce qu'il est :

- a) **Polyvalent** : le VGG16 peut être utilisé pour une variété de tâches de classification d'images, y compris la détection d'objets, la classification de scènes et l'identification de personnes.
- b) **Précision** : le VGG16 s'est avéré très précis en matière de classification d'images obtenant des résultats de pointe sur un certain nombre d'ensembles de données de référence.
- c) **Efficace** : VGG16 est relativement efficace, ce qui en fait un bon choix pour les applications en temps réel.

Voici quelques-unes des raisons pour lesquelles nous utilisons le VGG16 :

- a) **Précision** : le VGG16 s'est avéré très précis en matière de classification d'images atteignant des résultats de pointe sur un certain nombre de données de référence par exemple, sur l'ensemble de données ImageNet, le VGG16 atteint une précision de 92,7 %.
- b) **Apprentissage par transfert** : Le VGG16 peut être utilisé pour l'apprentissage par transfert qui est une technique permettant d'utiliser un modèle pré-entraîné pour améliorer les performances d'un nouveau modèle. L'apprentissage par transfert est souvent utilisé lorsque nous disposons d'un petit ensemble de données pour notre nouveau modèle.
- c) **Facilité d'utilisation** : VGG16 est disponible dans de nombreux frameworks d'apprentissage profond, tels que Keras et TensorFlow. Il est donc facile à utiliser pour les tâches de classification d'images.

Dans l'ensemble, le VGG16 est un CNN puissant et polyvalent, largement utilisé pour la classification d'images. Il est précis, efficace et facile à utiliser.

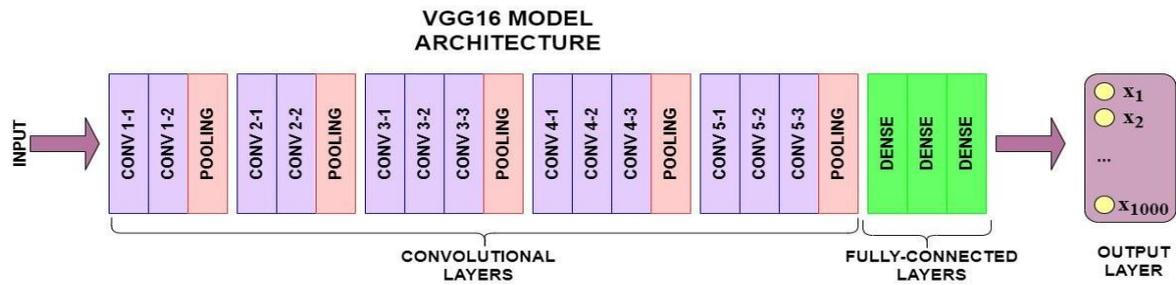


Figure II.16. L'architecture de VGG16

6.1. Apprentissage par transfert et VGG16

L'apprentissage par transfert est une technique d'apprentissage automatique qui consiste à réutiliser un modèle formé pour une tâche donnée comme point de départ d'un modèle pour une deuxième tâche. Cela peut s'avérer utile lorsque les données disponibles pour la seconde tâche sont limitées ou lorsque les deux tâches sont liées d'une manière ou d'une autre.

VGG16 est un réseau neuronal convolutif (CNN) qui a été entraîné sur l'ensemble de données ImageNet, qui contient plus de 14 millions d'images et 1000 classes différentes. Le VGG16 est un modèle très puissant, capable d'obtenir des résultats de pointe dans diverses tâches de classification d'images.

Pour utiliser l'apprentissage par transfert avec VGG16, nous pouvons commencer par charger le modèle pré-entraîné à partir d'une bibliothèque Keras. Nous pouvons ensuite geler les poids des premières couches du modèle et entraîner les dernières couches sur notre propre ensemble de données. Cela nous permet de tirer parti des connaissances que VGG16 a acquises sur l'ensemble de données ImageNet, tout en étant en mesure d'adapter le modèle à notre tâche spécifique.

L'apprentissage par transfert avec le VGG16 peut être un moyen très efficace de construire un modèle d'apprentissage automatique lorsque les données disponibles sont limitées. Il peut également être utilisé pour améliorer les performances d'un modèle lorsque l'on dispose de nombreuses données, mais que celles-ci sont bruyantes ou déséquilibrées.

6.2. Les avantages de l'utilisation de l'apprentissage par transfert avec le VGG16

- Réduit la quantité de données de formation nécessaires. Le VGG16 a été entraîné sur un vaste ensemble de données, de sorte qu'il a déjà une bonne compréhension des caractéristiques générales des images. Cela signifie que nous pouvons entraîner un modèle sur un ensemble de données plus petit, ce qui permet d'économiser du temps et des ressources.

- Améliore les performances du modèle. En affinant les dernières couches de VGG16 sur notre propre ensemble de données, nous pouvons améliorer les performances du modèle pour notre tâche spécifique.
- Est relativement facile à mettre en œuvre. Il existe de nombreux tutoriels qui montrent comment utiliser l'apprentissage par transfert avec le VGG16.

6.3. Les inconvénients de l'utilisation de l'apprentissage par transfert avec le VGG16

- Peut ne pas être efficace si les données sont très différentes de celles sur lesquelles le VGG16 a été entraîné. Dans ce cas, il peut être préférable de former un modèle à partir de zéro.
- Peut ne pas être en mesure d'atteindre les résultats les plus récents. L'apprentissage par transfert est une technique puissante, mais il se peut qu'elle ne permette pas d'obtenir des résultats de pointe pour toutes les tâches. Si vous souhaitez obtenir les meilleurs résultats possibles, vous devrez peut-être former un modèle à partir de zéro.

Dans l'ensemble, l'apprentissage par transfert avec VGG16 est une technique puissante qui peut être utilisée pour construire des modèles d'apprentissage profond lorsque les données disponibles sont limitées. Elle peut également être utilisée pour améliorer les performances d'un modèle lorsqu'il y a beaucoup de données disponibles, mais que celles-ci sont bruyantes ou déséquilibrées.

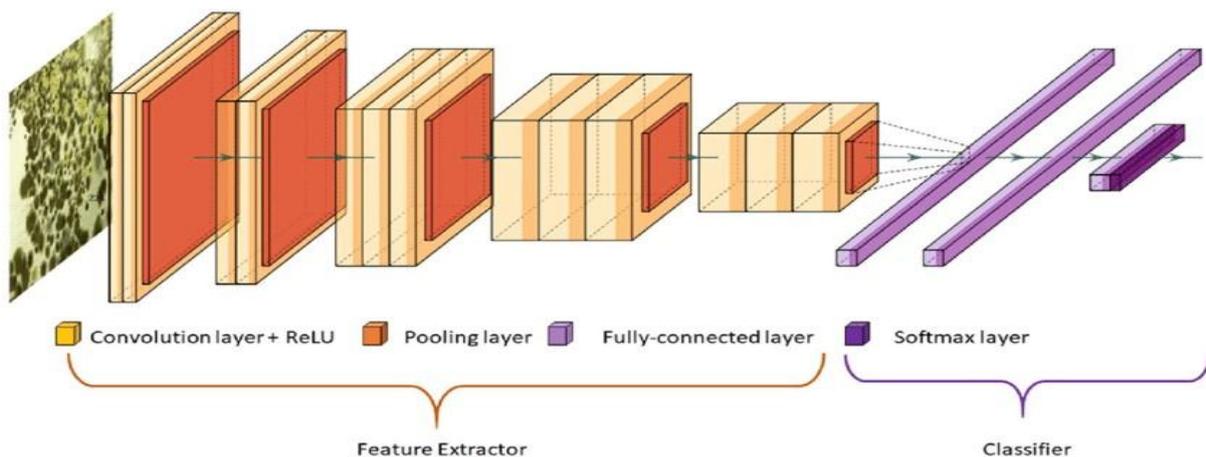


Figure II.17. L'architecture de VGG16

6.4. VGG16 et ses limites

Le VGG16 présente un certain nombre de limitations, notamment :

- a) **Complexité de calcul** : VGG16 est un modèle très coûteux en termes de calcul à former et à déployer. Il comporte un grand nombre de paramètres, ce qui nécessite beaucoup de données et de calculs pour le former. Cela peut rendre difficile l'utilisation du VGG16 sur des appareils aux ressources limitées, tels que les téléphones mobiles ou les systèmes embarqués.
- b) **Gradients de fuite** : Le VGG16 est sujet au problème du gradient de fuite, ce qui peut rendre difficile l'entraînement du modèle jusqu'à la convergence. En effet, le modèle est doté d'une architecture très profonde, ce qui peut rendre les gradients très faibles à mesure qu'ils se propagent dans le réseau.
- c) **Ajustement excessif** : Le VGG16 a tendance à être sur ajusté, ce qui peut entraîner des performances médiocres sur des données inédites. En effet, le modèle comporte un grand nombre de paramètres, ce qui peut rendre difficile sa généralisation à de nouvelles données.
- d) **Données requises** : Le VGG16 nécessite une grande quantité de données d'entraînement pour obtenir de bonnes performances. Cela peut représenter un défi, en particulier pour des tâches telles que la détection et la segmentation d'objets, qui nécessitent une grande variété de classes d'objets.

Malgré ces limitations, le VGG16 reste un modèle populaire pour les tâches de classification d'images. Il a obtenu des résultats de pointe sur un certain nombre d'ensembles de données de référence et est souvent utilisé comme référence pour d'autres modèles de classification d'images. Cependant, il est important d'être conscient des limites du VGG16 lorsqu'on l'utilise pour des applications réelles.

6.5. Moyens de surmonter les limites de la VGG16

- a) **Augmentation des données** : L'augmentation des données peut être utilisée pour accroître la taille de l'ensemble de données d'apprentissage et réduire l'ajustement excessif. Pour ce faire, on peut créer de nouveaux exemples de formation en appliquant des transformations aux données existantes, telles que le recadrage, le retournement et la rotation des images.

réseau neuronal pour prédire les boîtes englobantes et les probabilités de classe simultanément pour toutes les régions de l'image.

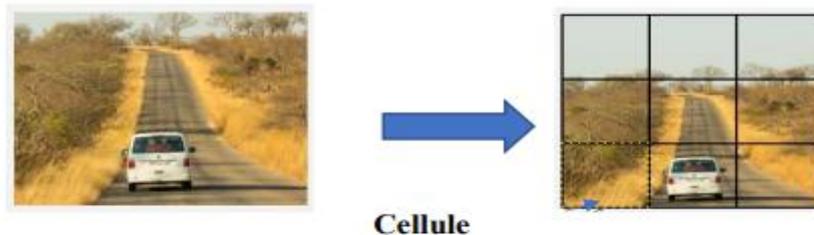
- Grille de prédiction : YOLO divise l'image en une grille de cellules. Pour chaque cellule de cette grille, le modèle prédit un certain nombre de boîtes englobantes ainsi que les probabilités de classe correspondantes.
- Régression directe : Contrairement à d'autres approches qui utilisent des régions de propositions, YOLO effectue une régression directe des coordonnées des boîtes englobantes, ce qui simplifie le processus de détection d'objets.
- Non max suppression : Après la prédiction des boîtes englobantes et des probabilités de classe, YOLO utilise une technique appelée "non-maximum suppression" pour éliminer les détections redondantes et fusionner les boîtes englobantes qui se chevauchent.

Modèles : YOLO a plusieurs variantes, notamment YOLOv1, YOLOv2, ..., YOLOv9, chacune introduisant des améliorations en termes de précision, de vitesse ou de capacités.

7.1. L'algorithme de modèle YOLO

Voici une description générale de l'algorithme du modèle YOLO ^[13] :

- Entrée de l'image : L'algorithme YOLO prend une image en entrée. Cette image est ensuite divisée en une grille de cellules.



FigureII.19. Diviser l'image

- Prédiction des boîtes englobantes et des probabilités de classe : Pour chaque cellule de la grille, le modèle prédit un certain nombre de boîtes englobantes (bounding boxes). Chaque boîte englobante est représentée par des coordonnées (x, y) relatives à la cellule et des largeurs et hauteurs (w, h) relatives à l'image entière. En plus des boîtes englobantes, le modèle prédit également les probabilités de classe pour chaque boîte, indiquant la probabilité qu'un objet de cette classe soit présent dans la boîte.

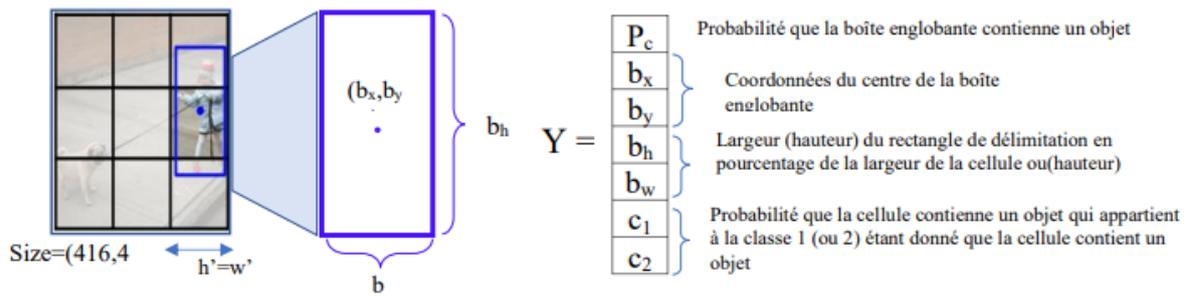


Figure II.20. Le vecteur prédit dans le cas d'une seule boîte

- Suppression non maximale : Après la prédiction des boîtes englobantes et des probabilités de classe, une technique appelée "non-maximum suppression" est utilisée pour éliminer les détections redondantes et fusionner les boîtes englobantes qui se chevauchent.

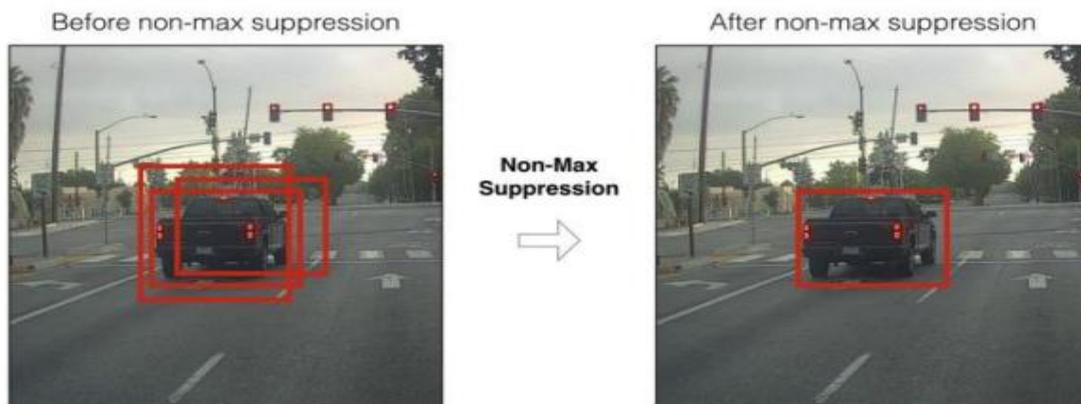


Figure II.21. Le résultat de suppression non maximale.

- Seuils de confiance et suppression des boîtes : les boîtes englobantes avec des probabilités de classe inférieures à un seuil de confiance prédéfini sont supprimées, ce qui permet de filtrer les détections moins fiables.
- Sortie de détection : la sortie de l'algorithme YOLO est une liste des boîtes englobantes détectées, chacune associée à une classe prédite et à une probabilité de confiance. Ces informations peuvent ensuite être utilisées pour visualiser et analyser les objets détectés dans l'image.



Figure II.22. Sortie de détection.

L'avantage principal de YOLO est sa capacité à effectuer la détection d'objets en temps réel, grâce à son architecture optimisée qui prédit les boîtes englobantes et les probabilités de classe simultanément pour toutes les régions de l'image. Cela rend YOLO particulièrement adapté à des applications telles que la surveillance vidéo, la détection de piétons pour les véhicules autonomes, la sécurité et bien d'autres.

Conclusion

Au cours de ce chapitre, nous avons présenté les réseaux neuronaux convolutifs. Ils sont en mesure d'extraire des caractéristiques provenant d'images d'entrée et de les classifier.

Par ailleurs, nous avons expliqué le modèle VGG16, qui est un réseau neuronal convolutif entraîné sur l'ensemble de données ImageNet, qui contient plus de 14 millions d'images et 1 000 classes différentes. Le modèle VGG16 est un modèle très puissant capable d'obtenir des résultats optimaux dans diverses tâches de classification d'images. Nous avons ensuite expliqué l'importance du modèle YOLO et son fonctionnement.

Enfin, ce chapitre donne un aperçu des avancées du domaine de l'intelligence artificielle et de l'apprentissage profond (deep learning).

CHAPITER 03

Classification et localisation des défauts de tôles

Introduction

Dans ce chapitre, nous avons utilisé le langage de programmation Python et plusieurs bibliothèques spécialisées telles que TensorFlow et Keras pour concevoir et évaluer des modèles de type réseau de neurones convolutifs (CNN) afin de contrôler la qualité des tôles après le laminage. En plus de l'entraînement et de l'évaluation des modèles, nous avons utilisé des méthodes de visualisation telles que GradCAM pour appréhender les choix effectués par nos modèles. Deux types de modèles sont présents, un CNN simple et le modèle VGG16 ont été considérés, chacun étant évalué pour sa capacité à identifier les défauts avec précision. Enfin, on a étudié comment le modèle YOLO (You Only Look Once) peut servir à détecter automatiquement les défauts, ce qui constitue une approche alternative et efficace pour améliorer le processus de contrôle de la qualité des feuilles métalliques.

1. Python

Python est un langage de programmation polyvalent et convivial, apprécié pour sa simplicité syntaxique et sa lisibilité. Conçu pour favoriser une programmation claire et concise, Python est largement utilisé dans divers domaines, de la science des données à la création de sites web dynamiques. Sa grande bibliothèque standard offre une multitude de fonctionnalités prêtes à l'emploi, facilitant le développement d'applications complexes avec un minimum d'effort. La popularité croissante de Python est également attribuable à sa communauté dynamique et collaborative, qui crée constamment de nouveaux modules, packages et frameworks pour répondre aux besoins changeants des développeurs du monde entier. En somme, Python est un outil polyvalent et puissant qui continue d'attirer un large éventail d'utilisateurs grâce à sa simplicité, sa flexibilité et sa richesse fonctionnelle.

2. Bibliothèques utilisées

- **TensorFlow** : Nous avons employé cette librairie pour définir les éléments fondamentaux de l'architecture CNN. Cette bibliothèque est destinée pour l'implémentation des algorithmes d'apprentissage profond, elle offre aussi une grande flexibilité dans le cadre de l'utilisation pour le développement d'un réseau des neurones.
- **Keras** : permet les bibliothèques utilisées avec TensorFlow est Keras, nous avons utilisé cette bibliothèque pour implémenter les couches les différentes couches, les fonctions d'activation et la préparation de la base d'apprentissage.
- **Scikit-learn** : est l'une des bibliothèques les plus utiles pour l'apprentissage automatique en Python. La bibliothèque scikit-learn contient de nombreux outils efficaces pour

l'apprentissage automatique et la modélisation statistique, notamment la classification, la régression, le clustering et la réduction de la dimensionnalité.

- **NumPy** : Nous avons utilisé cette bibliothèque pour adapter les types d'entrée selon la configuration du modèles utilisés, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux. Nous avons utilisé cette bibliothèque exactement dans le cas de balayage de l'image et l'extraction des fenêtres.
- **Matplotlib** : est une bibliothèque du langage de programmation Python destinée à tracer et visualiser des données sous formes de graphiques, nous avons utilisé cette bibliothèque pour visualiser note images sous formes de graphiques.
- **Pandas** : est un outil d'analyse et de manipulation de données open source rapide, puissant, flexible et facile à utiliser, construit sur le langage de programmation Python.

3. Configuration Utilisé dans l'implémentation

La configuration du matériel utilisé dans notre implémentation est :

CPU	i7-6700HQ (2.6 GHz)
GPU	Nvidia GTX 960M 4Gb
RAM	16 Gb
OS	Windows 10 Famille

Figure III.1. Caractéristiques techniques du matériel utilisé dans les calculs.

4. Base de données d'images

Ces images proviennent de la base de données des défauts de surface de l'université du Nord-Est (NEU), qui rassemble 7 types de défauts de surface typiques des tôles d'acier laminées à chaud : Fossé roulé (Rolled in scale), taches (Patches), craquelures (Crazing), surface piquée (Pitted surface), inclusions (Inclusion) et rayures (Scratches) et ligne de soudure (Welding line). La base de données comprend 2073. Dans cette base on trouve 1905 images pour l'entraînement et 84 images pour le test et 84 images pour validation ^[4].

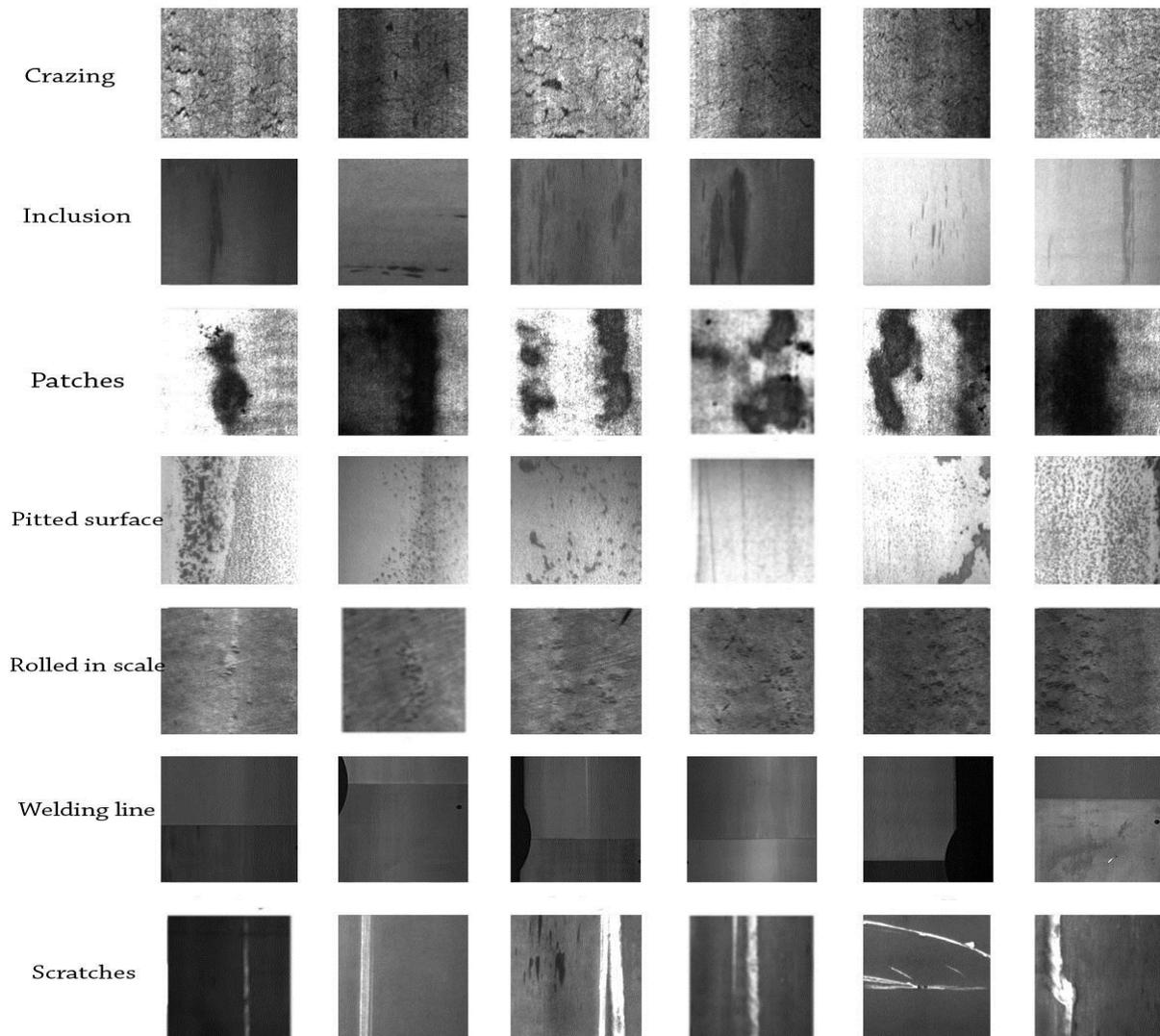


Figure III.2. Défauts de surface typiques des tôles d'acier [4].

5. Visualiser les prédictions avec GradCAM

GradCAM (Gradient-weighted Class Activation Mapping) est une technique d'interprétabilité des réseaux neuronaux convolutifs (CNN) utilisée pour comprendre quelles parties de l'image contribuent le plus à la prédiction d'une classe spécifique par le modèle. Cette technique permet de visualiser les régions d'intérêt de l'image qui ont été activées par le réseau pour effectuer une prédiction [15].

Voici comment fonctionne GradCAM :

- Calcul des gradients : Tout d'abord, les gradients de la sortie de la couche de sortie du réseau par rapport à l'activation de la dernière couche convolutive sont calculés. Ces gradients indiquent comment chaque activation de la dernière couche convolutive affecte la prédiction de la classe cible.

- **Pondération des gradients** : Les gradients sont ensuite pondérés en fonction de leur importance pour la classe cible. Cela se fait en prenant la moyenne des gradients pour chaque carte d'activation et en utilisant ces valeurs pour pondérer les activations de la dernière couche convolutive.
- **Combinaison des cartes d'activation** : Les activations de la dernière couche convolutive sont combinées en une carte d'activation, qui met en évidence les régions de l'image qui ont été les plus influentes pour la prédiction de la classe cible.
- **Visualisation** : La carte d'activation est superposée à l'image d'origine pour visualiser les régions d'intérêt qui ont conduit à la prédiction de la classe cible par le modèle.

6. Architectures des modèles

Au cours de nos tentatives, nous avons créé deux modèles (un simple CNN et le modèle VGG16) dotés d'architectures différentes.

Dans ce qui suit, nous présentons en détail l'architecture de ces deux modèles :

6.1. Le réseau de neurones convolutifs simple (CNN)

Le premier modèle que nous présentons dans la figure III.3 est composé de deux couches de convolution et deux couches de max pooling et deux couches entièrement connectées.

De taille 200*200, l'image d'entrée passe d'abord par la première couche de convolution. Cette couche est composée de 32 filtres de taille 3*3, chacune de nos couches de convolution est suivie d'une fonction d'activation ReLU, cette fonction force les neurones à renvoyer des valeurs positives, après cette convolution 32 "featuremaps" de taille 198*198 seront créées.

Nous appliquons ensuite le max pooling pour réduire la taille de l'image et le nombre de paramètres et de calculs. À la fin de cette couche, nous aurons 32 "featuremaps" de taille 99*99.

Nous procédons de la même manière avec la deuxième couche de convolution, cette couche est composée de 64 filtres, la fonction d'activation ReLU est toujours appliquée à chaque convolution.

De plus, la couche "Max pooling" est appliquée après la deuxième couche de convolution. La sortie de ces couches est constituée de 64 "featuremaps" de taille 48x48. Le vecteur caractéristique résultant des convolutions a une dimension de 147650.

En complément de ces quatre couches de convolution, nous utilisons un réseau de neurones composé de deux couches entièrement connectées. La première couche comporte

Chapitre 03 : Classification et localisation des défauts de tôles

128neurones où la fonction d'activation utilisée est celle de ReLU, et la seconde couche est une softmax qui calcule la distribution de probabilité des 7 classes (nombre de classes dans la base de données d'images NEU).

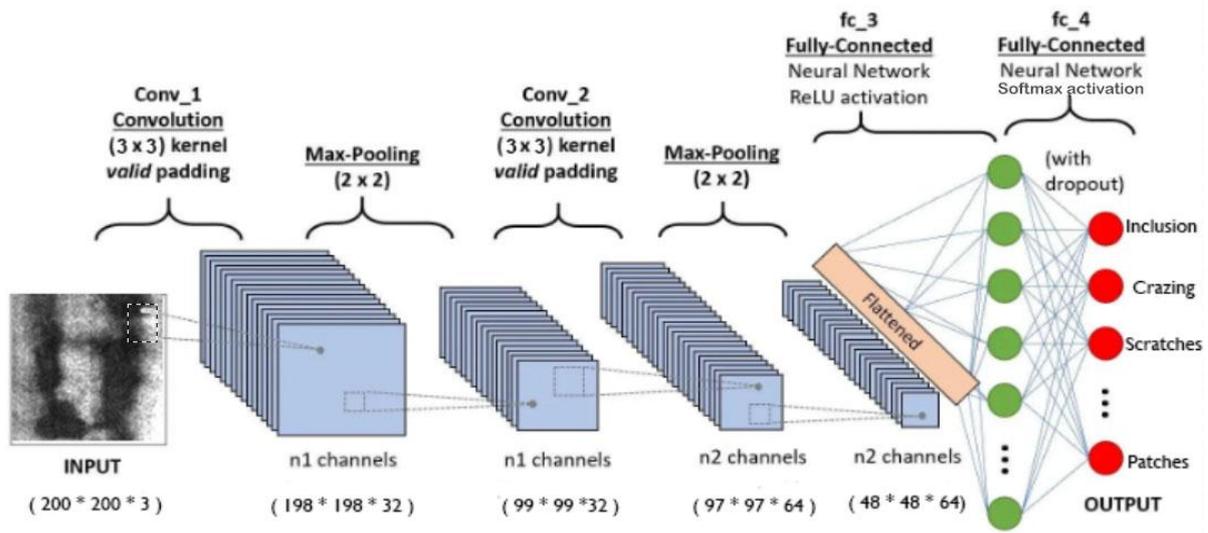


Figure III.3. Architecture du modèle 1 (CNN simple).

Tableau III.1. Configuration du modèle 1.

Layer (type)	Output Shape	Param #
conv2d_95 (Conv2D)	(None, 198, 198, 32)	896
max_pooling2d_94 (MaxPooling2D)	(None, 99, 99, 32)	0
conv2d_96 (Conv2D)	(None, 97, 97, 64)	18496
max_pooling2d_95 (MaxPooling2D)	(None, 48, 48, 64)	0
flatten_44 (Flatten)	(None, 147456)	0
dense_70 (Dense)	(None, 128)	18874496
dropout_31 (Dropout)	(None, 128)	0
dense_71 (Dense)	(None, 7)	903
=====		
Total params: 18894791 (72.08 MB)		
Trainable params: 18894791 (72.08 MB)		
Non-trainable params: 0 (0.00 Byte)		

6.2. Architecture du modèle VGG16

Le deuxième modèle VGG16 illustré à la figure III.5 est composé de 13 couches de convolution, de 5 couches de "max pooling" et de deux couches entièrement connectées.

L'architecture similaire du VGG16 est basée sur la répétition de blocs empilés. Voici les détails de ces blocs :

Blocs Convolutifs : VGG16 n'utilise pas de blocs complexes avec des couches multiples à l'intérieur. Chaque bloc est simple et constitué de seulement deux couches convolutives.

Taille de Filtre : Les deux couches convolutives à l'intérieur de chaque bloc utilisent une taille de filtre 3x3. Cela signifie que chaque filtre analyse une petite zone de 3x3 pixels dans l'image d'entrée.

Fonction d'Activation ReLU : Après chaque couche convolutive, on applique une fonction d'activation ReLU (Rectified Linear Unit). Cette fonction permet d'introduire de la non-linéarité dans le réseau, ce qui est important pour apprendre des représentations complexes des images.

Max Polling : Un pooling max est ajouté après chaque bloc convolutif. Le pooling max permet de réduire la dimensionnalité des données et d'introduire une certaine invariance à la translation. Il sélectionne la valeur maximale dans une zone de l'image définie par une fenêtre de pooling (souvent de taille 2x2). À la sortie de ces couches, nous aurons 512 featuremaps de taille 6*6. Le vecteur de caractéristiques issu des convolutions a une dimension de 18432.

Après ces quatre couches de convolution, nous utilisons un réseau de neurones de deux couches fully-connected. La première couche a 512 neurones où la fonction d'activation utilisée est le ReLU, et la deuxième couche est un softmax qui permet de calculer la distribution de probabilité des 7 classes (nombre de classe dans la base d'image NEU).

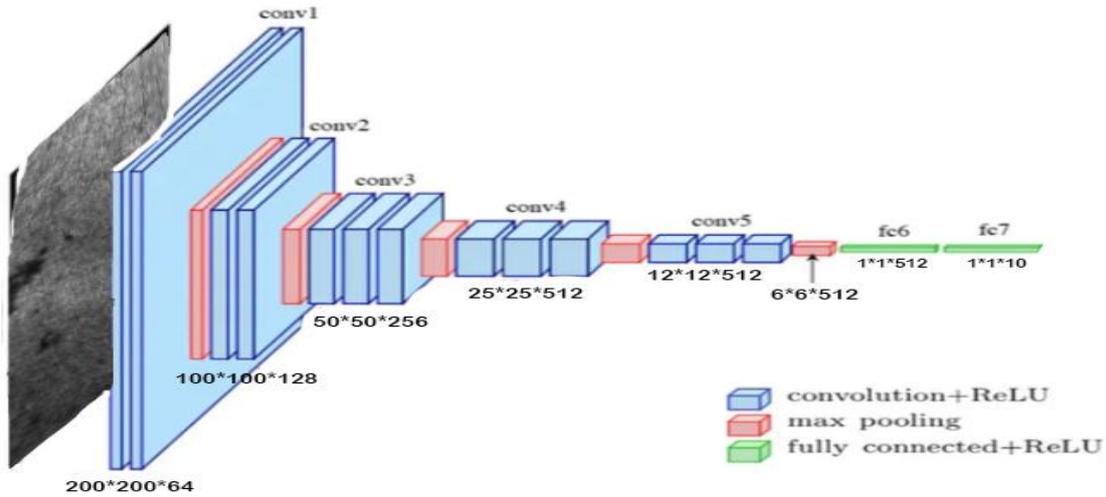


Figure III.4. Architecture du modèle VGG16.

Tableau III.2. Configuration du modèle 2 (VGG16).

```

Model: "model"
-----
Layer (type)                Output Shape                Param #
-----
input_1 (InputLayer)        [(None, 200, 200, 3)]      0
block1_conv1 (Conv2D)       (None, 200, 200, 64)       1792
block1_conv2 (Conv2D)       (None, 200, 200, 64)       36928
block1_pool (MaxPooling2D)  (None, 100, 100, 64)       0
block2_conv1 (Conv2D)       (None, 100, 100, 128)      73856
block2_conv2 (Conv2D)       (None, 100, 100, 128)      147584
block2_pool (MaxPooling2D)  (None, 50, 50, 128)        0
block3_conv1 (Conv2D)       (None, 50, 50, 256)        295168
block3_conv2 (Conv2D)       (None, 50, 50, 256)        590080
block3_conv3 (Conv2D)       (None, 50, 50, 256)        590080
block3_pool (MaxPooling2D)  (None, 25, 25, 256)        0
block4_conv1 (Conv2D)       (None, 25, 25, 512)        1180160
block4_conv2 (Conv2D)       (None, 25, 25, 512)        2359808
block4_conv3 (Conv2D)       (None, 25, 25, 512)        2359808
block4_pool (MaxPooling2D)  (None, 12, 12, 512)        0
block5_conv1 (Conv2D)       (None, 12, 12, 512)        2359808
block5_conv2 (Conv2D)       (None, 12, 12, 512)        2359808
block5_conv3 (Conv2D)       (None, 12, 12, 512)        2359808
block5_pool (MaxPooling2D)  (None, 6, 6, 512)          0
flatten (Flatten)           (None, 18432)               0
dense (Dense)                (None, 512)                 9437696
dense_1 (Dense)              (None, 7)                   3591
-----
Total params: 24155975 (92.15 MB)
Trainable params: 9441287 (36.02 MB)
Non-trainable params: 14714688 (56.13 MB)
  
```

7. Résultats obtenus et discussion

En vue de présenter les résultats obtenus des deux modèles, nous illustrons dans ce qui suit les résultats en termes de précision et d'erreur tout comme la matrice de confusion pour chacun des deux modèles.

A. Résultats du modèle 1

Il ressort de la figure III.5 que la précision de l'apprentissage de l'entraînement et de la validation augmente avec le nombre d'itérations, ce qui reflète le fait que le modèle apprend plus d'informations à chaque époque. Si la précision est réduite, nous aurons besoin de plus d'informations pour que notre modèle apprenne, et nous devons donc augmenter le nombre d'itérations, et vice versa. De même, la perte de l'entraînement et de validation diminue avec le nombre d'époques. Nous notons également que nous avons fixé le nombre d'époques (20 itérations) et le nombre de batches (32). En ce qui concerne l'apprentissage, la perte de classification est de 0,40 et la précision de la classification est de 86,40 %, tandis qu'en validation, la perte est de 0,27 et la précision est de 92,86 %.

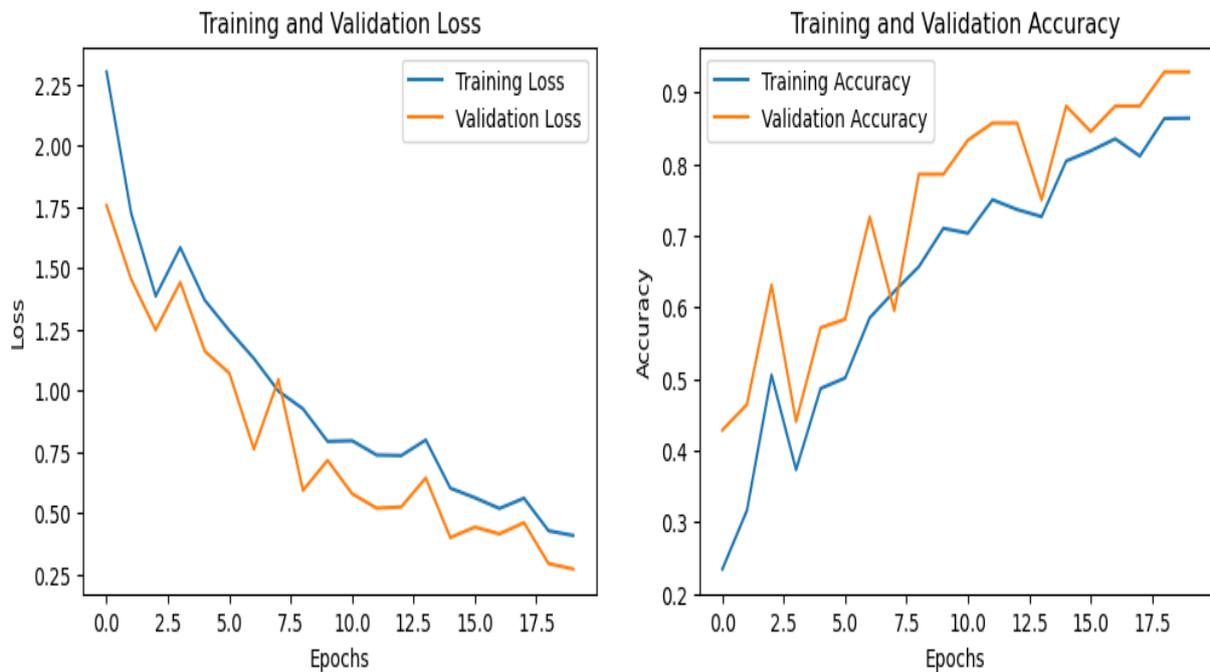


Figure III.5. Courbes de précision et de perte associées au modèle 01.

On voit sur la figure III.6 que le nombre total d'images mal classées est de 6, pour un taux d'erreur de 7,1%, alors que le nombre total d'images correctement classées est de 78, donnant un taux de précision de 92,9%.

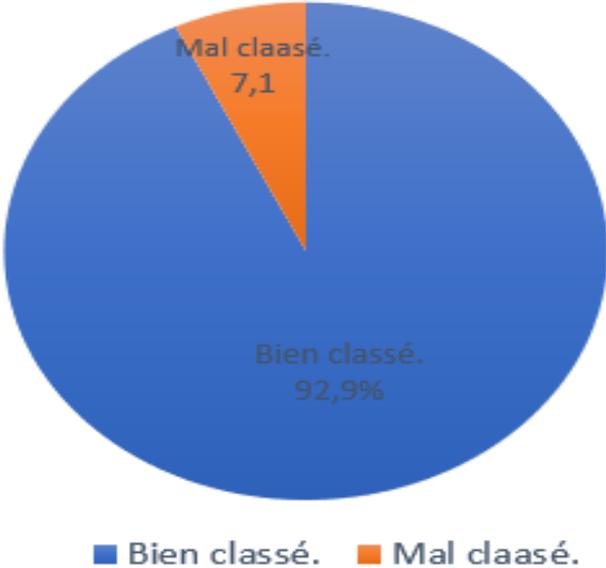


Figure III.6. Taux d'erreur du modèle 01.

Pour évaluer les performances de notre modèle, la matrice de confusion reflète les métriques Vrai Positif, Vrai Négatif, Faux Positif et Faux Négatif. On peut voir sur la figure III.7 une représentation de la position de ces métriques pour chaque classe. On constate ainsi que le modèle a correctement classifié la plupart des images, mais qu'il a légèrement mal classifié certaines images.

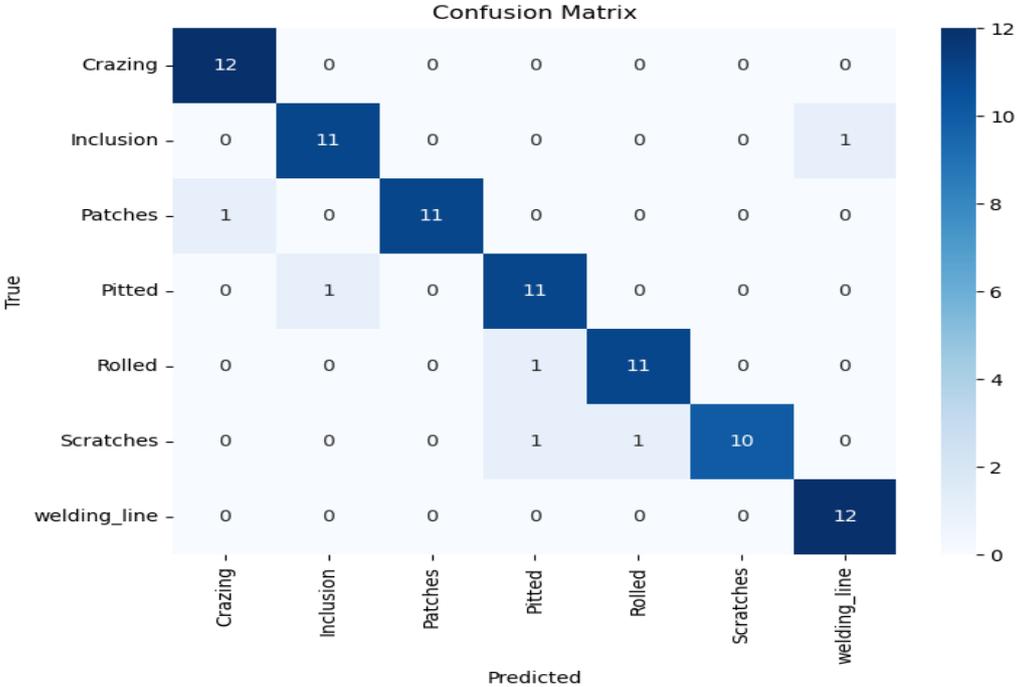


Figure III.7. Matrice de confusion pour le modèle 01.

Chapitre 03 : Classification et localisation des défauts de tôles

Sur la figure III.8, on peut voir un échantillon d'images avec leur classe prédite et la précision de validation, à côté de la vraie classe.

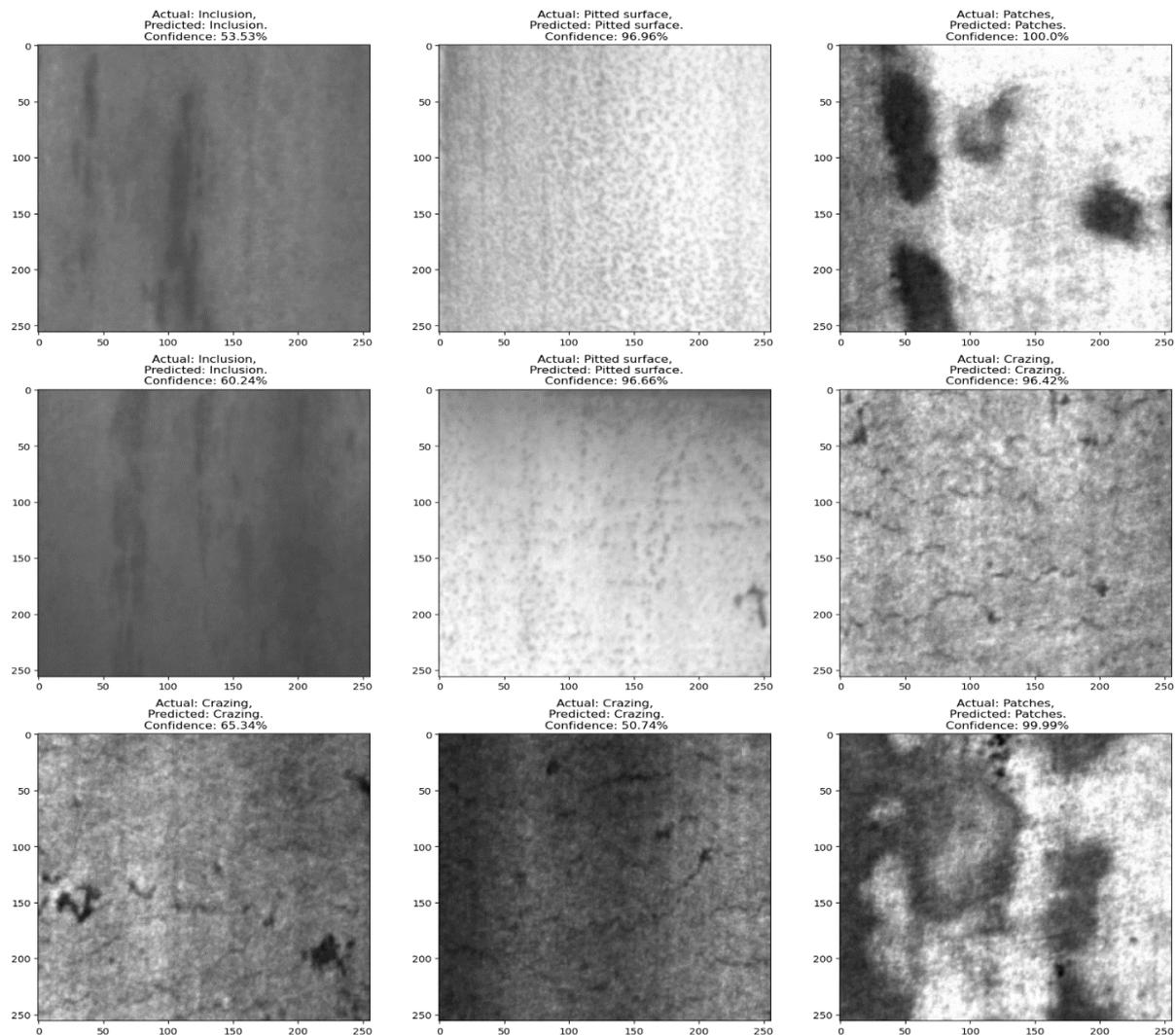


Figure III.8. Extrait des images prédites avec leurs labels respectifs du modèle 01.

A la figure III.9, on voit l'application de l'algorithme GradCam à quelques images échantillons ainsi que les cartes thermiques, qui sont des images distinctes de l'image originale et de GradCam, mais qui contiennent les mêmes informations.

Les parties plus claires indiquent un degré d'importance accru (le plus important pour la classification), le bleu étant le moins important et le rouge le plus important.

Le modèle GradCam révèle que le premier modèle met en évidence les parties les plus importantes de l'image (taches, inclusions, etc.), sans toutefois être précis, comme en témoigne la troisième image, qui ne fait pas ressortir les zones importantes de manière très satisfaisante. Néanmoins, dans l'ensemble, les résultats sont bons.

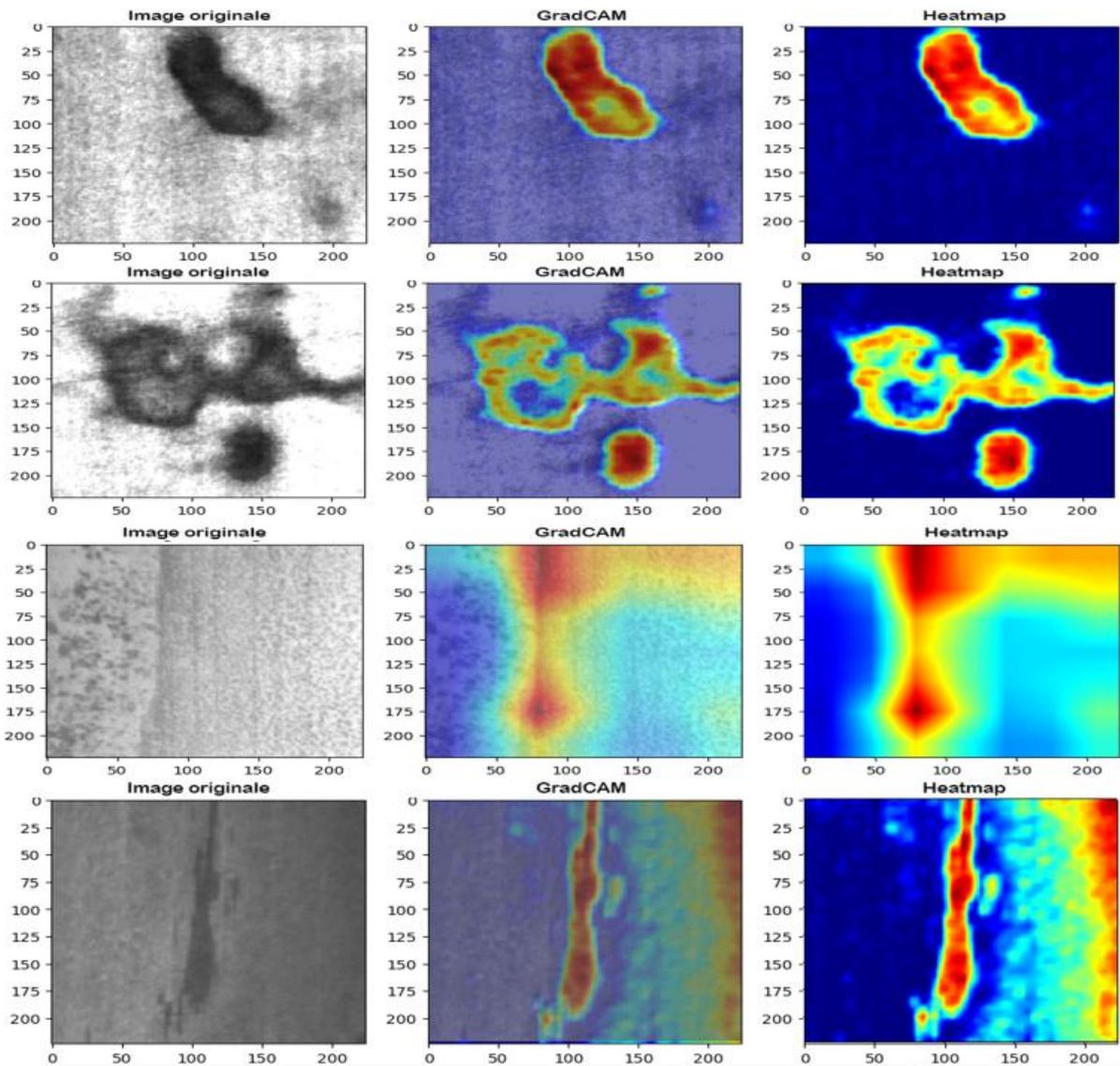


Figure III.9. Visualisation GradCAM du modèle 01.

B. Résultats du modèle 2

En comparaison avec le modèle 1 (CNN simple), la précision de l'entraînement et de validation a augmenté plus rapidement avec le nombre d'itérations, ce qui nous a donné un bon résultat en termes de précision, reflétant le fait qu'à chaque époque, le modèle acquiert des informations plus précises. Quant à la perte d'apprentissage, elle diminue assez rapidement par rapport au modèle 1.

Il convient également de noter que nous avons fixé le nombre d'itérations à 20 et le nombre de batches à 32, avec un dropout de 0,2. Il a fallu 3 heures pour l'apprentissage.

Chapitre 03 : Classification et localisation des défauts de tôles

Le résultat de l'entraînement affiche une perte de 0,0146 pour une précision de 100 %. En revanche, à l'issue de la validation, la perte est de 0,00048 et la précision est de 100 %.



Figure III.10. Précision et perte du modèle 02.

En effet, la figure III.11 fait apparaître que le nombre total d'images bien classées est de 84, avec un taux de précision de 100 % et un taux d'erreur de 0 %.

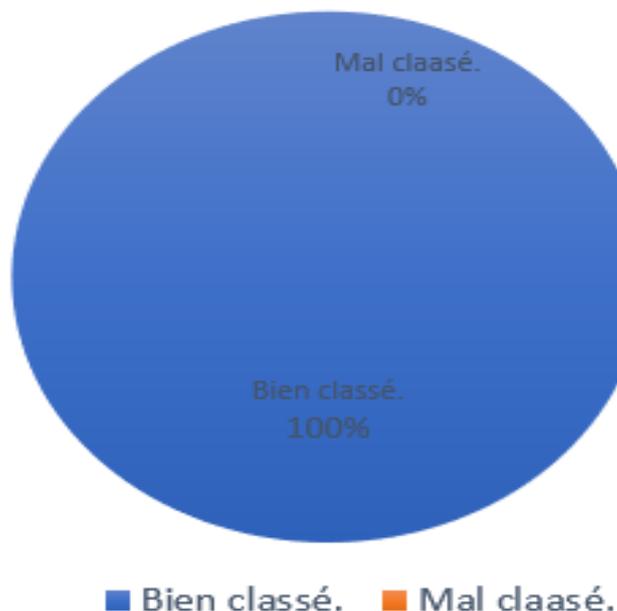


Figure III.11. Taux d'erreur du modèle 02.

Chapitre 03 : Classification et localisation des défauts de tôles

On peut voir sur la figure III.12 une représentation de la position de ces métriques pour chaque classe. On constate ainsi que le modèle a correctement classifié toutes les images.

Ce résultat indique la fiabilité et la précision du modèle.

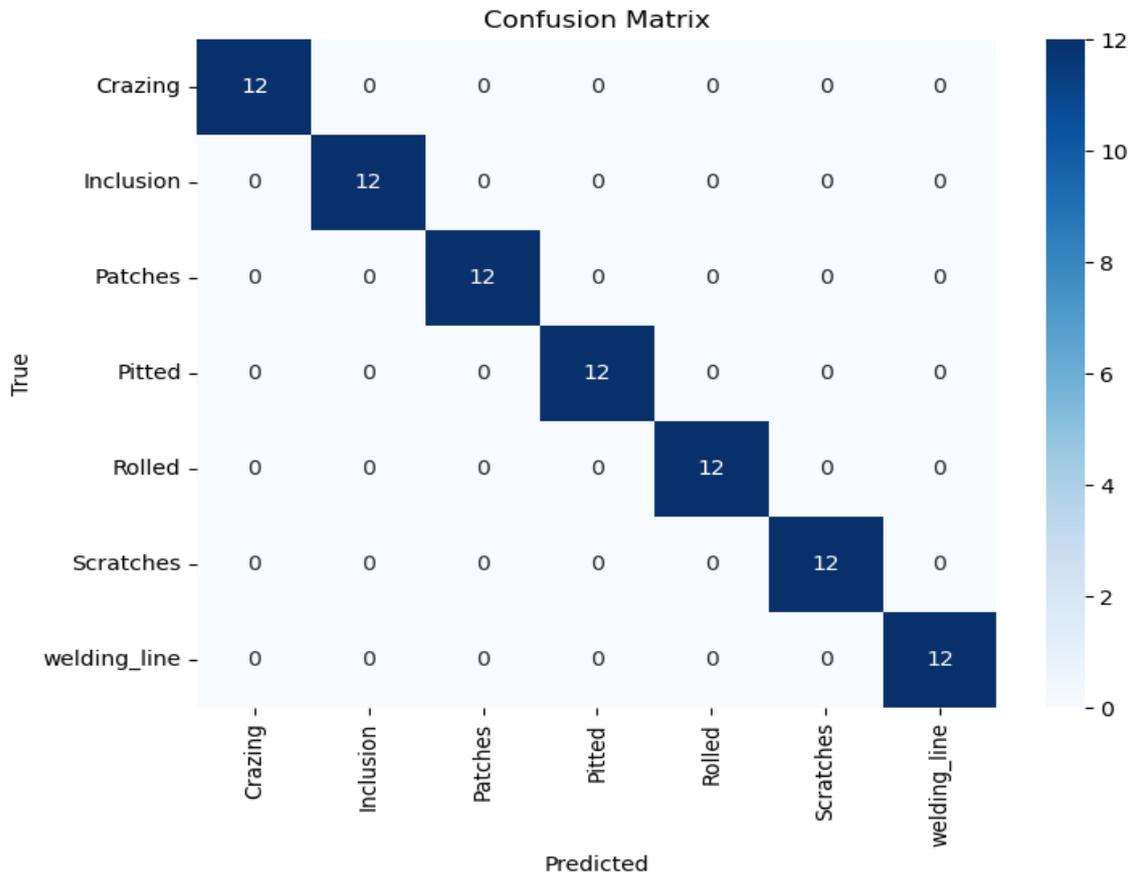


Figure III.12. Matrice de confusion pour le modèle 02.

Au terme des tests, le modèle a pu classer les images avec une précision de 100 %. Il est possible de voir sur la figure III.13 un échantillon d'images avec la classe prédite et la précision de validation, ainsi que sa vraie classe.

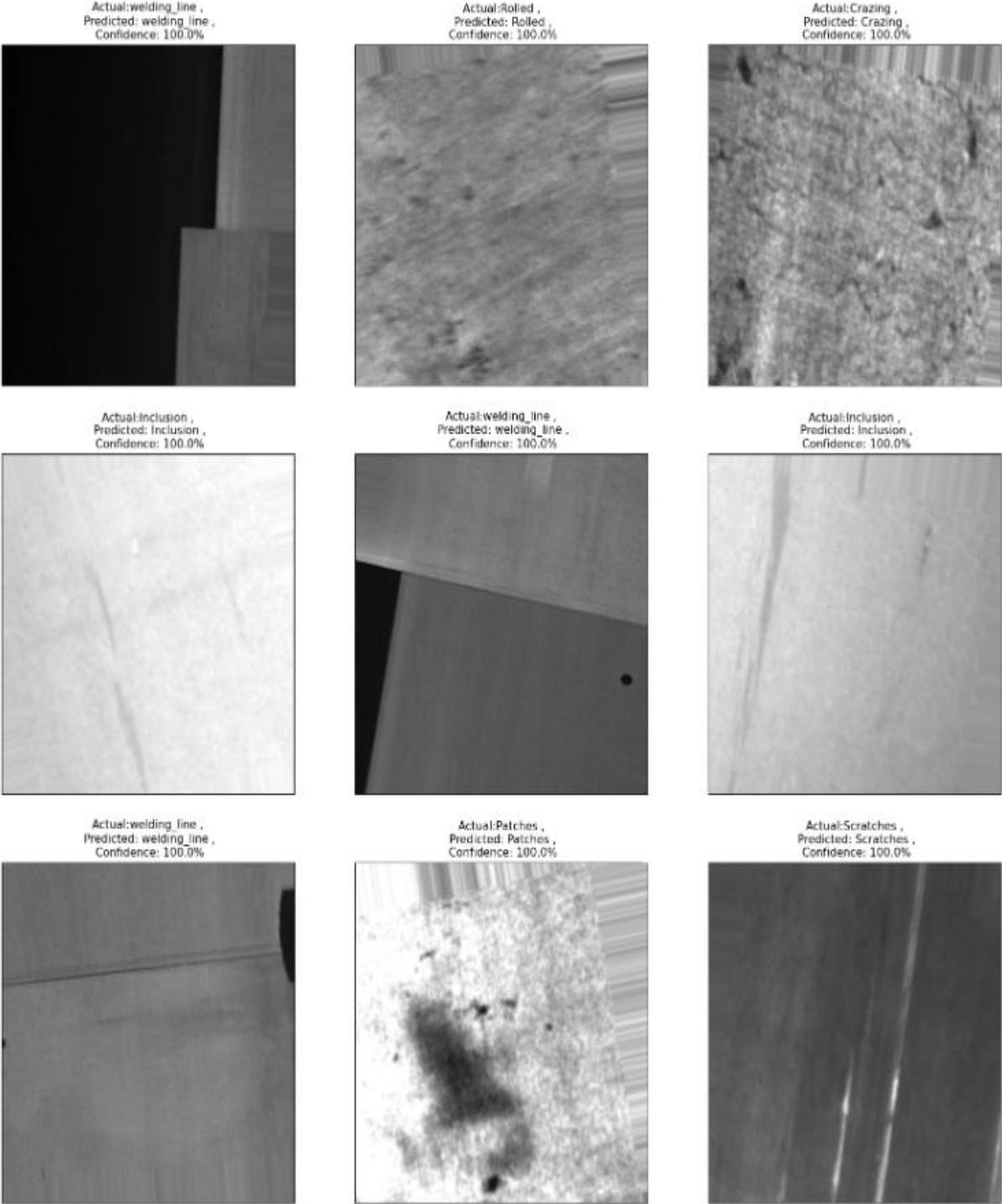


Figure III.13. Extrait des images prédites avec leurs labels respectifs du modèle 02.

Les résultats de l'utilisation de GradCam sur le second modèle sont excellents, contrastant avec le modèle précédent (Figure III.14). Elle permet de visualiser les régions d'intérêt qui ont permis au modèle de prédire le groupe cible avec une grande précision.

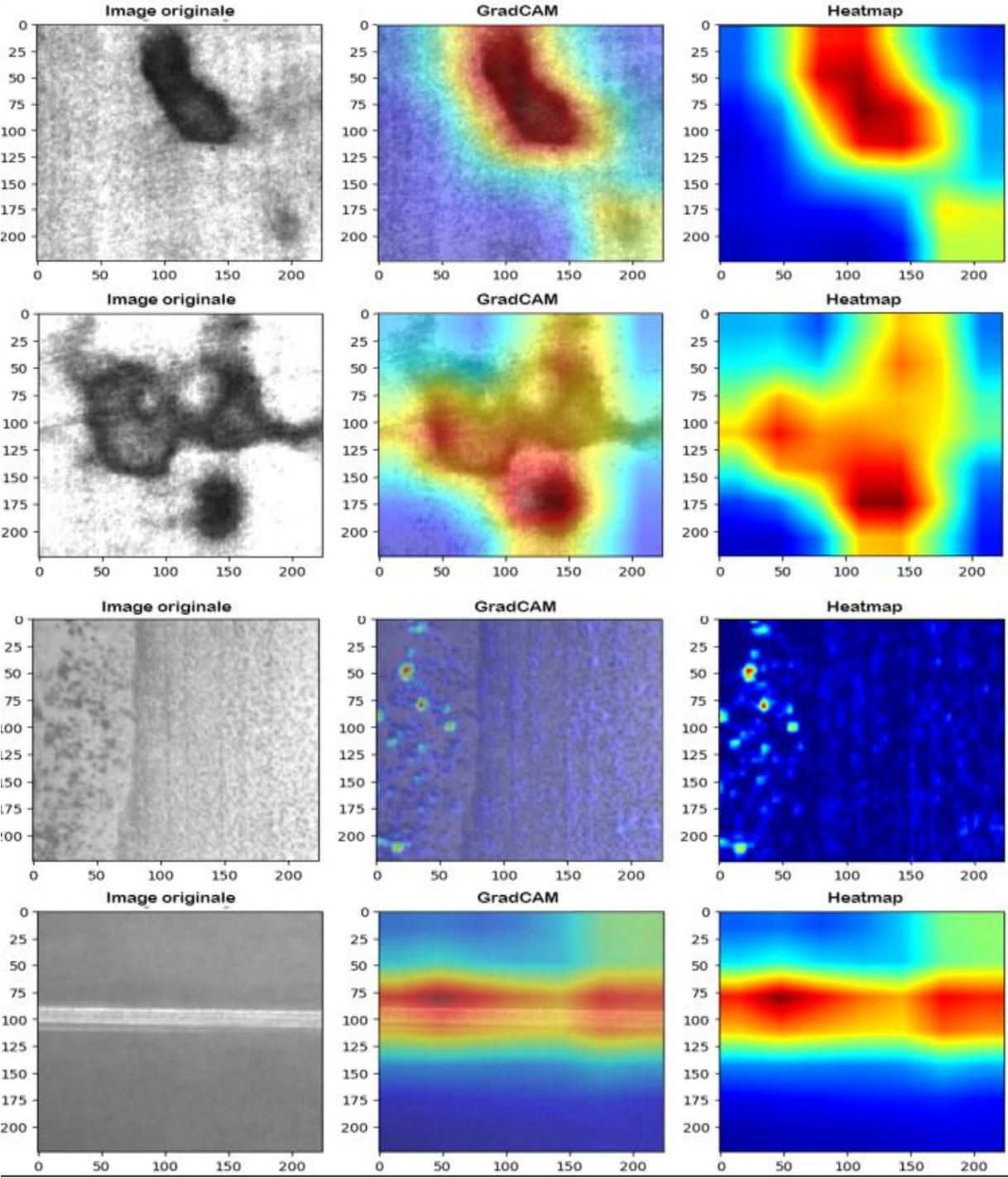


Figure III.14. Illustration GradCAM associée au modèle 02.

8. Comparaison des résultats

On peut constater dans le tableau ci-dessous les différents résultats issus des deux modèles :

Tableau III.3. Résultats comparés entre modèle 01 et modèle 02.

	Architecture Utilisé			Nombre d'itération	La précision d'entraînement	La précision de test	Erreur	Temps d'exécution
	Couche de convolution	Couche de pooling	Fully-connecté					
Modèle 01	2	2	2	20	86.40 %	92.9%	7.1%	1200 seconds (20 min)
Modèle 02	13	5	2	20	100 %	100 %	0 %	10800 seconds (3 heures)

On trouvera dans le tableau ci-dessus des informations sur l'architecture adoptée pour chaque modèle et son nombre d'époques. En outre, les résultats obtenus sont exprimés en termes de précision d'entraînement, de test et d'erreur, sans oublier le temps de calcul. On constate que le modèle 2 fournit les meilleurs résultats. Cette bonne performance s'explique par le nombre de couches de convolution, même si le temps de calcul reste relativement onéreux.

Le plus souvent, un réseau de neurones convolutifs de type VGG16 donne de bons résultats, tout en sachant que les performances de notre modèle sont altérées si on lui enlève une couche convolutive.

La profondeur de notre modèle VGG16 a permis d'améliorer la qualité des résultats obtenus. De même, la taille de la base d'entraînement est un facteur déterminant pour ce type de modèle ; celle-ci doit être suffisamment importante si on veut obtenir des résultats optimaux.

9. Application du modèle YOLO

Dans cette partie, nous allons tester/valider les performances du modèle yolov9 entraîné sur les 7 sous-classes retenues.

Nous pouvons apercevoir certains des résultats obtenus en consultant la figure suivante :

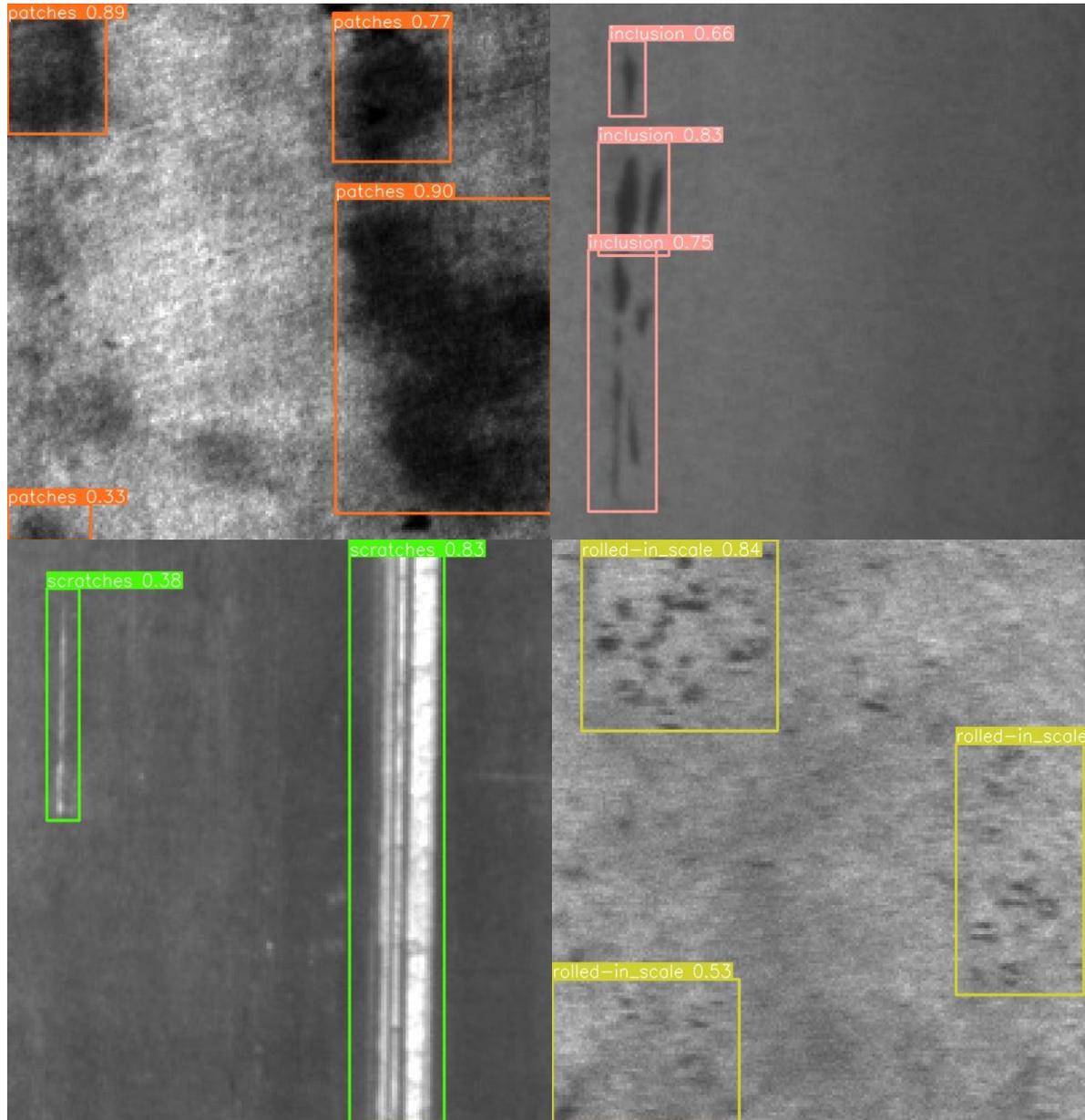


Figure III.15. Résultats de Détection par Yolov9.

Sur la base des résultats obtenus à la figure III.15, il apparaît que le modèle YOLO est plus précis dans la détection des différents objets de l'image, en particulier pour les classes dont le mAP (précision moyenne) est supérieur à 65 (plus ce dernier est élevé, plus les

performances du modèle en termes de précision et de récurrence dans la détection des objets sont bonnes).

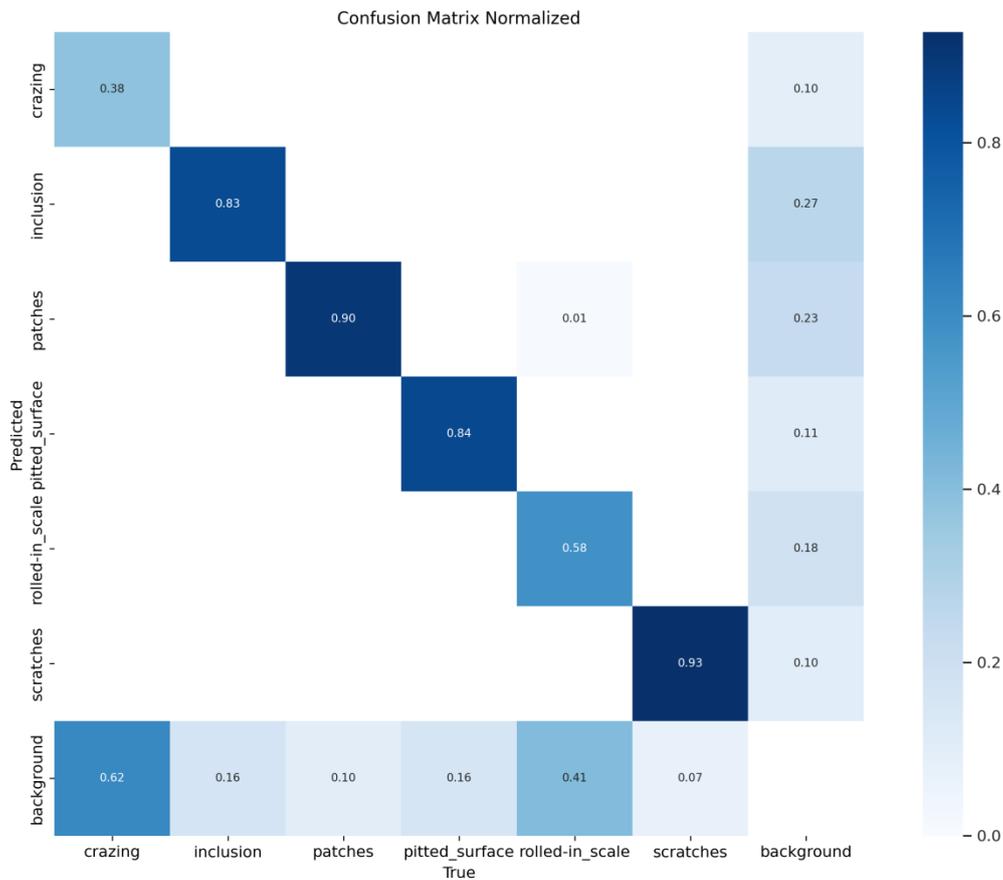


Figure III.16. Matrice de confusion normalisée du modèle YOLO

Pour évaluer les performances d'un modèle de détection d'objets, on utilise une matrice de confusion. On y trouve le nombre de fois où le modèle a correctement classifié un objet (représenté par la diagonale de la matrice) et le nombre de fois où il a incorrectement classifié un objet (représenté par les éléments situés en dehors de la diagonale).

Elle est normalisée, ce qui signifie que les valeurs de chaque cellule sont divisées par le nombre total de tests effectués pour cette classe. Cela facilite la comparaison de classes ayant différents nombres de prédictions.

Penchons-nous plus en détail sur la matrice de confusion :

Classes : La matrice de confusion montre les performances du modèle YOLO pour six classes: Fossé roulé (Rolled in scale), taches (Patches), craquelures (Crazing), surface piquée (Pitted surface), inclusions (Inclusion) et rayures (Scratches).

Chapitre 03 : Classification et localisation des défauts de tôles

Les performances : Les éléments diagonaux de la matrice de confusion reflètent le nombre d'objets correctement classés pour chaque classe. Dans la ligne "rayures" et la colonne "rayures", un résultat de 0,93 signifie que le modèle a bien classé 93 % des objets de la classe "rayures".

Mauvais classement : Les éléments off-diagonaux de la matrice de confusion désignent le nombre d'objets incorrectement classifiés. Dans la colonne "crazing" et la ligne "crazing", la valeur 0,38 signale qu'il y a eu 38 % d'objets "crazing" qui ont été incorrectement classés dans la catégorie "crazing".

Il ressort de la matrice de confusion que le modèle YOLO est performant pour certaines classes (par exemple, les surfaces piquetées), contrairement à d'autres (ex: les fissures). Il existe d'autres informations que l'on peut tirer de la matrice de confusion :

A priori, le modèle classe bien les objets de grande taille (taches) en revanche, il peut avoir des difficultés avec les objets plus petits ou plus minutieux (craquelures).

De plus, il pourrait y avoir une certaine confusion entre les classes "crazing" et "rolled in scale", car elles sont semblables d'un point de vue visuel.

Conclusion

Au cours de ce chapitre, nous avons examiné les résultats de deux modèles de réseaux de neurones convolutifs (CNN) pour la détection des défauts dans les tôles d'acier après le laminage. Le premier modèle utilisé (CNN simple) a atteint une précision de classification de 92,9 %, ce qui prouve qu'il est capable de distinguer les différents types de défauts présents dans les images prises des tôles d'acier. Pour sa part, le deuxième modèle, basé sur l'architecture VGG16, s'est avéré encore plus performant, atteignant une précision de classification de 100 %. Cette précision irréprochable suggère que le modèle a une capacité remarquable à détecter avec précision les défauts dans les images de tôles d'acier.

Le comparatif de ces deux modèles a mis en évidence l'influence de l'architecture du modèle sur ses performances de classification, avec des résultats considérablement améliorés avec le modèle VGG16, qui est plus approfondi et plus complexe. En dernier lieu, nous avons examiné le potentiel d'utilisation du modèle YOLO pour la détection automatique de défauts, en obtenant de bons résultats.

De tels résultats encourageants soulignent le potentiel des approches basées sur les réseaux de neurones convolutifs pour la surveillance et le contrôle de la qualité des pièces fabriquées.

Conclusion générale

Le secteur de la fabrication de tôles d'acier est à la fois complexe et précis en raison de son importance cruciale dans de nombreuses industries. Il arrive cependant qu'au cours du processus de fabrication, de nombreux défauts se forment à la surface, ce qui nuit à la qualité du produit. Pour le fabricant, cela se traduit également par des pertes coûteuses.

Pour ce faire, nous avons testé et évalué les performances de deux modèles de réseaux de neurones convolutifs permettant de classifier et repérer les défauts sur les tôles après le processus de laminage. Le premier modèle, doté d'une architecture CNN simple, a atteint une précision de classification de 92,9 % et un taux d'erreur de 7,1 %, ce qui montre une capacité significative à identifier les différents types de défauts présents dans les images de tôles d'acier.

De son côté, le second modèle basé sur l'architecture VGG16 fait preuve d'une précision de classification de 100 % et d'un taux d'erreur de 0 %, un résultat remarquable. Cette précision parfaite suggère que le modèle est extrêmement efficace pour identifier avec précision les imperfections dans les images de tôles d'acier.

Par ailleurs, la comparaison de ces deux modèles a mis en évidence la contribution de l'architecture du réseau aux performances de classification. En effet, des améliorations significatives ont été observées dans le cas du modèle VGG16, qui est plus profond et plus complexe.

En dernier lieu, une tentative a été faite d'utiliser le modèle YOLO pour détecter automatiquement les défauts. Les résultats obtenus sont satisfaisants, avec une précision de 80%.

De ce fait, l'intégration de l'intelligence artificielle dans le processus de contrôle permet désormais un suivi efficace du processus de laminage, contribuant ainsi à l'optimisation des performances globales de l'usine.

Bibliographie

- [1]. B. FERIÉL. Tout savoir sur le laminoir industriel, Publié en ligne le 18/04/2016.
<https://www.usinenouvelle.com/expo/guides-d-achat/le-laminoir-industriel-528>
- [2]. B. ILYES EL HOUARI. Année Universitaire 2017/2018. Laminage à chaud. Mémoire de fin d'étude. Université BADJI MOKHTAR-ANNABA.
- [3]. M. ZOHEIR. Année Universitaire 2018/2019. Détection et Identification des défauts de surface des produits plats dans les processus de laminage par vision artificielle et algorithmes intelligents. Thèse de doctorat. Université 8 Mai 1945-GUELMA.
- [4]. F. ISLAM, M. Rahman. Contrôle des défauts de surface des métaux à l'aide d'un réseau neuronal profond. Article de recherche. Conférence internationale sur l'ingénierie mécanique, industrielle et énergétique 2018.Khulna, BANGLADESH.
- [5]. COGNE Compagnies. Présentation de la vision industrielle : Guide d'automatisation du process et d'amélioration de la qualité.
- [6]. T. Abdelmalik, D. Luc. Cours de vision industrielle. Université du Littoral Côte d'Opale-France.
- [7]. Hennie de Harder. Tout ce que vous devez savoir sur la régularisation, Publié en ligne dans Towards Data Science le 25/01/2023.
<https://towardsdatascience.com/everything-you-need-to-know-about-regularization-64734f240622>
- [8]. A. Defazio. Apprentissage Profond, Publié en ligne le 24/02/2020.
<https://atcold.github.io/NYU-DLSP20/fr/>
- [9]. M. Mohammed Zakaria. Année Universitaire 2016/2017. Classification des images avec les réseaux de neurones convolutionnels. Mémoire de fin d'étude. Université Abou BakrBelkaid-Tlemcen.
- [10]. T. M'hamed, S. Yasmina. Année Universitaire 2018/2019.Etude comparative de solution de deeplearning de classification d'images médicales. Mémoire de fin d'étude. Université Mouloud Mammeri - Tizi-Ouzou.
- [11]. Antoine Krajnc. Qu'est-ce qu'un réseau de neurones en deeplearning, Publié en ligne le 03/05/2024.
<https://www.jedha.co/formation-ia/reseau-neurones-deep-learning>

[12]. H. Naciri, N. Chaoui. Année Universitaire 2002/2003. Conception et Réalisation d'un système automatique d'identification des empreintes digitales, Mémoire de fin d'étude, Université de Tlemcen.

[13]. M.Fethia. Année Universitaire 2020/2021. Détection d'objets par Deep Neural Network à l'aide du modèle YOLO en temps réel. Mémoire de fin d'étude. Université Mouloud Université 8 Mai 1945-GUELMA.

[14]. Sarah Madeleine. Présentation du réseau de neurones convolutifs appliqué aux images médicales. Publié en ligne le 11/05/2022.

<https://www.imaios.com/fr/ressources/blog/classification-des-images-medicales-comprendre-le-reseau-de-neurones-convolutifs-cnn>

[15]. Raphael Kassel. Qu'est-ce que la méthode Grad-CAM ? Publié en ligne le 09/07/2021.

<https://datascientest.com/grad-cam>