



République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de sciences et de technologies Houari Boumediène

Faculté d'Informatique

Département d'Intelligence artificielle et Science de données



Mémoire de Master

Informatique

Spécialité : Informatique visuelle - MIV-06

Thème

**Détection de changement en milieu urbain par images
d'observation de la terre et techniques avancées
d'apprentissage profond.**

Encadré par

— IFTENE M.

Réalisé par

— CHEMLOUL Mounir

— FERDJI Elias

Membres de jury

— PRÉSIDENTE : LAICHE N.

— MEMBRE : BELHADI H.

2022/2023

REMERCIMENTS

Nous souhaitons adresser nos remerciements les plus sincères aux personnes qui nous ont apporté leur aide et qui ont contribué de près ou de loin à l'élaboration de ce mémoire.

Toute notre reconnaissance et toute notre gratitude vont vers nos familles qui nous ont aidé et accompagné tout au long de cette expérience avec beaucoup de patience et d'enthousiasme.

On voudrait aussi exprimer notre reconnaissance envers les amis et notre promo qui nous ont apporté leur support moral et intellectuel tout au long de cette année.

Au terme de ce travail de recherche, nous souhaitons exprimer notre profonde gratitude au Dr. IFTENE pour son accompagnement précieux tout au long de notre parcours. Sa présence, son expertise et ses conseils éclairés ont été d'une aide inestimable pour mener à bien ce travail.

RÉSUMÉ

Dans ce mémoire de fin d'études, nous explorons d'abord le domaine de la télédétection, une technologie clé pour l'observation et la surveillance de la Terre. Ensuite, nous nous penchons sur la détection de changements, une application cruciale de la télédétection qui permet de suivre les modifications de l'environnement terrestre au fil du temps. Nous étudions les différentes approches et techniques utilisées pour détecter les changements, en mettant l'accent sur les méthodes basées sur l'apprentissage automatique. Le troisième chapitre se concentre sur l'application de l'apprentissage profond à la détection de changements. Nous explorons les différents types de réseaux de neurones profonds utilisés dans ce contexte, tels que les réseaux de neurones convolutifs (CNN). Par la suite, on présentera deux méthodes qu'on a adaptées et conçues pour notre tâche, nous détaillons l'architecture de chaque méthode, les étapes de l'entraînement du modèle et les métriques d'évaluation utilisées pour mesurer les performances. Enfin, nous présentons les résultats de nos expériences et interprétons ces résultats, démontrant l'efficacité de nos méthodes et soulignant les implications de notre travail pour le domaine de la télédétection et la détection de changements. Nous discutons des avantages et des limites de nos approches, ainsi que des perspectives d'amélioration et des pistes de recherche futures.

Mots-clés :

Télédéction, Détection de changement, Apprentissage profond, Images Satellitaires.

Table des matières

INTRODUCTION	11
1 LA TELEDETECTION ET IMAGES SATELLITAIRES	17
1.1 Définition de la télédétection	18
1.2 Fonctionnement de la télédétection	18
1.3 Images de télédétection	20
1.4 Les capteurs en télédétection	20
1.5 Les types d'images satellitaires	21
1.5.1 Les images panchromatiques	21
1.5.2 Les images multispectrales	21
1.5.3 Les images hyperspectrales	21
1.5.4 Les images thermiques	22
1.5.5 Les images radar	22
1.5.6 Caractéristiques des images satellitaires	23
1.6 Domaines d'application de la télédétection	23
i Cartographie et surveillance de l'environnement	24
ii Gestion des ressources naturelles	24
iii Géologie et étude des catastrophes naturelles	24
iv Planification urbaine et gestion des zones côtières	24
1.7 Conclusion	25
2 DETECTION DE CHANGEMENTS EN TELEDETECTION	26
2.1 Détection de changements:	26
2.1.1 Définition	26
2.2 Méthodes de détection de changements	28
2.2.1 Méthodes basées sur la différence d'images	29
i Soustraction simple	29
ii Soustraction du fond	29
2.2.2 Méthodes basées sur la transformation	30

2.2.3	Méthodes basées sur la classification	30
i	Classification non supervisée	30
ii	Méthodes supervisée	31
iii	Classification semi-supervisée (hybride)	31
3	APPRENTISSAGE PROFOND EN DETECTION DE CHANGEMENTS	33
3.1	L'apprentissage profond	33
3.2	Les réseaux de neurones	34
3.2.1	Définition d'un réseau de neurones	34
3.2.2	Le Perceptron	35
3.2.3	Le Multi layer perceptron	36
3.2.4	Les types de réseaux de neurones	37
i	Les réseaux de neurones récurrents (RNN)	37
ii	Les GANs (Generative Adversarial Networks)	38
iii	Les réseaux de neurones convolutionnels (CNN)	39
3.3	réseaux de neurones convolutifs	40
3.3.1	Les couches des CNN	40
i	Couche de convolution	40
ii	Couche de pooling	41
iii	Couche de entièrement connectée	42
iv	Les fonctions d'activation	43
3.4	Exemples d'architectures CNN	45
3.4.1	LeNet	45
i	Architecture	45
ii	Caractéristiques	45
3.4.2	AlexNet	46
i	Architecture	46
ii	Caractéristiques	47
3.4.3	VGG16	48
i	Architecture	48
ii	Caractéristiques	49
3.4.4	ResNet	49
i	Architecture	50
ii	Caractéristiques	50
3.4.5	Xception :	51
i	Architecture :	52
ii	Caractéristiques :	52

3.5	Les types d'apprentissage :	53
3.5.1	Entrainement de zero	53
3.5.2	Le transfert learning	53
i	Definition:	53
ii	Catégories de Transfert Learning:	54
iii	Techniques de Transfert Learning:	54
3.5.3	Etude Comparative	55
i	Précision sur le dataset ImageNet:	56
ii	Architecture:	56
iii	Nombre de paramètres:	56
iv	Transfert d'apprentissage:	56
v	Profondeur de l'architecture:	56
3.6	Conclusion	57
4	CONCEPTION ET IMPLEMENTATION DES METHODES	58
4.1	Première méthode:	59
4.1.1	Le modèle basé sur Xception:	59
4.1.2	Algorithme du Xception:	60
4.2	Deuxième méthode:	61
4.2.1	Extraction de caractéristiques	61
4.2.2	Amélioration de la discriminations des caractéristiques	61
4.2.3	Amélioration de l'apprentissage	64
4.2.4	Architecture globale du modèle proposé	65
4.2.5	Algorithme du modele proposé:	67
5	IMPLEMENTATION ET RESULTATS	69
5.1	Présentation du jeu de données	69
5.1.1	Contexte et motivation	69
i	SZTAKI Dataset:	69
ii	LEVIR-CD:	70
iii	SYSU-CD:	70
iv	Etude comparative:	71
5.1.2	Préparation et division du dataset	72
5.1.3	Complémentarité du dataset SYSU-CD par rapport aux ensembles de données existants	73
5.2	Outils utilisés:	73
5.2.1	Python	73

5.2.2	Pytorch	74
5.2.3	Tensorflow:	74
5.2.4	Google Colab	75
5.3	Détails d'apprentissage :	75
5.3.1	Premiere approche:	75
5.3.2	Deuxieme approche:	76
5.4	Critères d'évaluation	78
5.4.1	Classification accuracy	78
5.4.2	Recall (Sensibilité ou Taux de Vrais Positifs)	78
i	Utilité:	78
ii	Formule:	78
5.4.3	Score F1	79
i	Utilité:	79
ii	Formule:	79
5.5	Résultats et discussions :	79
5.5.1	Xception adapté à la détection de changements	79
i	Analyse qualitative:	79
ii	Analyse quantitative:	80
5.5.2	VGG-16 amélioré adapté à la détection de changements	81
i	Analyse qualitative:	81
ii	Analyse quantitative:	82
5.5.3	Discussion des résultats :	83
i	Comparaison entre les deux méthodes:	83

Table des figures

1	Logo de l'Asal.	13
1.1	Les sept étapes du processus de teledetection.	19
1.2	Reconstitution RVB d'une image multispectrale et gauche et une image panchromatique a droite prises avec le satellite IKONOS.	22
1.3	Reconstitution RVB d'une image multispectrale a gauche et une image panchromatique a droite prises avec le satellite QUICKBIRD.	23
2.1	Etapes de la détection de changements	28
2.2	Opération de soustraction de fond.	29
2.3	Diagramme de comparaison post classification (supervised method).	31
3.1	modèle d'un neurone formel.	35
3.2	exemple d'une architecture d'un multi perceptron.	37
3.3	Architecture classique d'un réseau de neurones convolutif (Paul BLANC-DURAND 2018).	38
3.4	Architecture d'un GAN.	39
3.5	Architecture classique d'un réseau de neurones convolutif (Paul BLANC-DURAND 2018).	40
3.6	Exemple de convolution. [1]	41
3.7	Exemple de maxpooling. [2]	41
3.8	Un réseau de neurones avec des couches entièrement connectées.	42
3.9	Courbe de la fonction d'activation ReLU	43
3.10	Courbe de la fonction d'activation sigmoïde	44
3.11	Courbe de la fonction d'activation Tanh	44
3.12	Architecture LeNet-5(LeCun et al 1998).	45
3.13	Architecture du ALEXNET (Researchgate.net).	47
3.14	Architecture du VGG16 (Researchgate.net).	49
3.15	Architecture du resnet50 (Luqman Ali et Fady Shibata Alnajjar et al 2021).	50
3.16	Architecture du réseau Xception.	52

3.17 Comparaison des CNN les plus populaires.(Chris Kawatsu et al, 2017)	55
4.1 Module CBAM	62
4.2 Channel attention du CBAM.	62
4.3 Spatial attention du CBAM	63
4.4 Architecture du modèle proposé.	66
5.1 exmeple d'échantillon du dataset SZTAKI. (Lien du dataset ici)	70
5.2 exmeple d'échentillon du dataset LEVIR-CD. (Lien du dataset ici	70
5.3 exmeple d'échentillon du dataset SYSU-CD (Lien du dataset ici).	71
5.4 Illustration de la séparation des données	73
5.5 Logo de Python.	74
5.6 Logo de Pytorch.	74
5.7 Logo de Tensorflow.	75
5.8 Logo de google colab.	75
5.9 Courbe de précision sur 1000 epoch	76
5.10 Courbe de coût sur 1000 epoch	76
5.11 Courbe de précision sur 50 epoch	77
5.12 Courbe de coût sur 50 epoch	77

Liste des tableaux

5.1	Etude comparative entre les datasets	71
5.2	Hyperparamètres du deuxième modèle	77
5.3	Analyse qualitiative Xception	80
5.4	Analyse quantitative Xception	81
5.5	Analyse qualitative du VGG et de la méthode proposée	82
5.6	Analyse quantitative du VGG simple et de la méthode proposée	82
5.7	Analyse qualitiative des deux méthodes sur la même paire d'images.	83
5.8	Analyse quantitative de la méthode 1 et de la méthode 2 sur la même paire d'images.	83

LIST OF ALGORITHMS

1	Fine tuning du réseau Xception	60
2	Entraînement du réseau proposé	67

INTRODUCTION

Dans le contexte actuel de l'urbanisation croissante, où 70% de la population algérienne est concentrée dans les zones urbaines, la surveillance, l'aménagement et le développement de ces milieux sont devenus des enjeux majeurs. Cette concentration massive de la population entraîne une consommation accrue des espaces et modifie l'écosystème, dégradant ainsi les conditions de vie.

La caractérisation de ces environnements urbains est une tâche complexe et délicate, nécessitant une approche multidimensionnelle. L'aspect géographique, par exemple, est un élément clé qui peut être étudié grâce aux techniques d'observation de la Terre. Les images multi-source de télédétection spatiale ont démontré leur potentiel pour extraire des informations précieuses, non seulement pour la modélisation et le suivi de l'environnement urbain, mais aussi pour diverses autres applications, telles que le développement des constructions, la croissance de la population, la gestion de l'énergie, l'aménagement des espaces verts, les transports urbains, la pollution de l'air et de l'eau, le ramassage des déchets et le recyclage, etc.

C'est dans ce contexte que s'inscrit notre projet de fin d'études, qui fait partie du projet de recherche et de développement CTS/D.OT/S.TSIT01/2022, validé par le Comité Scientifique du Centre des Techniques Spatiales. Intitulé « Caractérisation Des Composantes d'un Milieu Urbain par Images D'observation de la Terre et Techniques Avancées D'apprentissage Machine », ce projet vise à caractériser les différentes composantes du milieu urbain en utilisant, développant et évaluant des méthodes avancées d'apprentissage machine, en particulier les méthodes d'apprentissage profond.

Notre objectif est de détecter les changements dans l'environnement urbain et de caractériser ses différentes composantes en utilisant des méthodes d'apprentissage profond basées sur les réseaux de neurones à convolutions. Ces méthodes sont dédiées au traitement de

différents types d'images optiques de télédétection spatiale, notamment des données multi-résolution et multiday. Nous visons également à évaluer l'apport de ces méthodes en termes d'automatisation et d'amélioration des performances par rapport aux approches classiques. Le projet se concentrera sur les tâches et les aspects méthodologiques suivants : la préparation des jeux de données de différents types de milieu urbain, l'utilisation et le développement de méthodes avancées de traitement des données, la segmentation et la classification des données, la détection d'objets, la détection de changements, et l'évaluation et la validation des résultats.

Dans ce mémoire, nous allons proposer deux méthodes de détection de changement dans les images optiques de télédétection spatiale en utilisant les réseaux de neurones convolutifs et comparer les résultats de ces dernières. Nous commençons par explorer le domaine de la télédétection puis on va établir les généralités et l'état de l'art de la détection de changement, un domaine qui a connu des avancées significatives mais qui présente encore des défis en termes de conception et de formation systématique des réseaux.

Ensuite, nous plongeons dans le domaine de l'apprentissage profond et des réseaux de neurones, explorant les possibilités qu'ils offrent pour améliorer les méthodes de détection de changement. Nous nous concentrerons sur la conception d'un nouveau modèle qui, nous l'espérons, apportera des améliorations significatives et le comparer avec un deuxième modèle plus profond.

L'organisme d'accueil

L'Agence spatiale algérienne ,également désignée par son acronyme ASAL est l'agence spatiale responsable du programme spatial algérien. L'Agence spatiale algérienne est un établissement public national à caractère spécifique, doté de la personnalité morale et de l'autonomie financière. Elle a été créée auprès du chef du gouvernement par le décret présidentiel no 02-48 du 16 janvier 2002.



FIGURE 1 – Logo de l’Asal.

L’ASAL conçoit et met en œuvre de la politique nationale de promotion et de développement de l’activité spatiale. Son objectif principal est de faire de l’outil spatial un vecteur performant de développement économique, social et culturel du pays et d’assurer la sécurité et le bien-être de la communauté nationale.

Organisation

L’Agence spatiale algérienne est constituée d’une structure centrale et de quatre entités opérationnelles.

Organisation administrative

L’agence est dotée d’un conseil d’administration et d’un conseil scientifique et technique. Elle a à sa tête un directeur général nommé par le président de la République. Les membres du conseil d’administration sont, outre le directeur général et un représentant du chef du gouvernement, les ministres détenant les principaux portefeuilles susceptibles d’être concernés par les applications satellitaires représentant 15 départements ministériels. L’agence comprend un comité scientifique composé d’experts dans les domaines des technologies et applications spatiales.

Entités opérationnelles :

Centre des techniques spatiales (CTS) Le Centre des techniques spatiales (CTS) est chargé de mener toutes les actions d’études et de recherches scientifiques et techniques dans les domaines :

- De la technologie spatiale, notamment les techniques liées aux capteurs, aux radiomètres, aux télécommunications spatiales, aux stations terriennes de réception et de contrôle ainsi qu’aux engins et instruments d’observation de la terre et de l’atmosphère;

de la physique de la télédétection aérospatiale, du bilan d'énergie au sol et de la physique de l'atmosphère; de la méthodologie de traitement des images satellitaires et du traitement des banques de données images; de la géodésie spatiale et des systèmes de références, des techniques et systèmes de navigation par satellites, de la radio-astronomie et l'altimétrie spatiale, de la détermination du champ de pesanteur et du géoïde, et des applications géodynamiques .

- De la géomatique, des bases de données et systèmes d'informations géographiques, des méthodes d'acquisition (topographique, photogrammétrie, télédétection et cartographie), de traitement et de restitution des données géographiques.

Centre des applications spatiales (CAS) Le Centre des applications spatiales (CAS) est chargé de mettre en œuvre les actions d'exploitation des satellites et des systèmes découlant des programmes spatiaux, en relation avec les différents secteurs utilisateurs.

Le centre assure la réalisation des projets opérationnels sectoriels et intersectoriels basés sur la télédétection et les systèmes d'information géographique, particulièrement dans les domaines de l'environnement et des risques naturels, de l'agriculture et des ressources en eau, de l'aménagement du territoire et de l'urbanisme ainsi que de la géologie et des sciences de la terre.

Centre de développement des satellites (CDS) Le Centre de développement des satellites (CDS), inauguré le 23 février 2012, le CDS est implanté dans la commune de Bir El Djir, dans la wilaya d'Oran, est une entité opérationnelle de l'Agence spatiale algérienne dont la réalisation est une action planifiée dans le programme Spatial National horizon 2020. Construit sur une superficie de 4,7 hectares, et pourvu d'un espace vie comprenant des logements de fonction et des terrains de sport, le CDS apportera une impulsion certaine au processus de maîtrise des technologies spatiales en Algérie. Le CDS est constitué d'infrastructures modernes (ateliers et laboratoires) dédiés à la conception, l'assemblage des satellites ainsi que des moyens de test et d'essais d'environnement. Le Centre de développement des satellites (CDS) est chargé de la conception, du développement et de la réalisation des systèmes spatiaux prévus dans le cadre du programme spatial national, notamment :

la mise à contribution de l'industrie nationale dans les domaines connexes des technologies spatiales, notamment les domaines de la mécanique, de l'électronique, de l'optique, de l'informatique et des télécommunications.

la réalisation des satellites, la conduite des tests fonctionnels sur les satellites (essai d'interférence et de compatibilité électromagnétique, essai de vide thermique, essai de vibration et

essai acoustique); l'assurance qualité des activités d'intégration et d'essai sur les systèmes spatiaux; l'émergence d'un tissu industriel dans les domaines connexes des technologies spatiales, notamment dans les domaines de l'électronique, l'informatique, la mécanique, l'optique et des télécommunications.

Centre d'exploitation des systèmes de télécommunications (CEST) Le Centre d'exploitation des systèmes de télécommunications (CEST) est chargé de la gestion, de l'exploitation et de la commercialisation des produits et services de satellites de télécommunications prévus dans le cadre du programme spatial national, notamment :

- La gestion technique des infrastructures terrestres de réception et de contrôle .
- La prise en charge des produits et services des satellites en relation avec les secteurs utilisateurs concernés .
- La définition et la mise en œuvre d'une politique de commercialisation des produits et services.

Mission

L'Asal est l'instrument de conception et de mise en œuvre de la politique nationale de promotion et de développement de l'activité spatiale.

Son objectif principal est de faire de l'outil spatial un vecteur performant de développement économique, social et culturel du pays et d'assurer la sécurité et le bien-être de la communauté nationale.

Elle est dotée d'un conseil d'administration composé des représentants de 15 départements ministériels; d'un comité scientifique composé d'experts dans les domaines des technologies et applications spatiales;

Missions et attributions

1. Proposer au gouvernement les éléments d'une stratégie nationale dans le domaine de l'activité spatiale et d'en assurer l'exécution .
2. Mettre en place une infrastructure spatiale destinée à renforcer les capacités nationales.
3. Mettre en œuvre les programmes annuels et pluriannuels de développement des activités spatiales nationales en relation avec les différents secteurs concernés et d'en assurer le suivi et l'évaluation.

4. Proposer au Gouvernement les systèmes spatiaux les mieux adaptés aux préoccupations nationales et d'assurer, pour le compte de l'état, leur conception, leur réalisation et leur exploitation.
5. Proposer au gouvernement une politique de coopération bilatérale et multilatérale adaptée aux besoins nationaux.
6. Assurer le suivi et l'évaluation des engagements découlant des obligations de l'État en matière d'accords régionaux et internationaux dans les domaines de l'activité spatiale.

CHAPITRE**1****LA TELEDETCTION ET IMAGES
SATELLITAIRES**

Dans ce premier chapitre, nous allons explorer le domaine de la télédétection et des images satellitaires. La télédétection, une technologie qui nous permet d'acquérir des informations sur la surface de la Terre sans être en contact direct avec elle, a révolutionné notre façon de comprendre et d'interagir avec notre environnement.

Nous commencerons par définir la télédétection et expliquer son fonctionnement, avant de nous pencher sur les images satellitaires, un outil essentiel de la télédétection. Nous examinerons les différents types de capteurs utilisés en télédétection, ainsi que les divers types d'images satellitaires, y compris les images panchromatiques, multispectrales, hyperspectrales etc... Chacun de ces types d'images offre une perspective unique et précieuse sur la surface de la Terre, et nous discuterons de leurs caractéristiques spécifiques.

Ensuite, nous explorerons les nombreuses applications de la télédétection et nous examinerons comment elle est utilisée dans ces différents contextes.

1.1 Définition de la télédétection

"l'acquisition d'informations sur un objet ou un phénomène sans contact physique avec l'objet" [3].

"La télédétection est une technique permettant d'obtenir des informations sur la surface terrestre à partir de capteurs installés sur des satellites, des avions, des drones ou des ballons. Cette technique est utilisée dans de nombreux domaines, tels que la cartographie, la surveillance environnementale, la gestion des ressources naturelles, la planification urbaine, la géologie, etc." [4].

La télédétection est donc une technique permettant l'observation de la terre a partir de l'espace ou des airs grâce à différents moyens tels que les satellites, les ballons ou encore les avions, afin de pouvoir analyser les différentes caractéristiques des objets en se basant sur leurs capacité à réfléchir l'énergie en fonction de leurs propriétés physiques et ainsi fournir des information sur leurs nature et leurs état. Pour la collecte de données, il existe différents capteurs par exemple des caméras ou des radars qui enregistrent le rayonnement électromagnétique afin de pouvoir étudier les objets physiques.

1.2 Fonctionnement de la télédétection

Dans la télédétection, on se base sur la capacité des capteurs à récupérer le rayonnement électromagnétique , qu'il soit émis ou réfléchi par les objets sur terre tels que les bâtiments, les arbres, le sol ... Les satellites jouent un rôle clé dans cette collecte d'informations, captant les signaux émis ou réfléchis par une multitude d'objets sur notre planète, puis les données sont partagées avec d'autres plateformes d'observation ou peuvent par exemple être envoyées à des organisations telles que les agences spatiales qui vont ensuite se charger de traiter pour les visualiser et les interpréter. Ces signaux sont ensuite traités pour produire des informations utiles. Un exemple courant de ce processus est la récupération d'images satellitaires, qui peuvent être analysées pour déterminer l'humidité des sols.

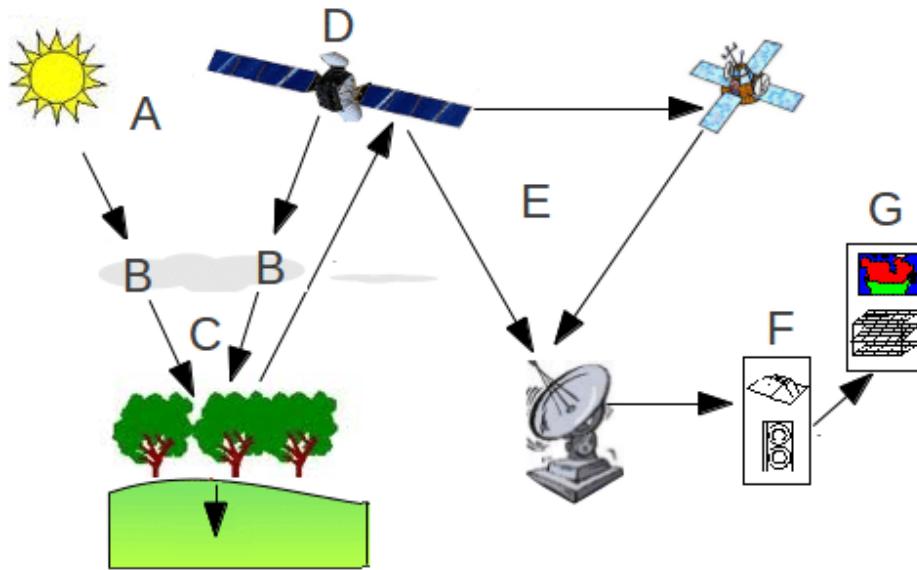


FIGURE 1.1 – Les sept étapes du processus de teledetection.

(A) : Source d'énergie

(B) : Rayonnement émis

(C) : Interaction avec la cible

(D) : Enregistrement de l'énergie par le capteur

(E) : Transmission, réception et traitement

(F) : Interprétation et Analyse

(G) : Application

On distingue deux types de télédétection : la télédétection active et passive. Dans la télédétection passive, le satellite s'appuie sur le rayonnement émis par des sources naturelles comme le soleil afin de pouvoir récupérer le rayonnement émis par le ou les objets étudiés comme par exemple les satellites Landsat-7, Landsat-8 et Modis qui possèdent des capteurs optiques passifs, mais l'inconvénient de cette méthode réside dans le fait qu'il est impossible de récupérer les rayonnements de nuit ou quand la trajectoire est obstruée par les nuages, c'est pour cela donc que la détection active complète la télédétection passive dans la mesure où dans la télédétection active, c'est la plateforme d'observation aérienne elle-même qui va se charger d'émettre les ondes électromagnétiques vers l'objet d'intérêt, comme par exemple le satellite Sentinel-1 avec ses capteurs radars actifs .

Dans le second cas, la terre peut être surveillée en permanence de jour comme de nuit, de plus les changements météorologiques n'influent pas sur les données désirées, par exemple un satellite peut émettre des ondes radar si il y a des nuages car ces derniers ne sont pas visibles dans les fréquences élevées comme les fréquences radar [5].

Dotés de capteurs de pointe, ces satellites sont spécialement conçus pour générer des images détaillées nous fournissant une fenêtre privilégiée sur notre monde en nous permettant de percevoir et de comprendre les complexités de notre planète de manière sans précédent

1.3 Images de télédétection

Une image de télédétection est une représentation numérique ou analogique de la surface de la Terre ou d'une partie de celle-ci, obtenue à l'aide de capteurs embarqués sur des plateformes aériennes ou spatiales. Ces capteurs enregistrent les réflexions ou émissions d'énergie électromagnétique provenant de la surface terrestre dans différents segments du spectre électromagnétique, qui peuvent ensuite être analysés pour obtenir des informations sur les caractéristiques et les conditions de cette surface. Ces images peuvent être capturées dans diverses bandes du spectre électromagnétique, y compris la lumière visible, l'infrarouge, le micro-ondes, etc. Chaque bande du spectre révèle des informations différentes sur la surface de la Terre, permettant aux scientifiques d'étudier une variété de phénomènes environnementaux et géologiques.

1.4 Les capteurs en télédétection

Les capteurs sont des outils permettant de quantifier l'énergie électromagnétique émise ou réfléchie par la surface terrestre afin de les utiliser pour l'obtention d'images satellitaires qui seront par la suite analysées avec pour but l'extraction d'informations sur la surface terrestre[6]. Il existe plusieurs types de capteurs en télédétection. Les principaux sont :

Les capteurs optiques

Les capteurs optiques sont les capteurs utilisés pour la télédétection passive, ils récupèrent les ondes électromagnétiques émis ou réfléchis par la surface de la terre afin de pouvoir construire leurs images. Généralement les capteurs optiques servent à récupérer des images aériennes sur trois canaux.

Les capteurs radar

Les capteurs radar comme par exemple le capteur SAR, sont ceux que l'on emploie lors de la télédétection active. Il permettent de récupérer des informations cruciales grâce à la sensibilité de ces derniers tout en donnant la possibilité de le faire de jour comme de nuit

sans être obstrué par de mauvaises conditions météorologiques. Cependant la qualité spatiale ainsi que le contraste sur les images radar sont faibles.

Les capteurs Lidar

Un signal lumineux généré par un laser ou une LED est transmis à l'objet dans un système LiDAR pour mesurer le temps de vol du signal optique afin de calculer la distance entre la caméra et la cible. Contrairement aux systèmes qui scannent un objet point par point, les technologies utilisant le laser appartiennent à une catégorie appelée scanner LiDAR où l'objet entier est rayé avec l'impulsion de lumière.

1.5 Les types d'images satellitaires

1.5.1 Les images panchromatiques

Ce sont des images en noir et blanc obtenues à partir de capteurs qui mesurent la lumière dans les bandes spectrales du visible. Les images panchromatiques ont une résolution spatiale élevée. Elles sont utilisées généralement comme données complémentaires afin d'améliorer la qualité d'autres images ayant une résolution spatiale moins élevée par exemple les images SAR capturées avec le capteur SAR (Synthetic Aperture Radar). La résolution spatiale élevée signifie qu'elles permettent de distinguer plus de détails au niveau de la surface terrestre [7].

1.5.2 Les images multispectrales

Ce sont des images obtenues à partir de capteurs qui mesurent la lumière réfléchie par la surface terrestre dans plusieurs bandes spectrales, généralement dans le visible et l'infrarouge proche. Les images multispectrales permettent de distinguer les différentes caractéristiques de la surface terrestre en fonction de leur réflectance dans ces différentes bandes spectrales.

1.5.3 Les images hyperspectrales

Les images hyperspectrales sont des images qui possèdent une résolution spectrale élevée qui sont très utilisées dans les applications de vision par ordinateur. Cependant la résolution spatiale de ce type d'images est faible et nécessite généralement d'être fusionnée avec une autre image (une image panchromatique par exemple) afin d'être corrigée [8].

1.5.4 Les images thermiques

Ce sont des images obtenues à partir de capteurs thermiques qui mesurent l'énergie infrarouge émise par la surface terrestre et peuvent servir à la surveillance des feux de forêt par exemple. Les images thermiques permettent de visualiser les variations de température de la surface terrestre, cependant elles possèdent une résolution spatiale faible ainsi qu'un mauvais contraste[9].

1.5.5 Les images radar

Les images radar fournissent des informations précieuses sur la surface de la terre, en effet, elles permettent de retranscrire une représentation de la surface terrestre accrue en termes de rugosité de la surface, la forme et l'orientation des objets grâce à sa sensibilité à la géométrie des cibles. De plus Les images radar permettent de visualiser la topographie de la surface terrestre, même sous les nuages et la végétation [7]



FIGURE 1.2 – Reconstitution RVB d'une image multispectrale et gauche et une image panchromatique a droite prises avec le satellite IKONOS.

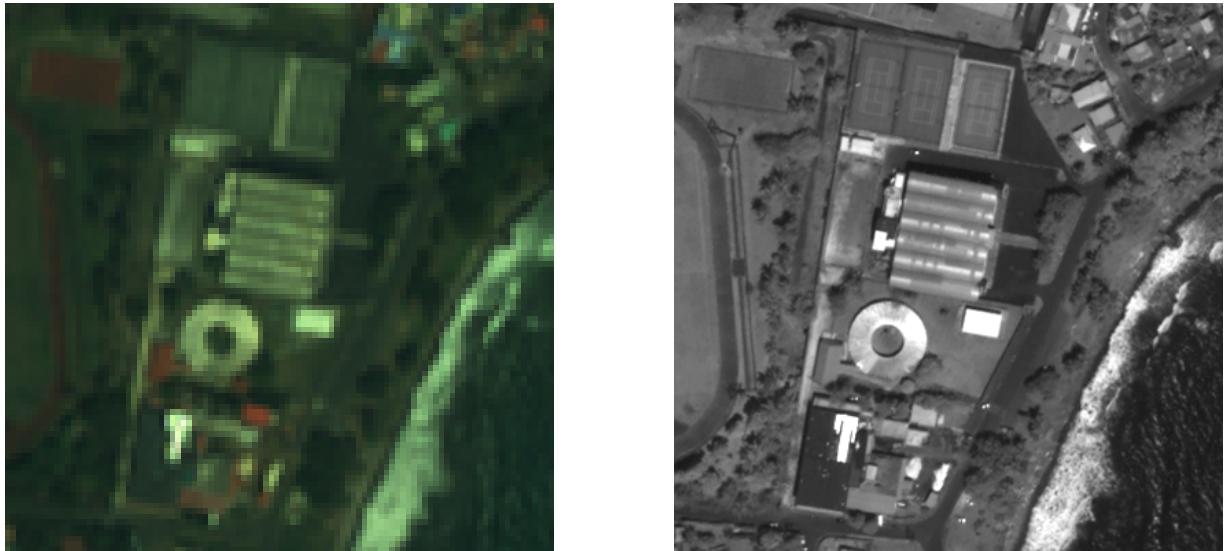


FIGURE 1.3 – Reconstitution RVB d'une image multispectrale a gauche et une image panchromatique a droite prises avec le satellite QUICKBIRD.

1.5.6 Caractéristiques des images satellitaires

Les images satellitaires présentent plusieurs caractéristiques importantes à prendre en compte pour leur utilisation [10] :

- **Résolution spatiale** : La résolution spatiale d'une image satellitaire représente la taille de la zone couverte par un pixel, généralement exprimée en mètre/pixel. Elle dépend de la taille des capteurs ,de la distance focale et de l'altitude de vol. Plus la résolution spatiale est fine, plus l'image est détaillée
- **Résolution spectrale** : C'est le nombre de bandes spectrales dans lesquelles l'image a été observée.
- **Résolution radiométrique** : La résolution radiométrique est la gamme de valeurs de luminosité disponibles.
- **Résolution temporelle** : La résolution temporelle est la fréquence à laquelle de nouvelles images de la même zone sont acquises.

1.6 Domaines d'application de la télédétection

La télédétection est une technologie utile dans de nombreux domaines d'application. Dans cette section, nous présentons quelques exemples d'applications de la télédétection.

i Cartographie et surveillance de l'environnement

Les images issues de la télédétection permettent de faire une cartographie très fidèle de la surface terrestre avec tout ce qu'elle englobe comme forêts, cours d'eau, océans ... afin de se rendre compte de l'état de notre environnement. Cette cartographie est très pratique car elle permet avec la notion de détection de changements de surveiller l'évolution de l'environnement. Nous pouvons prendre pour exemple dans les travaux de Ibraim et al [11] qui surveillent la déforestation grâce aux images de la forêt d'Argania issues du satellite Sentinel-2, mais aussi les travaux de Su et al [12] pour l'étude du réchauffement des océans.

ii Gestion des ressources naturelles

La télédétection devient un incontournable dans le domaine de l'agriculture aujourd'hui grâce à de nombreux travaux visant à minimiser l'impact environnemental, tout en augmentant la production et la productivité. Il existe aujourd'hui plusieurs applications qui nous permettent de surveiller la vigueur de la végétation et du stress dû à la sécheresse mais aussi d'évaluer le développement des cultures[13]. Dans la foresterie, les travaux de Suratman et al [14] nous ont permis de nous rendre compte de l'importance de la télédétection dans le domaine, notamment lors de l'élaboration de stratégies de planification et de gestion forestière qui nécessite la collecte de données sur les ressources forestières C'est aussi le cas pour la domaine de l'élevage d'algues et d'animaux marins qui est en pleine expansion, la télédétection vient en aide dans le but de surveiller la concurrence de la mariculture avec les autres activités des zones côtières[15].

iii Géologie et étude des catastrophes naturelles

En géologie, la télédétection apporte une grande aide en permettant aux scientifiques de mieux comprendre les phénomènes géophysiques qui conduisent à des risques naturels tels que les tremblements de terre ou les éruptions volcaniques [16]. L'avancée dans la compréhension de ces phénomènes permet aujourd'hui de mieux prédire leurs occurrences et d'améliorer les outils de prédition et de décision afin de les atténuer[17].

iv Planification urbaine et gestion des zones côtières

Dans le cas de la planification urbaine, ce domaine est très étudié en télédétection dans la branche de la détection de changements qui se base sur résolution temporelle des images afin de détecter les changements au niveau urbain. Il existe d'ailleurs une multitude de jeux de données aujourd'hui en rapport avec la détection de changements en milieu urbain, un des plus connus est LEVIR-CD. La télédétection aide aussi à la gestion des zones côtières

comme nous le montrent Christopher Small et Robert J. Nicholls [18]dans leurs étude de distribution de la population au niveau des côtes

1.7 Conclusion

En conclusion, la télédétection est une technique puissante qui permet d'acquérir des informations sur la surface terrestre à partir de capteurs embarqués sur des satellites ou des avions. Les images obtenues grâce à cette technique offrent une vue d'ensemble de la planète, ce qui permet de surveiller et de mieux comprendre les phénomènes terrestres. Les différents types d'images, tels que les images optiques, panchromatiques, multispectrales et hyperspectrales, offrent des informations variées sur la surface terrestre, ce qui permet de répondre à différents besoins en matière de cartographie, de surveillance environnementale, d'agriculture, de gestion des ressources naturelles, etc.

Cependant, une des applications les plus importantes de la télédétection est la détection de changements. En effet, l'analyse des images acquises à différents moments permet de détecter les changements survenus sur la surface terrestre, tels que l'urbanisation, la déforestation, les mouvements de terrain, etc. Cette analyse peut être réalisée de manière manuelle ou automatisée, et permet de mieux comprendre les phénomènes de changement survenant sur notre planète.

CHAPITRE 2

DETECTION DE CHANGEMENTS EN TELEDETECTION

Dans ce chapitre, nous allons explorer le domaine de la détection de changement en télédétection, une discipline qui a connu des avancées significatives ces dernières années. Nous allons nous pencher sur les différentes méthodes utilisées pour détecter les changements, notamment les méthodes basées sur la différence d'images, les méthodes basées sur la transformation et celles basées sur la classification. Chacune de ces méthodes offre une perspective unique et des outils précieux pour analyser et interpréter les données de télédétection. Tout au long de ce chapitre, nous allons examiner en détail ces méthodes, en mettant en lumière leurs forces et leurs limites. En conclusion, nous replacerons les méthodes étudiées dans leur contexte en les alignant sur les avancées technologiques et les tendances actuelles dans le domaine de la détection de changement en télédétection.

2.1 Détection de changements :

2.1.1 Définition

La détection de changements en télédétection est un processus basé sur l'identification des variations d'état d'un objet ou d'un phénomène afin d'identifier les différences entre deux ou plusieurs images prises à des moments différents. Autrement dit, le processus de détection de changement repose sur l'observation des différences entre plusieurs images multi-

temporelles capturées sur la même zone afin d'assister l'homme dans des tâches telles que la surveillance, la cartographie urbaines... Mishra S. et al. [19] nous rapportent les six étapes principales de la détection de changements à savoir :

- La nature du problème de la détection de changements
 - La détection de changements vise à identifier les variations d'état d'un objet ou d'un phénomène à partir de plusieurs images prises à des moments différents.
 - L'objectif est d'assister les utilisateurs dans des tâches telles que la surveillance ou la cartographie urbaine.
- Sélection des données
 - Il est essentiel de choisir les bonnes images multi-temporelles pour effectuer la détection de changements.
 - Les images doivent être capturées sur la même zone géographique pour permettre une comparaison précise.
- Pré-traitement des images
 - Avant de procéder à la détection de changements, les images doivent être pré-traitées pour éliminer les artefacts et améliorer leur qualité.
- Traitement des images ou classification
 - Une fois les images pré-traitées, elles sont analysées pour identifier les changements significatifs.
- Conception ou adaptation d'un algorithme de détection de changements
 - Un algorithme spécifique est conçu ou adapté pour détecter les changements dans les images.
- Evaluation des résultats et validation
 - Les résultats de la détection de changements sont évalués pour mesurer leur précision et leur fiabilité.

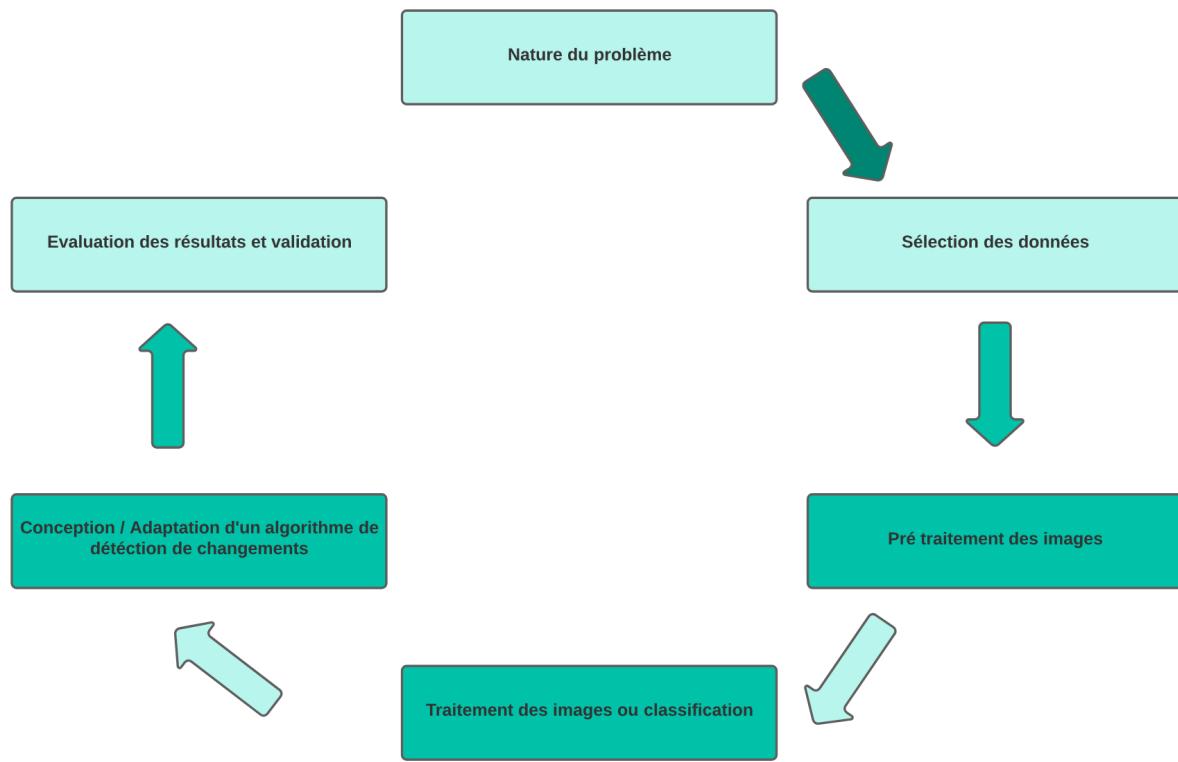


FIGURE 2.1 – Etapes de la détection de changements

2.2 Méthodes de détection de changements

La détection de changements repose sur un large éventail de méthodes, chacune offrant des approches uniques pour identifier et analyser les modifications survenues. Parmi ces méthodes, on retrouve notamment celles basées sur la différence d'images, les transformations et la classification. Chacune de ces approches présente des avantages et des limitations spécifiques, et leur sélection dépendra des caractéristiques des données et des objectifs de l'étude.

2.2.1 Méthodes basées sur la différence d'images

Les méthodes algébriques qui se basent sur la différence d'images consistent à soustraire une image I₁ à une autre image I₂ pour obtenir en sortie une nouvelle image qui illustre les changements, ces changements apparaissent sous forme de pixels qui diffèrent des pixels présents sur les images originales. Bien que cette méthode est simple et rapide, elle est très sensible aux variations atmosphériques et aux erreurs de calage entre les images[20].

i Soustraction simple

La méthode de soustraction d'images consiste à soustraire les valeurs des pixels des deux images pour rendre en sortie une toute nouvelle image qui représente les différences entre ces deux dernières. Les pixels avec la même valeur dans les deux images auront une valeur de 0 dans l'image de résultante , tandis que les pixels qui ont une valeur différente ont une valeur non nulle. Pour une meilleure précision de la détection, un seuil (threshold) de détection peut être utilisé pour filtrer les pixels de faible intensité. cette méthode est plutôt simple et directe et qui rend des résultats faciles à interpréter.

ii Soustraction du fond

La soustraction du fond est une autre méthode algébrique qui implique une image de fond, qui est calculée en effectuant une moyenne des pixels sur plusieurs images initiales. on applique par la suite une soustraction entre cette résultante et l'image actuelle pour obtenir l'image de différence. Les pixels de l'image de différence au-dessus d'un seuil prédéfini sont considérés comme changements.

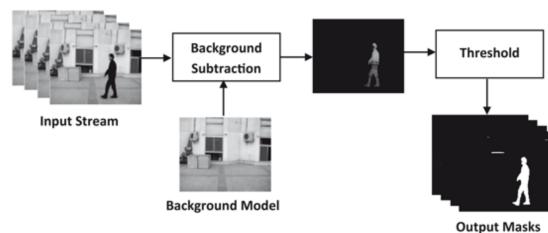


FIGURE 2.2 – Opération de soustraction de fond.

2.2.2 Méthodes basées sur la transformation

Les méthodes basées sur la transformation reposent principalement sur la transformation des images de manière à accentuer les changements et à supprimer les différences apparentes des zones inchangées. Certaines de ces méthodes, telles que l'ACP (Analyse en Composantes Principales), peuvent également être catégorisées comme des techniques d'apprentissage machine classiques. Ces techniques sont capables de mettre en évidence les informations pertinentes. Cependant, elles dépendent toujours de la sélection appropriée d'un threshold pour détecter les changements, et l'interprétation des zones modifiées peut être plus difficile et complexe lors de l'utilisation des images transformées.

2.2.3 Méthodes basées sur la classification

Les méthodes basées sur la classification utilisent des algorithmes de classification pour identifier les différentes classes d'objets présentes dans les images. La classification est ensuite comparée entre deux ou plusieurs images pour détecter les changements qui ont eu lieu. Cette méthode est plus complexe que la méthode de différence d'images, mais elle est également plus précise[20].

i Classification non supervisée

L'approche de classification non supervisée est une méthode de détection de changement qui ne nécessite pas de connaissances préalables des classes d'objets présentes dans les images. elle consiste à sélectionner des groupes de pixels similaires spectralement et regroupe l'image T1 en clusters primaires, puis étiquette les groupes similaires (spectralement) dans l'image T2 en clusters primaires, et enfin détecte et identifie les changements et produit les résultats correspondants.

ii Méthodes supervisée

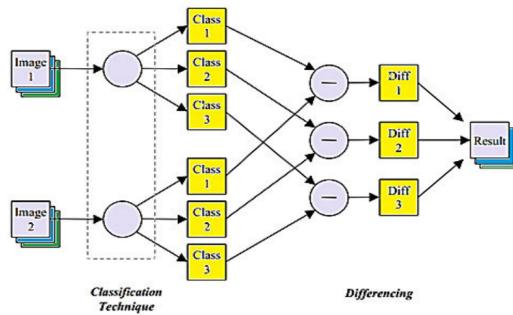


FIGURE 2.3 – Diagramme de comparaison post classification (supervised method).

La comparaison post-classification est l'approche supervisée la plus utilisée dans l'analyse de la détection des changements. Elle implique la classification indépendante de plusieurs images, suivie d'une comparaison des cartes thématiques pour identifier les zones de changement. Cette méthode présente des avantages tels que la réduction des différences de capteur et d'atmosphère, ainsi que la fourniture d'une matrice complète de changement de couverture terrestre. Cependant, les résultats dépendent de la précision des images de classification individuelles et peuvent être affectés par les différences spectrales entre les images utilisées[21].

iii Classification semi-supervisée (hybride)

La méthode semi-supervisée ou hybride est une approche qui combine les méthodes supervisées et non supervisées. Elle utilise des informations à la fois de l'ensemble de données étiquetées et non étiquetées pour entraîner un modèle de classification.

conclusion

Au cours de ce chapitre, nous avons examiné différentes méthodes de détection de changements, y compris les approches basiques telles que les méthodes algébriques (comme la soustraction et la soustraction de fond), les méthodes basées sur les transformations et les méthodes basées sur la classification. Ces méthodes ont été largement utilisées dans le passé pour détecter les changements dans les images et les données. Cependant, ces techniques traditionnelles rencontrent des difficultés face à l'explosion des volumes de données disponibles aujourd'hui.

La différence d'images, par exemple, est une technique qui compare deux images pixel par pixel pour identifier les changements. Cette approche peut être sensible aux variations d'éclairage, de perspective, et d'autres facteurs environnementaux, ce qui peut entraîner des erreurs de détection. De même, les méthodes de classification traditionnelles, peuvent être inefficaces face à des volumes de données spectrales élevées. Ces méthodes peuvent avoir du mal à gérer la haute dimensionnalité et la complexité de ces données, ce qui peut entraîner une performance de classification médiocre. Face à ces défis, l'apprentissage approfondi offre une solution prometteuse. Dans le prochain chapitre, nous explorerons plus en détail l'apprentissage approfondi et les réseaux de neurones.

CHAPITRE 3

APPRENTISSAGE PROFOND EN DETECTION DE CHANGEMENTS

L'apprentissage profond est une branche de l'intelligence artificielle qui concerne la construction et l'étude des systèmes informatiques capables d'apprendre à partir de données. De tels systèmes sont capables d'améliorer leurs performances en apprenant de nouvelles observations [22]. Dans ce chapitre nous allons commencer par reprendre les bases de l'apprentissage profond en le définissant et en parlant des différents types d'apprentissage automatique qui existent, puis nous définirons les réseaux de neurones et leurs rôles dans la détection de changements avant de conclure sur une comparaison entre certaines architectures courantes.

3.1 L'apprentissage profond

Les méthodes traditionnelles d'apprentissage automatique, bien qu'efficaces dans de nombreux contextes, ont montré leurs limites face à des défis spécifiques. Par exemple, les méthodes classiques peuvent rencontrer des difficultés lorsqu'elles sont confrontées à de grands volumes de données radiométriques.

Ces méthodes peuvent ne pas être suffisamment robustes pour gérer la complexité et la variabilité inhérentes à ces données. Ces méthodes peuvent aussi avoir du mal à gérer la haute dimensionnalité et la complexité de ces données, ce qui peut entraîner une performance de classification médiocre.

C'est là qu'intervient le Deep Learning. Cette branche de l'apprentissage automatique utilise des réseaux de neurones artificiels avec de multiples couches de neurones, ce qui lui permet d'apprendre des représentations de plus en plus complexes des données. Cette capacité est particulièrement utile pour traiter des données non structurées ou de grande dimension, comme les images ou les données radiométriques. Par exemple, il existe un type de modèle de Deep Learning, qui peut apprendre à reconnaître des motifs complexes dans les images, ce qui le rend plus robuste face aux variations d'éclairage, de perspective, et d'autres facteurs environnementaux. De plus, les réseaux de neurones sont capables de gérer efficacement la haute dimensionnalité des données, ce qui peut améliorer la performance de classification. En somme, le Deep Learning offre une solution puissante et flexible aux défis posés par les méthodes traditionnelles d'apprentissage automatique. Il continue d'être un domaine de recherche actif, avec de nouvelles techniques et applications étant constamment développées pour répondre aux défis de l'analyse de données.

3.2 Les réseaux de neurones

Les réseaux de neurones sont des modèles d'apprentissage automatique inspirés par le fonctionnement du cerveau humain. Ils sont constitués de couches de neurones interconnectés, où chaque neurone est une unité de traitement qui effectue des calculs sur les données en entrée. Les connexions entre les neurones sont pondérées, ce qui permet au réseau de générer des prédictions ou des classifications sur de nouvelles données [23].

3.2.1 Définition d'un réseau de neurones

Les réseaux de neurones sont des modèles d'apprentissage automatique qui utilisent un grand nombre de neurones artificiels interconnectés pour apprendre à partir de données. Chaque neurone est connecté à d'autres neurones et possède une fonction d'activation qui transforme les signaux d'entrée en signaux de sortie. L'ensemble des connexions entre les neurones forme un graphe pondéré appelé réseau de neurones.

Les réseaux de neurones peuvent être utilisés pour de nombreuses tâches, telles que la classification, la segmentation, la détection d'objet, la génération de texte, etc. De nombreux types de réseaux de neurones ont été proposés, chacun étant conçu pour résoudre des problèmes spécifiques.

3.2.2 Le Perceptron

Le perceptron est le réseau de neurones le plus simple, constitué d'une seule couche de neurones. Il est souvent utilisé pour des problèmes de classification binaire.

Le Perceptron est un algorithme d'apprentissage supervisé de classification binaire inventé par Frank Rosenblatt en 1957 [24]. Il est considéré comme l'un des plus anciens types de réseaux de neurones artificiels. Le Perceptron est composé d'une couche d'entrée, d'une couche de sortie et d'un seul neurone de sortie.

Le but du Perceptron est de trouver une frontière de décision linéaire pour séparer les données en deux classes. Cela est réalisé en utilisant une fonction de seuil (step function) qui produit une sortie binaire en fonction de la somme pondérée des entrées. Si la somme pondérée est supérieure à un seuil prédéfini, le neurone de sortie envoie un signal 1, sinon il envoie un signal 0.

La formule de sortie du Perceptron est définie comme suit :

$$y = f\left(\sum_{i=1}^n w_i x_i + b\right) \quad (3.1)$$

où "y" est la sortie binaire, "f" est la fonction de seuil (par exemple, la fonction échelon de Heaviside), " w_i " est le poids de l'entrée " x_i " et "b" est le biais (ou seuil) du neurone.

La fonction de seuil du Perceptron ne permet pas de représenter des frontières de décision non linéaires. Pour résoudre ce problème, des réseaux de neurones plus complexes, comme les réseaux de neurones à couches multiples, ont été développés. Cependant, malgré ses limites, le Perceptron reste un élément clé de l'histoire des réseaux de neurones artificiels et est toujours étudié et utilisé aujourd'hui.

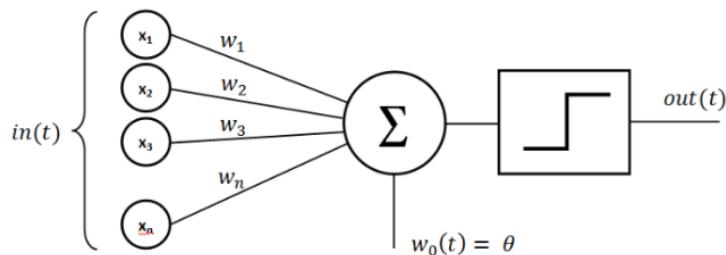


FIGURE 3.1 – modèle d'un neurone formel.

3.2.3 Le Multi layer perceptron

Le Multi-Layer Perceptron (MLP) est un type de réseau de neurones artificiels (RNA) largement utilisé dans diverses applications d'apprentissage automatique, y compris la classification et la régression. Il a été introduit pour la première fois par Paul Werbos en 1974 [25], mais son utilisation a été popularisée dans les années 1980 grâce à la théorie de l'apprentissage profond.

Le MLP est composé de plusieurs couches de neurones interconnectés, chacune étant une fonction non linéaire d'une combinaison linéaire des entrées de la couche précédente. La première couche est la couche d'entrée, qui reçoit les données d'entrée et les transmet à la couche suivante. La dernière couche est la couche de sortie, qui produit les prédictions du modèle[22].

Chaque couche intermédiaire est appelée couche cachée et peut avoir un nombre arbitraire de neurones. Les poids entre les neurones de chaque couche sont ajustés par l'algorithme de rétropropagation de gradient lors de l'apprentissage du modèle.

La formule pour calculer la sortie de chaque neurone dans une couche intermédiaire est la suivante :

$$a_j = f \left(\sum_{i=1}^n w_{ij} x_i + b_j \right) \quad (3.2)$$

où a_j est la sortie du neurone j , w_{ij} est le poids entre les neurones i et j , x_i est l'entrée du neurone i , b_j est le biais du neurone j et f est une fonction d'activation non linéaire, telle que la fonction sigmoïde ou la fonction ReLU.

Le MLP est capable de modéliser des relations non linéaires complexes entre les entrées et les sorties, ce qui en fait un choix populaire pour de nombreuses applications d'apprentissage automatique. Cependant, il peut souffrir de surapprentissage si le nombre de couches cachées est trop élevé ou si le nombre de neurones dans chaque couche est trop grand par rapport à la taille de l'ensemble de données.

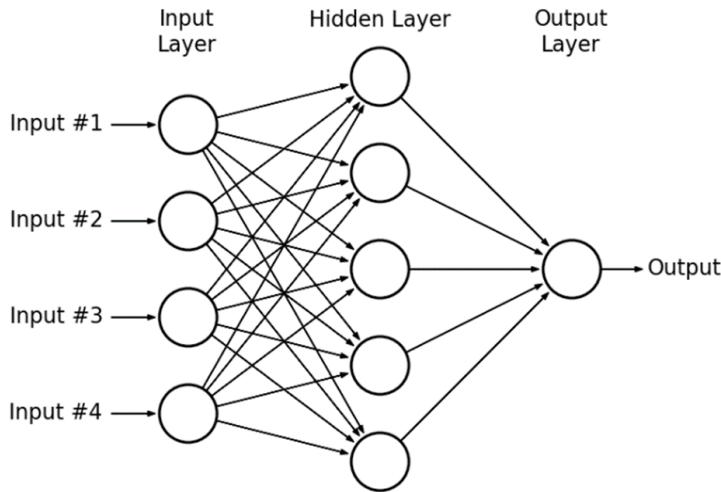


FIGURE 3.2 – exemple d'une architecture d'un multi perceptron.

3.2.4 Les types de réseaux de neurones

i Les réseaux de neurones récurrents (RNN)

Les réseaux de neurones récurrents (RNN) sont une classe de réseaux de neurones artificiels spécialement conçus pour traiter les données séquentielles et temporelles, tels que les signaux audio, le texte et les séquences de mouvements dans les vidéos. Contrairement aux réseaux de neurones feedforward, les RNN possèdent des connexions récurrentes qui leur permettent de maintenir une mémoire interne de l'information passée, ce qui les rend particulièrement adaptés pour capturer les dépendances à long terme dans les données séquentielles [26]. Les RNN ont été largement utilisés dans diverses applications, notamment la reconnaissance de la parole, la traduction automatique, la génération de texte et la prédiction de séries temporelles [23].

Cependant, les RNN traditionnels souffrent de problèmes d'apprentissage à long terme, tels que la disparition et l'explosion du gradient, qui limitent leur capacité à capturer des dépendances à long terme . Pour résoudre ces problèmes, des variantes de RNN, telles que les Long Short-Term Memory (LSTM) , ont été développées. Ces architectures utilisent des mécanismes de portes pour contrôler le flux d'information à travers les cellules récurrentes, permettant ainsi une meilleure conservation de l'information à long terme.

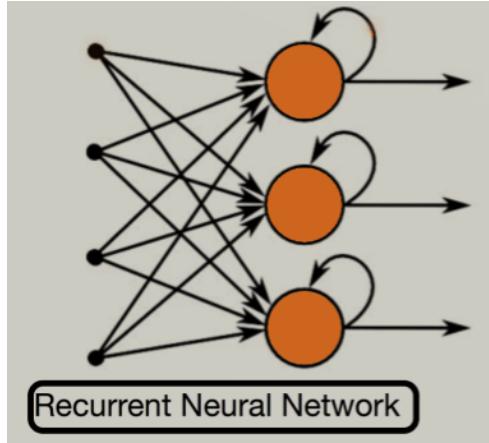


FIGURE 3.3 – Architecture classique d'un réseau de neurones convolutif (Paul BLANC-DURAND 2018).

ii Les GANs (Generative Adversarial Networks)

Les GANs, ou Generative Adversarial Networks, sont un type de réseau de neurones qui permet de générer de nouvelles données à partir d'un jeu de données existant. Ils ont été introduits pour la première fois par Ian Goodfellow et al en 2014[27].

Le fonctionnement d'un GAN est basé sur l'interaction entre deux réseaux de neurones : le générateur et le discriminateur. Le générateur a pour but de créer de nouvelles données qui ressemblent à celles du jeu de données existant, tandis que le discriminateur a pour but de différencier les données réelles des données générées par le générateur[28].

Le générateur prend en entrée un vecteur de bruit aléatoire et le transforme en une donnée synthétique. Cette donnée synthétique est ensuite soumise au discriminateur, qui la compare avec une donnée réelle et détermine si elle est vraie ou fausse. Le générateur est ensuite ajusté en fonction de la réponse du discriminateur, de sorte qu'il puisse créer des données synthétiques de plus en plus réalistes.

Les GANs ont été utilisés dans de nombreuses applications, notamment la génération d'images, de vidéos, de musique, de texte et de modèles 3D. Ils ont également été utilisés pour la création de contenu artistique, la synthèse de données manquantes et la création de données d'entraînement supplémentaires pour les modèles d'apprentissage automatique.

Le fonctionnement mathématique des GANs est basé sur la théorie des jeux et la minimisation de la divergence de Kullback-Leibler. La fonction de perte du générateur est basée sur la divergence de Jensen-Shannon, tandis que la fonction de perte du discriminateur est basée sur la cross-entropy.

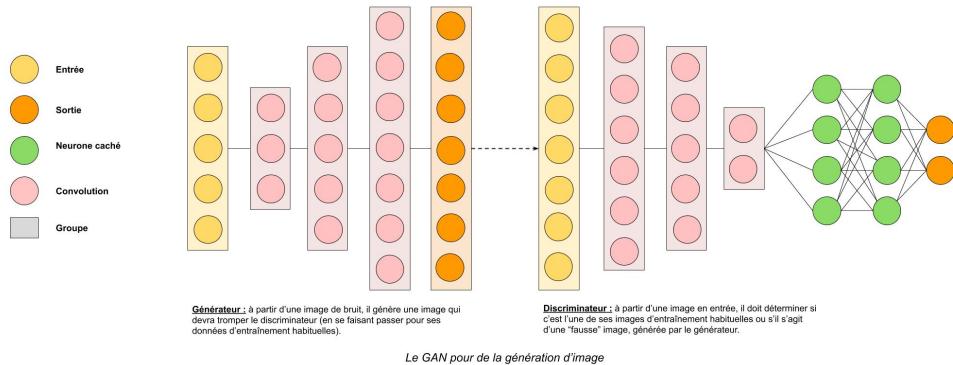


FIGURE 3.4 – Architecture d'un GAN.

iii Les réseaux de neurones convolutionnels (CNN)

Les réseaux de neurones convolutifs (Convolutional Neural Networks ou CNNs) sont une catégorie de réseaux de neurones qui ont révolutionné le domaine de la vision par ordinateur. Contrairement aux réseaux de neurones classiques, les CNNs exploitent la structure spatiale des images et des données 2D, ce qui en fait des modèles très efficaces pour la classification et la segmentation d'images.

Les CNNs ont été introduits pour la première fois en 1989 par Yann LeCun et ses collègues dans leur célèbre article intitulé "Backpropagation Applied to Handwritten Zip Code Recognition" [29]. Ce réseau, appelé LeNet-5, était initialement utilisé pour reconnaître les chiffres écrits à la main sur des enveloppes.

Le fonctionnement d'un CNN est basé sur des opérations de convolution, de pooling et de classification. Les opérations de convolution consistent à appliquer un filtre de petite taille à l'image en entrée pour extraire des caractéristiques locales. Les opérations de pooling réduisent la taille de la carte des caractéristiques en prenant la valeur maximale ou moyenne dans des régions voisines. Les couches de classification comprennent des neurones qui effectuent des opérations linéaires et non linéaires sur les caractéristiques pour produire une sortie de classe.

La popularité des CNN a connu une croissance exponentielle en 2012 lorsque l'équipe de recherche dirigée par Geoffrey Hinton a remporté le défi ImageNet en utilisant un CNN profond baptisé AlexNet. Depuis lors, les CNN ont été largement adoptés et ont démontré leur efficacité dans de nombreuses tâches de vision par ordinateur, telles que la classification d'images, la segmentation sémantique, la détection d'objets et la reconnaissance de texte dans les images, entre autres.

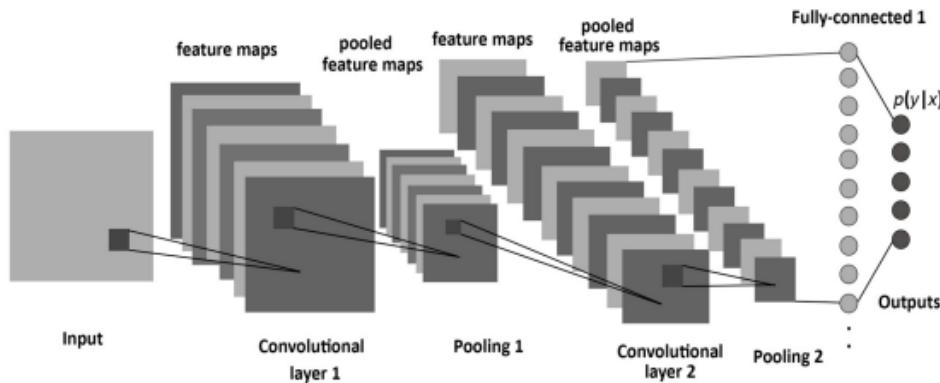


FIGURE 3.5 – Architecture classique d'un réseau de neurones convolutif (Paul BLANC-DURAND 2018).

3.3 réseaux de neurones convolutifs

Les réseaux de neurones convolutifs (CNN) sont largement utilisés dans la détection de changement, une tâche qui consiste à détecter les différences entre deux images prises à des moments différents. Les CNN permettent de capturer et d'analyser les caractéristiques visuelles spécifiques qui représentent les changements dans les images. En utilisant des couches de convolution et d'autres techniques d'apprentissage, les CNN sont capables d'apprendre à distinguer les motifs et les structures qui indiquent un changement significatif entre les images, ce qui facilite la détection de changement automatique [30].

3.3.1 Les couches des CNN

i Couche de convolution

Les couches de convolution appliquent des filtres à l'image d'entrée pour extraire des caractéristiques locales. Chaque filtre est déplacé sur l'image en effectuant des multiplications et des sommes pondérées pour calculer une nouvelle carte de caractéristiques. Les filtres de convolution permettent de détecter des motifs spécifiques dans l'image, tels que des bords, des textures ou des structures plus complexes [30].

$$y_{i,j} = f \left(\sum_{m=-k}^k \sum_{n=-k}^k W_{m+k, n+k} \cdot x_{i+m, j+n} + b \right) \quad (3.3)$$

Dans cette équation : $y_{i,j}$ est la sortie de la convolution à la position (i, j) .

f est la fonction d'activation non linéaire appliquée à chaque élément de la convolution (par exemple, ReLU, sigmoïde, tangente hyperbolique, etc.).

$W_{m+k, n+k}$ est le poids du filtre de convolution à la position $(m+k, n+k)$.

$x_{i+m, j+n}$ est l'entrée à la position $(i+m, j+n)$.

b est le biais de la convolution.

k est la taille du filtre de convolution (par exemple, pour un filtre 3x3, $k = 1$).

0	1	1	1	0	0	0
0	0	1	1	1	0	0
0	0	0	1	1	1	0
0	0	0	1	1	0	0
0	0	1	1	0	0	0
0	1	1	0	0	0	0
1	1	0	0	0	0	0

I

1	0	1
0	1	0
1	0	1

K

$$=$$

1	4	3	4	1
1	2	4	3	3
1	2	3	4	1
1	3	3	1	1
3	3	1	1	0

$I * K$

FIGURE 3.6 – Exemple de convolution. [1]

ii Couche de pooling

Les couches de pooling réduisent la taille spatiale des caractéristiques extraites. Elles réalisent cela en regroupant les valeurs voisines et en ne conservant que les valeurs les plus importantes par exemple la couche MaxPool récupère le maximum des valeurs présentes dans le noyau, ou encore la couche AvgPool qui en fait la moyenne. Le pooling permet de réduire la quantité de calculs nécessaires et rend le modèle invariant aux petites translations et déformations dans l'image[30]. L'équation de la couche maxpool :

$$P(i, j) = \max(I(2i + m, 2j + n)) \quad (3.4)$$

$P(i, j)$ est la valeur de la carte de caractéristiques après la couche de pooling.

I est la carte de caractéristiques en entrée.

m, n sont les indices de la taille de la fenêtre de pooling.

12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

$\xrightarrow{2 \times 2 \text{ Max-Pool}}$

20	30
112	37

FIGURE 3.7 – Exemple de maxpooling. [2]

iii Couche de entièrement connectée

Les couches entièrement connectées se trouvent généralement à la fin du CNN et sont responsables de la classification ou de la régression des caractéristiques extraites. Chaque neurone dans ces couches est connecté à tous les neurones de la couche précédente. Ces couches finales utilisent les caractéristiques apprises pour prendre une décision finale, comme la classification d'une image en différentes classes prédéfinies.

$$y_i = f \left(\sum_{j=1}^n W_{ij} \cdot x_j + b_i \right) \quad (3.5)$$

Dans cette équation : y_i est la sortie du i -ème neurone de la couche entièrement connectée.

f est la fonction d'activation non linéaire appliquée à chaque neurone (par exemple, ReLU, sigmoïde, tangente hyperbolique, etc.).

W_{ij} est le poids de la connexion entre le j -ème neurone de la couche précédente et le i -ème neurone de la couche entièrement connectée.

x_j est la sortie du j -ème neurone de la couche précédent

b_i est le biais du i -ème neurone de la couche entièrement connectée.

n est le nombre total de neurones dans la couche précédente. Cette équation est appliquée pour chaque neurone i dans la couche entièrement connectée.

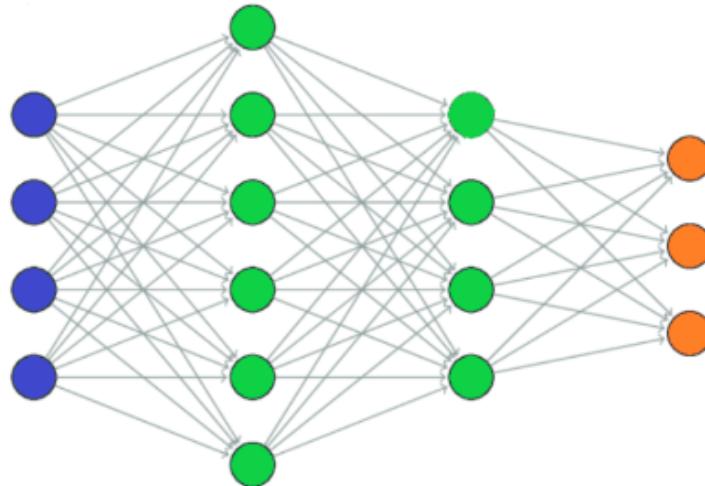


FIGURE 3.8 – Un réseau de neurones avec des couches entierement connectées.

Les différentes couches des CNN travaillent ensemble pour extraire les caractéristiques pertinentes, réduire la dimensionnalité et prendre des décisions basées sur ces caractéristiques. Cette architecture en couches permet aux CNN d'apprendre des représentations hié-

rarchiques des données, en capturant des motifs de plus en plus abstraits à mesure que l'information se propage à travers les couches. Cela rend les CNN particulièrement adaptés à la détection de changement et à d'autres tâches de vision par ordinateur [30].

iv Les fonctions d'activation

La fonction ReLU Il s'agit de la fonction la plus utilisée grâce à sa simplicité et sa performance[31], elle retourne simplement x si x est positif, et 0 si x est négatif son équation est :

$$f(x) = \max(0, x) \quad (3.6)$$

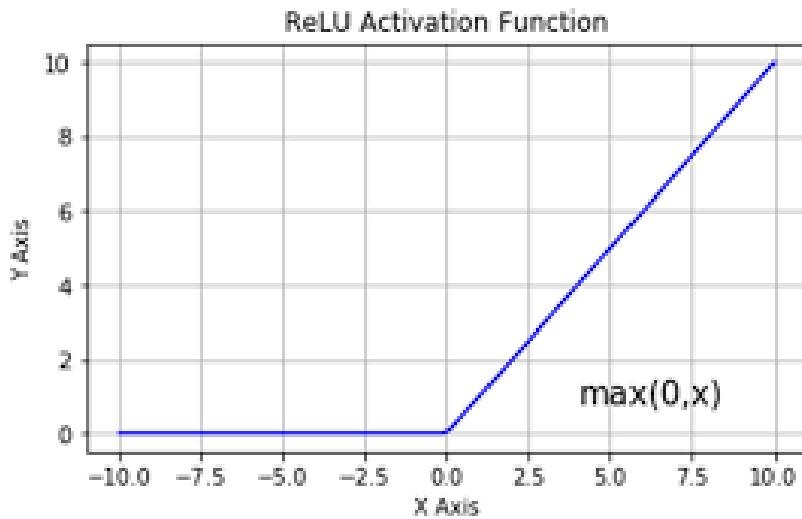


FIGURE 3.9 – Courbe de la fonction d'activation ReLU

La fonction sigmoïde La fonction d'activation sigmoïde est une fonction très utilisée aussi en deep-learning, elle permet de réduire l'intervalle de sorte à $[0, 1]$ [31] afin de traiter les valeurs comme des probabilités. Son équation est :

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3.7)$$

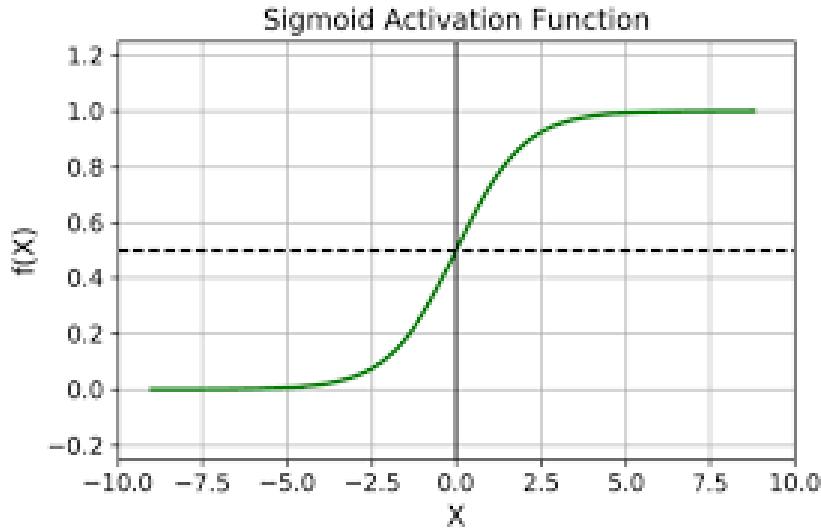


FIGURE 3.10 – Courbe de la fonction d'activation sigmoïde

La fonction tangente hyperbolique Tanh La fonction *Tanh* elle aussi populaire dans le domaine du deep-learning nous permet de réduire l'intervalle de sortie à $[-1, 1]$ [31], son équation est :

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3.8)$$

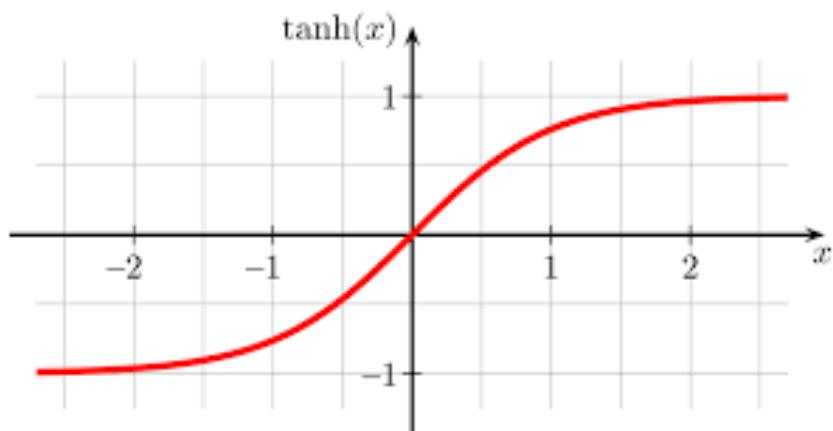


FIGURE 3.11 – Courbe de la fonction d'activation Tanh

3.4 Exemples d'architectures CNN

3.4.1 LeNet

LeNet, également connu sous le nom de LeNet-5, est l'un des premiers modèles de réseau neuronal convolutif (CNN) développés par Yann LeCun et ses collègues à la fin des années 1990 [32]. Il a été conçu pour la reconnaissance de caractères manuscrits et a posé les bases des architectures de CNN qui ont suivi.

i Architecture

LeNet-5 est composé de plusieurs couches interconnectées, chacune avec un rôle spécifique dans le processus de reconnaissance de caractères. Voici une vue d'ensemble de l'architecture de LeNet-5 :

- Couche d'entrée : Le modèle prend en entrée une image de taille fixe (par exemple, 32x32 pixels pour le dataset MNIST). Cette couche est suivie d'une opération de convolution et d'une fonction d'activation.
- Couches de convolution : LeNet-5 utilise deux couches de convolution avec des noyaux de petite taille pour extraire les caractéristiques de l'image. Chaque couche de convolution est suivie d'une fonction d'activation non linéaire, généralement une sigmoïde ou une tangente hyperbolique.
- Couches entièrement connectées : Les sorties des couches de pooling sont aplatis et connectées à une ou plusieurs couches entièrement connectées. Ces couches finales sont responsables de la classification des caractères manuscrits en différentes classes.

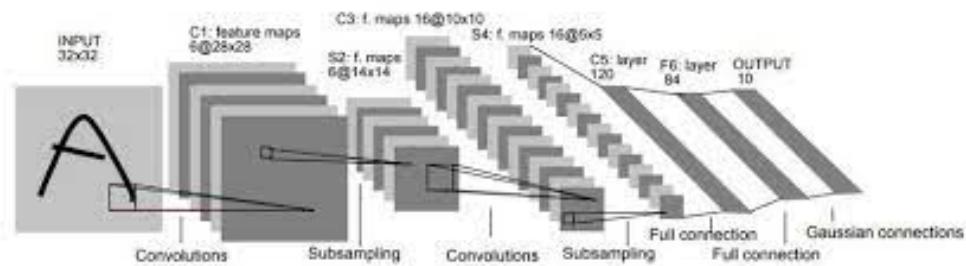


FIGURE 3.12 – Architecture LeNet-5(LeCun et al 1998).

ii Caractéristiques

- Conception simple : LeNet-5 est caractérisé par une conception simple et une architecture relativement peu profonde par rapport aux modèles modernes (7 couches). Ce-

pendant, il a été démontré qu'il était efficace dans des tâches spécifiques telles que la reconnaissance de caractères manuscrits.

- Utilisation de convolutions et de pooling : LeNet-5 utilise des convolutions pour extraire les caractéristiques de l'image et des opérations de pooling pour réduire la dimensionnalité et rendre le modèle plus invariant aux translations.
- Activation non linéaire : Le modèle utilise la fonction tangente hyperbolique pour introduire de la non-linéarité dans le réseau et capturer des motifs complexes dans les images.
- Entraînement par rétropropagation du gradient : LeNet-5 est entraîné à l'aide de l'algorithme de rétropropagation du gradient, qui permet d'ajuster les poids du réseau pour minimiser la fonction de perte et améliorer les performances de classification.
- Utilisation de la base de données MNIST : LeNet-5 a été développé et évalué sur le dataset MNIST, qui est un ensemble de données largement utilisé pour la reconnaissance de chiffres manuscrits. Il a établi des performances de pointe sur cette tâche.

LeNet-5 a ouvert la voie aux développements ultérieurs des CNN et a jeté les bases des architectures modernes. Bien qu'il soit relativement simple par rapport aux modèles actuels, LeNet-5 a démontré l'efficacité des réseaux de neurones convolutifs pour la reconnaissance d'images et a inspiré de nombreuses avancées dans le domaine de l'apprentissage profond.

3.4.2 AlexNet

AlexNet est un réseau de neurone convolutif (CNN) introduit en 2012 par Alex Krizhevsky et al. Il a marqué une avancée significative dans le domaine de la vision par ordinateur [33].

i Architecture

AlexNet comprend huit couches principales, dont cinq couches de convolution et trois couches entièrement connectées. La structure générale de l'architecture est la suivante :

- Couche 1 : Convolution avec 96 filtres de taille 11x11, suivie d'une fonction d'activation ReLU et d'une couche de pooling avec une fenêtre de taille 3x3 et un pas de 2.
- Couche 2 : Convolution avec 256 filtres de taille 5x5, fonction d'activation ReLU et couche de pooling .
- Couche 3 : Convolution avec 384 filtres de taille 3x3, fonction d'activation ReLU .
- Couche 4 : Convolution avec 384 filtres de taille 3x3, fonction d'activation ReLU .

- Couche 5 : Convolution avec 256 filtres de taille 3x3, fonction d'activation ReLU, suivie d'une couche de pooling .
- Couche 6 : Fully connected avec 4096 neurones et fonction d'activation ReLU. Cette couche est suivie d'une couche de dropout pour réduire le surapprentissage.
- Couche 7 : Fully connected avec 4096 neurones et fonction d'activation ReLU. Également suivie d'une couche de dropout.
- Couche 8 : Fully connected avec le nombre de neurones correspondant au nombre de classes de sortie (par exemple, 1000 pour ImageNet), suivi d'une fonction d'activation softmax pour obtenir les probabilités de chaque classe .

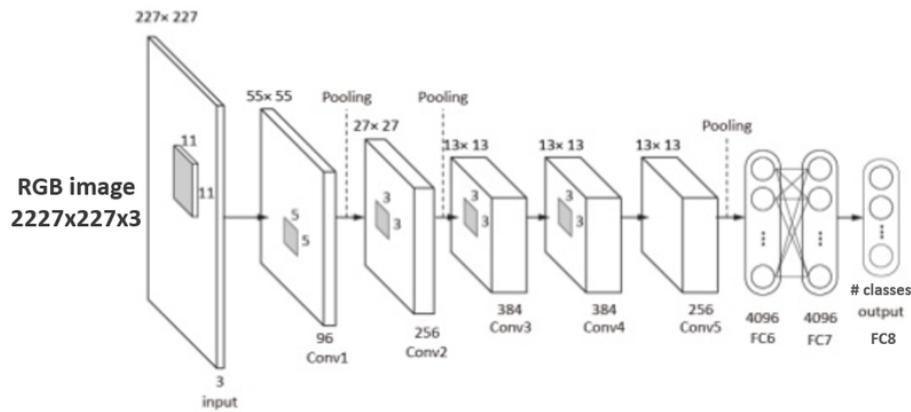


FIGURE 3.13 – Architecture du ALEXNET (Researchgate.net).

ii Caractéristiques

AlexNet se distingue par les caractéristiques suivantes :

- Utilisation de la fonction d'activation ReLU : AlexNet a été l'un des premiers CNN à utiliser la fonction d'activation ReLU (Rectified Linear Unit) au lieu de la fonction d'activation sigmoïde ou tangente hyperbolique . La fonction ReLU aide à résoudre le problème de la disparition du gradient et accélère la convergence de l'apprentissage.
- Utilisation de l'augmentation des données : AlexNet utilise des techniques d'augmentation des données telles que le recadrage aléatoire, la rotation et le miroir horizontal pour augmenter la taille du jeu de données d'entraînement et réduire le surapprentissage.
- Utilisation du dropout : Le dropout est utilisé dans les couches fully connected pour régulariser le modèle et réduire le surapprentissage . Il désactive aléatoirement certains

neurones lors de l’entraînement, forçant le réseau à apprendre des représentations plus robustes et indépendantes .

- Utilisation de la descente de gradient stochastique (SGD) avec moment : AlexNet utilise l’optimiseur SGD avec moment pour mettre à jour les poids du réseau pendant l’entraînement . Cela aide à accélérer la convergence de l’apprentissage en tenant compte de l’historique des gradients .

3.4.3 VGG16

VGG16 est un modèle de réseau neuronal convolutif (CNN) développé par Karen Simonyan et Andrew Zisserman en 2014 [34]. Il a été conçu pour la classification d’images et a obtenu de très bonnes performances lors du défi ImageNet 2014. VGG16 est caractérisé par son architecture simple et profonde, qui utilise des convolutions de petite taille et des couches entièrement connectées pour extraire des caractéristiques complexes des images.

i Architecture

L’architecture de VGG16 est composée de plusieurs couches de convolution, de pooling et de couches entièrement connectées. Voici une vue d’ensemble de l’architecture de VGG16 :

- Couches de convolution : VGG16 utilise 13 couches de convolution avec des noyaux de taille 3x3 pour extraire les caractéristiques de l’image. Chaque couche de convolution est suivie d’une fonction d’activation ReLU.
- Couches de pooling : Après certaines couches de convolution, il y a une couche de pooling (max pooling) qui réduit la dimension spatiale de la sortie et extrait les caractéristiques les plus importantes.
- Couches entièrement connectées : Les sorties des couches de pooling sont aplatis et connectées à trois couches entièrement connectées. Les deux premières couches entièrement connectées ont 4096 neurones chacune, tandis que la dernière couche a autant de neurones que de classes dans le problème de classification.

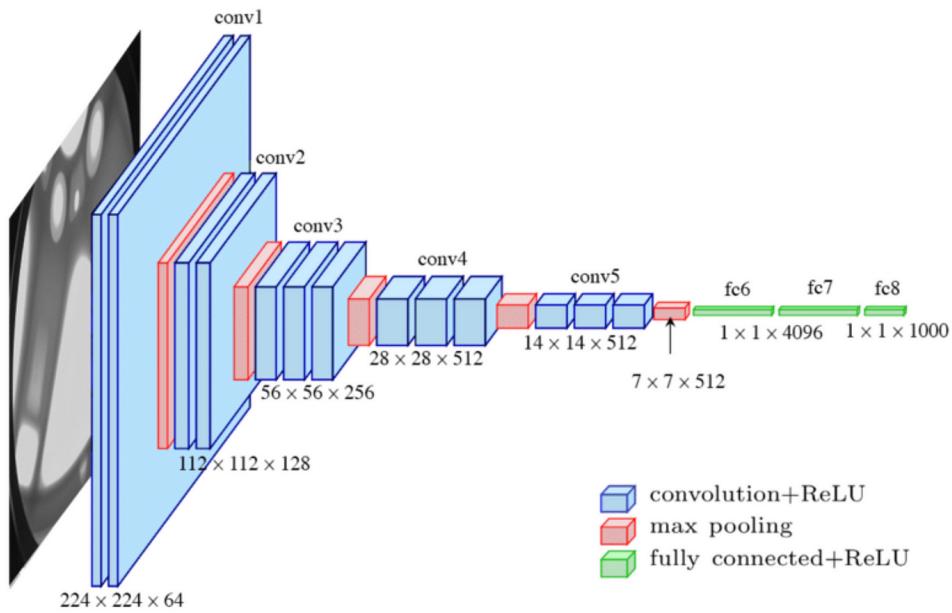


FIGURE 3.14 – Architecture du VGG16 (Researchgate.net).

ii Caractéristiques

VGG16 se distingue par les caractéristiques suivantes :

- Conception simple et profonde : VGG16 est caractérisé par une conception simple et profonde, avec 16 couches pondérées. Il utilise des convolutions de petite taille pour capturer des caractéristiques locales et des couches entièrement connectées pour combiner ces caractéristiques en représentations globales.
- Utilisation de la fonction d'activation ReLU : VGG16 utilise la fonction d'activation ReLU pour introduire de la non-linéarité dans le réseau et accélérer la convergence de l'apprentissage par rapport aux fonctions d'activation sigmoïde ou tanh.
- Performances élevées : VGG16 a obtenu de très bonnes performances lors du défi ImageNet 2014, démontrant son efficacité pour la classification d'images [34].

3.4.4 ResNet

ResNet (Residual Neural Network) est une architecture de réseau neuronal convolutif (CNN) qui a été introduite en 2015 par Kaiming He et al. [35]. L'une des principales caractéristiques de ResNet est l'utilisation de blocs résiduels qui permettent un apprentissage en profondeur plus efficace en évitant le problème de la disparition du gradient.

i Architecture

ResNet est composé de plusieurs blocs résiduels qui facilitent l'apprentissage en profondeur. L'architecture de base, appelée ResNet-50, comporte 50 couches principales. Voici une vue d'ensemble de l'architecture de ResNet-50 :

- Couche d'entrée : Une seule couche de convolution suivie d'une fonction d'activation ReLU et d'une opération de pooling.
- Blocs résiduels : ResNet-50 comprend 4 blocs résiduels, chacun composé de plusieurs couches de convolution. Chaque bloc utilise des connexions résiduelles pour sauter des couches et ajouter les sorties de ces couches à la sortie finale du bloc.
- Couches entièrement connectées : Après les blocs résiduels, il y a des couches entièrement connectées qui réduisent progressivement la dimensionnalité de la sortie et génèrent les probabilités de classe.

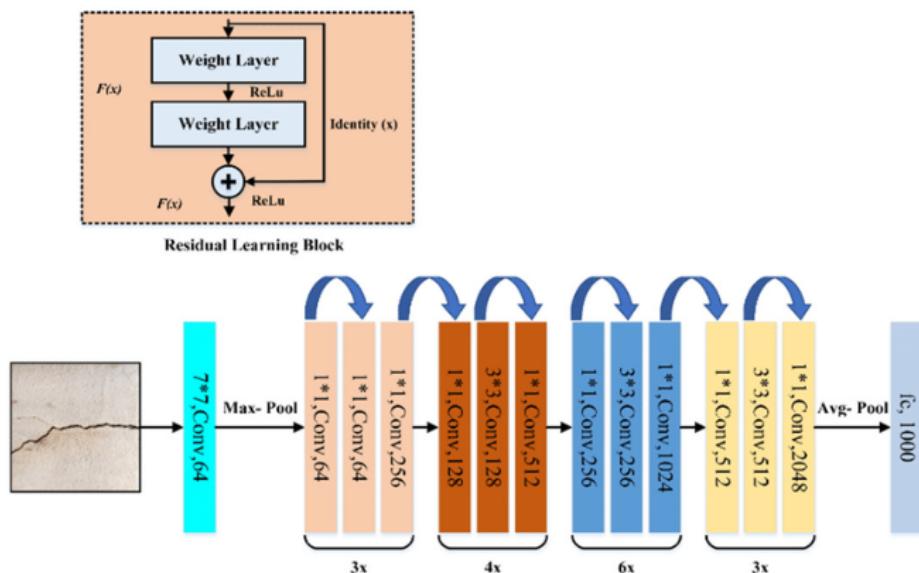


FIGURE 3.15 – Architecture du resnet50 (Luqman Ali et Fady Shibata Alnajjar et al 2021).

ii Caractéristiques

- Blocs résiduels : Les blocs résiduels sont la caractéristique principale de ResNet. Ils permettent un apprentissage en profondeur en ajoutant des connexions résiduelles qui sautent des couches. Cela permet de préserver les informations précédemment apprises et facilite la propagation du gradient.
- Architecture en profondeur : ResNet a été conçu pour être très profond, atteignant jusqu'à 152 couches dans certaines variantes. L'apprentissage en profondeur permet de

capturer des caractéristiques complexes et abstraites des images, ce qui améliore les performances de classification.

- Utilisation de la fonction d'activation ReLU : Comme de nombreux autres CNN, ResNet utilise la fonction d'activation ReLU pour introduire de la non-linéarité dans le réseau et résoudre le problème de la disparition du gradient.
- Utilisation du pré-entraînement et du transfert learning : ResNet bénéficie de l'utilisation du pré-entraînement et du transfert learning. Les modèles pré-entraînés sur de grands ensembles de données, tels que ImageNet, peuvent être utilisés comme point de départ pour des tâches de classification ou de détection spécifiques, permettant d'obtenir de bonnes performances avec moins de données d'entraînement.

ResNet a été une avancée majeure dans les CNN en permettant l'apprentissage en profondeur et en obtenant des performances exceptionnelles dans de nombreux domaines de la vision par ordinateur, notamment la classification d'images, la détection d'objets et la segmentation sémantique.

3.4.5 Xception :

Xception, acronyme pour "Extreme Inception", est un réseau de neurones convolutif profond qui a été proposé par François Chollet[36], dans un article de recherche en 2017 intitulé "Xception : Deep Learning with Depthwise Separable Convolutions". Il s'agit d'une extension de l'architecture Inception, qui a été introduite par Google. Xception partage plusieurs similarités avec l'architecture Inception. Tout d'abord, comme Inception, Xception est un réseau de neurones convolutif profond conçu pour la classification d'images.

Dans un second temps , il s'appuie sur l'hypothèse d'Inception, qui postule que les cartes de caractéristiques spatiales et de canaux dans les couches convolutives peuvent être traitées séparément. Cependant, Xception va plus loin en adoptant cette hypothèse de manière plus rigoureuse. Alors que Inception mélange encore les caractéristiques spatiales et de canaux à travers ses convolutions, Xception propose de les séparer complètement. Cette approche est réalisée par l'utilisation de convolutions séparables en profondeur, qui sont le cœur de l'architecture Xception. Malgré le même nombre de paramètres que Inception V3, Xception surpassé ce dernier en termes de performance sur de grands jeux de données grâce à une utilisation plus efficace des paramètres du modèle.

i Architecture :

L'architecture Xception se compose de 36 couches de convolution qui forment la base d'extraction des caractéristiques du réseau. Ces couches de convolution sont structurées en 14 modules, avec des connexions résiduelles linéaires autour de chacun d'eux, à l'exception du premier et du dernier module. En résumé, l'architecture Xception est une pile linéaire de couches de convolution séparables en profondeur avec des connexions résiduelles. Cette approche permet de traiter de manière indépendante les corrélations entre les canaux et les corrélations spatiales, ce qui rend l'architecture Xception très flexible et facile à définir et à modifier.

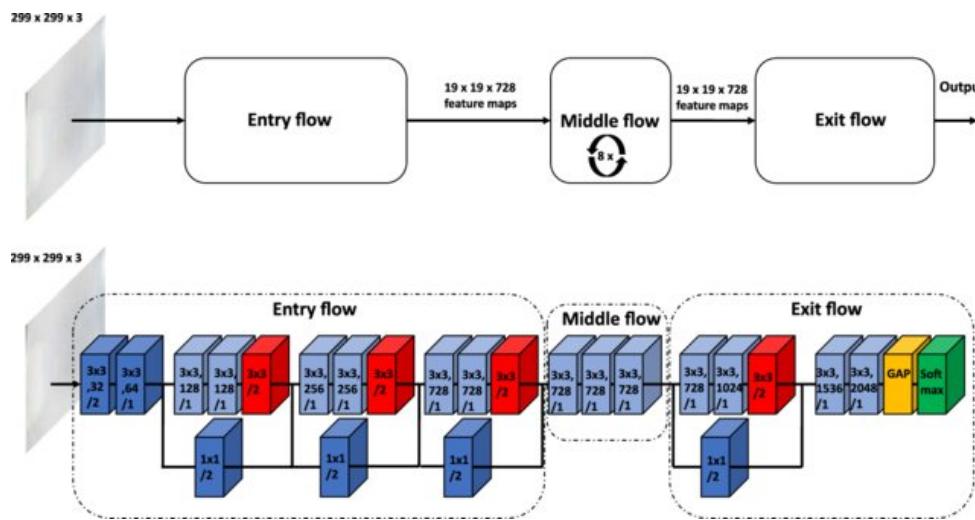


FIGURE 3.16 – Architecture du réseau Xception.

ii Caractéristiques :

La couche de convolution séparable en profondeur (Depthwise Separable Convolutions) est une technique utilisée dans l'architecture Xception pour traiter les corrélations spatiales et celles des canaux de manière plus efficace. Contrairement à une convolution traditionnelle qui effectue des opérations sur tous les canaux d'entrée simultanément, la convolution séparable en profondeur divise le processus en deux étapes distinctes. La première étape consiste en une convolution en profondeur (depthwise convolution) où chaque canal d'entrée est traité individuellement par un noyau de convolution spécifique à ce canal. Cela permet de capturer les corrélations spatiales dans chaque canal de manière indépendante. En séparant les opérations sur les canaux, cette étape réduit considérablement le nombre de paramètres à apprendre, ce qui permet d'économiser des ressources computationnelles.

La deuxième étape est une convolution point par point (pointwise convolution) qui combine les informations spatiales extraites précédemment en une représentation finale. Cette

convolution point par point utilise un noyau de taille 1x1 pour effectuer une convolution sur l'ensemble des canaux d'entrée. Elle permet de capturer les corrélations entre les canaux et de fusionner les informations spatiales extraites précédemment.

3.5 Les types d'apprentissage :

3.5.1 Entrainement de zero

Pour qu'un réseau neuronal convolutif (CNN) puisse accomplir des tâches spécifiques avec une précision optimale, il est impératif qu'il soit entraîné sur un ensemble de données pertinentes. Le processus d'entraînement d'un CNN implique l'exposition du réseau à un ensemble de données annotées, où les entrées (c'est-à-dire les images) sont liées à des étiquettes correspondantes (classes ou catégories). L'expression "from scratch" désigne la méthode qui consiste à initialiser les poids du CNN de manière aléatoire, puis à le former intégralement sur un nouvel ensemble de données. Cette approche peut être privilégiée lorsque les données sont spécifiques à un domaine particulier. Cependant, la formation "from scratch" présente certains désavantages. Premièrement, elle nécessite souvent un ensemble de données large et représentatif pour produire des résultats satisfaisants. Deuxièmement, elle peut s'avérer très coûteuse en termes de temps de calcul et de ressources requises, car le réseau doit apprendre toutes les caractéristiques à partir de zéro.

3.5.2 Le transfert learning

Dans cette partie, on va explorer le concept de "Transfert Learning", une méthode d'apprentissage profond qui utilise un modèle pré-entraîné comme base pour un autre modèle d'apprentissage surtout qu'un entraînement d'un CNN peut être très couteux en terme ressource. Cette approche est particulièrement bénéfique lorsque les données disponibles pour l'apprentissage sont limitées.

i Definition :

Le transfert d'apprentissage est une technique d'apprentissage profond qui utilise les connaissances acquises lors des entraînements sur des tâches précédentes pour améliorer les performances sur une nouvelle tâche. Au lieu de commencer l'apprentissage à partir de zéro, le transfert learning tire parti des représentations apprises par un modèle pré-entraîné sur une tâche similaire ou liée. Cela permet d'économiser du temps et des ressources en utilisant les connaissances déjà obtenues pour accélérer l'apprentissage sur notre nouvelle tâche[37].

ii Catégories de Transfert Learning :

Le Transfert Learning peut être classé en trois types principaux : inductif, transductif et non supervisé [38].

Transfert Learning inductif : Dans le cadre de l'apprentissage par transfert, nous nous confrontons à deux tâches distinctes : la tâche source et la tâche cible. Les algorithmes d'apprentissage sont conçus pour capitaliser sur les connaissances préalablement acquises à partir de la tâche source, dans le but d'optimiser les performances lors de l'exécution de la tâche cible. Ainsi, le domaine source sert de fondement pour améliorer l'efficacité dans le domaine cible [39].

Transfert Learning transductif : les tâches à accomplir dans les domaines source et cible sont comparables voire similaires, mais les domaines en eux-mêmes varient, que ce soit en termes de données ou de distributions de probabilités marginales.

Transfert Learning non supervisé : dans ce cadre on se retrouve dans une situation qui ressemble beaucoup à celle du transfert inductif. Cependant, l'accent est mis ici sur les tâches non supervisées dans le domaine cible. Bien que les domaines source et cible soient similaires, les tâches à accomplir sont différentes. Un défi particulier dans ce contexte est que nous ne disposons d'aucune donnée étiquetée, ni dans le domaine source, ni dans le domaine cible.

iii Techniques de Transfert Learning :

Il existe plusieurs stratégies et techniques de transfert d'apprentissage, parmi lesquelles : **fine tuning**

Le fine tuning est une technique de transfer learning qui consiste à réutiliser un modèle pré-entraîné et à le ré-entraîner sur un jeu de données plus petit et spécifique à une tâche donnée. Cela permet d'obtenir de meilleures performances qu'en entraînant un modèle à partir de zéro. Afin de bien utiliser le fine tuning il faut bien choisir le modèle pré-entraîné adapté à la tâche et pendant l'entraînement et aussi ne pas entraîner les premières couches qui capturent des caractéristiques génériques, mais se baser sur les dernières couches spécifiques à la tâche. (geler les couches inférieures) [40].

Extraction de caractéristiques :

Cette approche consiste à utiliser un modèle pré-entraîné comme un extracteur de caractéristiques. Les sorties de certaines couches du modèle sont utilisées comme entrées pour un nouveau modèle [41].

Dans le contexte des CNN, les caractéristiques sont des motifs visuels tels que les bords, les textures, les formes, les couleurs, etc. Les couches de convolution dans les CNN appliquent

des filtres pour détecter ces caractéristiques dans les images. Les couches de pooling réduisent la dimensionnalité en extrayant les caractéristiques les plus importantes, et pour finir, les fonctions d’activations viennent améliorer les performances du CNN en apprenant les caractéristiques abstraites à travers des transformations linéaires ayant pour but [19] :

- Ajouter une courbure non linéaire au paysage d’optimisation pour améliorer la convergence de l’apprentissage du réseau.
- Ne pas augmenter de manière significative la complexité de calcul du modèle.
- Ne pas entraver le flux de gradient pendant l’apprentissage.
- Conserver la distribution des données pour faciliter un meilleur apprentissage du réseau.

En utilisant des couches convolutives empilées, les CNN sont capables d’apprendre des caractéristiques de plus en plus abstraites et complexes, ce qui en fait des extracteurs de caractéristiques puissants pour diverses tâches de télé-détection.

3.5.3 Etude Comparative

Chris Kawatsu et al [42], dans leurs étude nous fournissent en 2017 ce graphe de comparaison entre les différents extracteurs de caractéristiques CNN qui nous aidera donc à les départager en fonction nombre de leurs nombre paramètres ainsi que leurs précision après entraînement sur le dataset ImageNet.

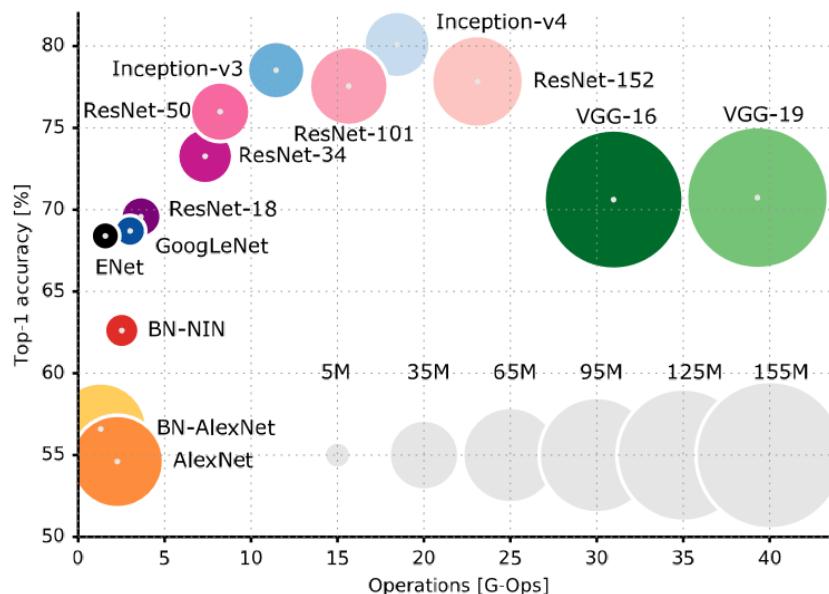


FIGURE 3.17 – Comparaison des CNN les plus populaires.(Chris Kawatsu et al, 2017)

i Précision sur le dataset ImageNet :

VGG16 a une précision supérieure à 70% sur le dataset ImageNet, ce qui est supérieur à celle de ResNet-18, AlexNet et LeNet. Cela signifie que VGG16 est capable de classer correctement les images avec une plus grande précision, ce qui est un facteur clé dans de nombreuses applications de vision par ordinateur.

ii Architecture :

L'architecture de VGG16 est plus simple et plus uniforme que celle de ResNet-18 et AlexNet. Cela rend VGG16 plus facile à comprendre et à implémenter tout en restant capable de capturer des caractéristiques complexes.

iii Nombre de paramètres :

VGG16 est le réseau de cette liste qui possède le plus de paramètres, ce qui peut augmenter sa consommation de mémoire et de puissance de calcul cependant cela signifie que le modèle a une capacité d'apprentissage accrue, ce qui peut lui permettre de mieux généraliser à partir de nouvelles données.

iv Transfert d'apprentissage :

Les poids pré entraînés du VGG16 sur la base de données ImageNet servent de point de départ à beaucoup de réseaux de neurones, de plus son intégration dans les bibliothèques de plusieurs langages de programmation le rendent plus accessible. En conclusion, bien que VGG16 ait un nombre de paramètres élevé, ses avantages en termes de précision, de simplicité d'architecture et de précision en font un choix solide pour de nombreuses applications de vision par ordinateur.

v Profondeur de l'architecture :

Le Xception est le réseau le plus profond entre les CNN cités ce qui le rend capable d'apprendre des représentations plus complexes et plus abstraites des données. Cependant, l'entraînement de réseaux plus profonds peut être plus difficile.

3.6 Conclusion

Après une étude approfondie de l'apprentissage profond et des réseaux de neurones, en particulier les CNN, nous pouvons affirmer que ces techniques constituent des outils puissants pour la détection de changements et l'analyse d'images en général. Les CNN, en particulier, se sont avérés être d'excellents extracteurs de caractéristiques, capables de capturer des informations complexes et hiérarchiques à partir de données d'image.

En comparant différents modèles, nous avons constaté que le VGG16, bien qu'étant moins profond, offre une précision remarquable, ce qui le rend particulièrement utile pour l'avancement de la détection de changements. Cependant, le modèle Xception, avec sa structure plus profonde et son approche innovante de la convolution séparable en profondeur, a démontré une performance supérieure à celle de nombreux autres modèles.

Dans notre recherche, nous allons utiliser ces deux modèles pour une comparaison directe. Bien que différents dans leur conception et leur profondeur, le VGG16 et le Xception sont tous deux des outils précieux dans le domaine de l'analyse d'images.

CHAPITRE 4

CONCEPTION ET IMPLEMENTATION DES METHODES

Introduction

La détection de changements dans les images de télédétection à haute résolution est un domaine de recherche important en télédétection, avec des applications dans l'urbanisme, la surveillance environnementale et la gestion des catastrophes. Les méthodes traditionnelles de détection de changements, telles que la soustraction d'images et la classification supervisée dont on a parlé précédemment, ont montré des limites en termes de précision et de robustesse face aux variations d'échelle, aux changements d'illumination et aux différences de résolution. Récemment, les approches basées sur l'apprentissage profond ont été proposées pour améliorer les performances de détection de changements dans les images de télédétection. Plusieurs recherches récentes comme celle de (Mengxi Liu et al, 2021) [43], (C. Zhang et al, 2020)[44] , (J. Chen et al, 2020) [45] ont vu le jour et ont proposé de nouvelles méthodes basée sur l'apprentissage profond.

Dans ce chapitre, nous présentons le contexte et la motivation de notre recherche, inspirée des nouvelles méthodes et qui vise a concevoir une nouvelle architecture mais aussi d'implémenter un autre modèle auquel on apportera des modification en évaluant leurs performances qu'on comparera au final.

4.1 Première méthode :

4.1.1 Le modèle basé sur Xception :

Dans cette première méthode on récupère le modèle Xception dont on a parlé un peu plus haut avec son architecture complète qui se compose de 36 couches, ces couches sont divisées sur 3 flows :

- Entry flow : Le flux d'entrée est responsable de la première partie du réseau et vise à extraire les caractéristiques de bas niveau des images en entrée. Il comprend une séquence de couches de convolution, de normalisation par lots et de fonctions d'activation. Cette partie du réseau permet de capturer les détails et les contours des images.
- Middle flow : Le flux intermédiaire est la partie centrale du réseau et est responsable de l'extraction des caractéristiques de plus haut niveau. Il est composé de plusieurs modules Inception, qui sont des blocs de convolution avec des connexions résiduelles. Ces modules permettent de capturer des caractéristiques plus complexes et abstraites des images. Le passage par ses modules se fait 8 fois.
- End flow : Le flux de sortie est la dernière partie du réseau et est responsable de la classification finale. Il comprend une combinaison de couches de pooling global moyen et de couches entièrement connectées. Le pooling global moyen agrège les caractéristiques spatiales de l'image pour obtenir une représentation globale, tandis que les couches entièrement connectées effectuent la classification finale en assignant des probabilités aux différentes classes.

Nous passons à l'adaptation du modèle original et à son fine tuning sur la base de données d'intérêt comme suit :

1. Chargement du modèle pré-entraîné sur la base de données IMageNet.
2. Modification de la couche de classification : remplacement de la couche de classification finale du modèle Xception par une nouvelle couche de classification adaptée à notre tâche de détection de changement tout en prenant en compte le nombre de classes des cartes de changement.
3. Congélation des couches : congélation de toutes les couches du modèle Xception sauf la nouvelle couche de classification que nous avons ajoutée. Cela empêchera les poids pré-entraînés d'être modifiés lors du fine tuning.
4. Fine tuning du modèle : entraîner le modèle en utilisant l'ensemble de données d'intérêt en utilisant comme méthode d'optimisation la descente de gradient stochastique, tout en ajustant les hyperparamètres du modèle.

5. Validation du modèle : évaluation des performances du modèle en utilisant un ensemble de données de validation.
6. Test et évaluation : une fois que le modèle est entraîné, on utilise de nouvelles images de notre base de données pour prédire les changements tout en les comparant avec les masques de changements obtenus par soustraction des deux images.

Avec cette technique de fine-tuning, nous exploitons les connaissances préalables du modèle Xception tout en l'adaptant à la tâche spécifique de détection de changement. A noter que jusqu'à présent, aucune étude n'a été réalisée en utilisant Xception sur ce dataset spécifique, on vérifiera les résultats plus bas.

4.1.2 Algorithme du Xception :

Algorithm 1 Fine tuning du réseau Xception

Input : Une paire d'images t_1 , t_2 et le masque de changement y , Taux d'apprentissage lr , Nombre d'epochs N , Taille du batch B , Loss function *Binary cross entropy*, Fonction d'activation *Sigmoid function*

```
for m = 1 to N do
    Fine tuning du modèle par :
    Passage des images  $t_1$  et  $t_2$  dans le Module Xception
    Utilisation de Blocs Residuels
    Application Global AveragePooling
     $Loss \leftarrow \text{Binary cross entropy}$ 
    Propagation en arrière pour ajuster les poids.
end
Output : Carte de changement  $Dist$ 
```

4.2 Deuxième méthode :

4.2.1 Extraction de caractéristiques

Nous proposons d'utiliser une architecture basée sur deux réseaux de neurones convolutifs (CNN) pré-entraînés, plus précisément, un modèle VGG16 pré-entraîné avec les poids provenant de l'ensemble de données ImageNet.

Cette approche offre plusieurs avantages, notamment une meilleure convergence du modèle grâce au fine tuning. L'architecture de l'extracteur de caractéristiques dans notre proposition se compose de 5 blocs distincts. Chaque bloc se compose d'une série de couches de convolution avec des noyaux de taille 3x3 est utilisée pour obtenir des caractéristiques de bas niveau à partir des images fournies. Ensuite, une fonction d'activation ReLU est mise en œuvre pour améliorer la qualité des caractéristiques obtenues. Afin d'élargir le champ récepteur et de conserver les détails spatiaux importants, une couche de max-pooling avec un pas de 2 est employée pour diminuer de moitié la taille des caractéristiques. Dans le processus d'extraction de caractéristiques, les images IT1 et IT2 servent d'entrées pour l'extracteur de caractéristiques. Cet extracteur traite les deux images séparément et génère un ensemble de vecteurs de caractéristiques multi échelle pour chacune d'elles. À la fin de ce processus, on obtient les groupes de vecteurs de caractéristiques $F_m = \text{feat1}_m, \dots, \text{feat4}_m$, où m représente T1 pour les caractéristiques extraites de l'image IT1 et T2 pour celles extraites de l'image IT2.

4.2.2 Amélioration de la discriminations des caractéristiques

Dans notre processus, une fois que nous avons extrait les caractéristiques des images, notre objectif est de les rendre encore plus discriminantes. Pour cela, nous utilisons un module appelé CBAM, qui améliore la capacité des caractéristiques à différencier les images les unes des autres. Ensuite, nous construisons une carte de distances en utilisant la mesure de distance euclidienne. Cette mesure est couramment utilisée dans de nombreux domaines, y compris la télédétection, en raison de son équilibre entre performance [46] et facilité d'implémentation[47]. Une faible distance entre les caractéristiques suggère une forte similarité, tandis qu'une distance élevée indique une dissimilarité importante.

Convolutional Block Attention Module (CBAM) Le Module d'Attention de Bloc Convolutif (CBAM) constitue une méthode avant-gardiste visant à optimiser les performances des réseaux de neurones convolutifs en s'inspirant de certaines caractéristiques du système visuel humain, notamment en se concentrant sur les éléments saillants d'une scène pour saisir la structure visuelle. Le CBAM cherche à reproduire cette capacité en introduisant deux

mécanismes d'attention distincts : channel attention et spatial attention. [48]

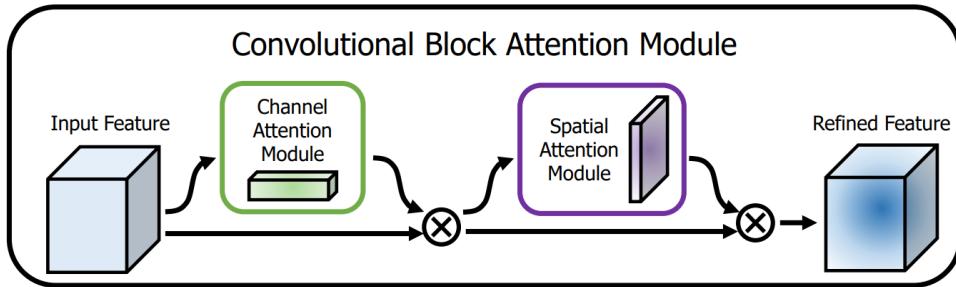


FIGURE 4.1 – Module CBAM

Channel attention Le module channel attention se concentre sur la détermination de "ce qui" est significatif dans une image donnée. Pour ce faire, le CBAM utilise un sous-module de canaux qui exploite à la fois les sorties de max-pooling et d'average-pooling avec un réseau partagé. Ce réseau partagé est composé d'un perceptron multicouche (MLP) avec une couche cachée. Afin de réduire la surcharge des paramètres, la taille de l'activation cachée est définie à $\mathbb{R}^{C/r \times 1 \times 1}$, où r est le ratio de réduction.

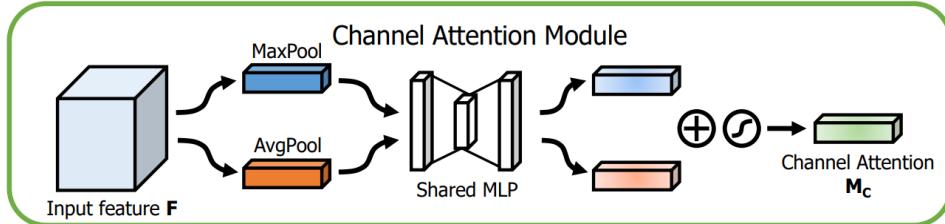


FIGURE 4.2 – Channel attention du CBAM.

Les opérations de max-pooling et d'average-pooling génèrent deux descripteurs de contexte spatial différents : F_c avg et F_c max. Ces descripteurs sont ensuite transmis au réseau partagé pour produire la carte du channel attention $M_c \in \mathbb{R}^{C \times 1 \times 1}$. Après l'application du réseau partagé à chaque descripteur, les vecteurs de caractéristiques de sortie sont fusionnés en utilisant une sommation élément par élément :

$$\begin{aligned} M_c(F) &= \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \\ &= \sigma(W_1(W_0(F_{c_{avg}})) + W_1(W_0(F_{c_{max}}))) \end{aligned} \tag{4.1}$$

Spatial attention Le spatial attention se concentre sur la détermination de "où" se trouve une partie informative de l'image. Pour ce faire, le CBAM utilise des opérations de pooling pour mettre en évidence les régions informatives. Plus précisément, il applique des opérations d'average-pooling et de max-pooling le long de l'axe des canaux et les concatène pour générer un descripteur de caractéristiques efficace.

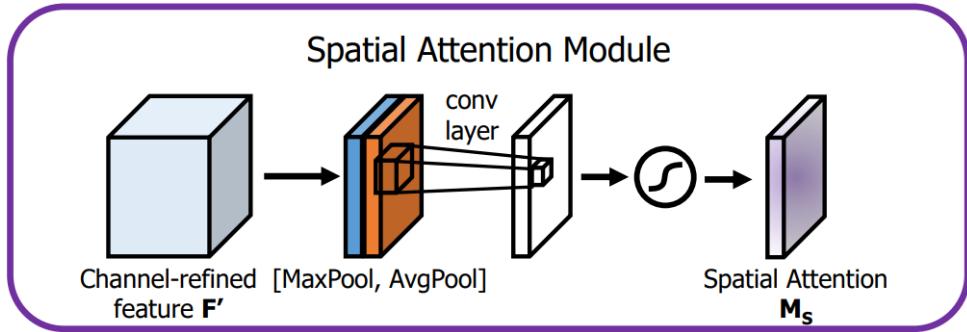


FIGURE 4.3 – Spatial attention du CBAM

Le CBAM agrège les informations de canaux d'une carte de caractéristiques en utilisant deux opérations de pooling, générant ainsi deux cartes 2D : $F_s^{\text{avg}} \in \mathbb{R}^{1 \times H \times W}$ et $F_s^{\text{max}} \in \mathbb{R}^{1 \times H \times W}$. Ces cartes sont ensuite concaténées et convoluées par une couche de convolution standard, produisant la carte du spatial attention 2D :

$$\begin{aligned} Ms(F) &= \sigma(f^{7 \times 7}([AvgPool(F); MaxPool(F)])) \\ &= \sigma(f^{7 \times 7}([F_{s_{\text{avg}}}; F_{s_{\text{max}}}])) \end{aligned} \quad (4.2)$$

Intégration du channel et spatial attention Le CBAM combine le channel et spatial attention en appliquant séquentiellement les deux mécanismes d'attention à une carte de caractéristiques intermédiaire $F \in \mathbb{R}^{C \times H \times W}$. Les valeurs du channels attention sont diffusées le long de la dimension spatiale, et vice versa :

$$\begin{aligned} F' &= Mc(F) \otimes F \\ F'' &= Ms(F') \otimes F' \end{aligned} \quad (4.3)$$

En intégrant ces mécanismes d'attention, le CBAM permet aux réseaux de neurones convolutifs de mieux capturer la structure visuelle des images, améliorant ainsi leur performance globale.

Batch Contrastive Loss La Batch Contrastive Loss (BCL Loss) est une fonction de perte qui joue un rôle crucial dans l'amélioration de la performance des réseaux de neurones convolutifs, en particulier lorsqu'elle est combinée avec le Convolutional Block Attention Module (CBAM). Qian Shi et al. [ref] ont étudié l'application de la BCL Loss dans le contexte des réseaux Siamese, où elle est utilisée pour évaluer la similarité entre les cartes de distance et les masques de changements [49]

$$LBCL(Xn, Yn) = \sum_{i,j=0}^2 (1 - y_{i,j})d_{i,j}^2 + y_{i,j} \max(d_{i,j} - m)^2 \quad (4.4)$$

Selon cette formule, la perte des paires de pixels inchangées ($y = 0$) ou les paires de pixels modifiées ($y = 1$) dépendent de la distance entre les paires de pixels $d(i, j)$. Cette approche permet d'exprimer efficacement le degré de correspondance entre les échantillons appariés, ce qui est essentiel pour obtenir une extraction précise des changements.

4.2.3 Amélioration de l'apprentissage

Dans notre approche, nous utilisons un module de Deep Supervision pour améliorer l'apprentissage de notre modèle. Ce module génère deux cartes de distances intermédiaires en évaluant les différences entre les premières caractéristiques extraites de chaque image. Ensuite, nous comparons ces cartes de distances intermédiaires à l'aide d'une fonction de perte basée sur le coefficient de Dice (Dice Loss) par rapport aux labels de référence. Cette méthode permet d'optimiser la capacité du réseau à apprendre des caractéristiques pertinentes et à pondérer les régions importantes de l'image, ce qui améliore la performance globale du modèle.

Architecture Dans un premier temps, nous mettons en œuvre deux couches Deep Supervision (DS) qui comprennent chacune deux opérations de déconvolution suivies d'une couche sigmoïde. L'objectif de cette architecture est d'augmenter la résolution de l'image à sa taille d'origine grâce aux deux déconvolutions successives utilisant un noyau de taille 3x3. Par la suite, la couche sigmoïde est employée pour pondérer la distribution de probabilité de chaque pixel. Cette approche nous permet d'obtenir deux cartes de changements intermédiaires, qui sont ensuite comparées individuellement aux labels de référence. L'appréciation de ces cartes en comparaison avec les étiquettes est effectuée en employant la fonction de perte Dice Loss. Cette technique contribue à l'amélioration des performances du modèle en se concentrant sur les zones cruciales de l'image et en optimisant l'acquisition des attributs significatifs. Pour finir l'erreur totale sera égale à la somme des coûts des deux images.[43]

Dice Loss La Dice Loss constitue une fonction de coût fondée sur le coefficient de Dice. [50] La formule du coefficient de Dice s'exprime comme suit :

$$\text{Dice} = \frac{2 \sum_i (f_i \cdot g_i)}{\sum_i f_i^2 + \sum_i g_i^2} \quad (4.5)$$

où f_i et g_i représentent respectivement les éléments des prédictions du modèle et des masques de changements. L'objectif consiste à maximiser le coefficient de Dice, qui varie entre 0 et 1. Ainsi, la Dice Loss est définie par la relation suivante :

$$\text{Dice Loss} = 1 - \text{Dice} \quad (4.6)$$

En optimisant cette fonction de coût, on cherche à accroître la similarité entre les prédictions du modèle et les masques de changements, en maximisant le coefficient de Dice.

4.2.4 Architecture globale du modèle proposé

L'architecture du modèle proposé est composée de trois parties principales. La première partie est l'extracteur de caractéristiques. Il s'agit d'un élément crucial de notre architecture, car il est responsable de la création de vecteurs de caractéristiques à partir des images en entrée. Pour ce faire, nous utilisons deux réseaux VGG-16 siamois pour capturer les caractéristiques à partir des images bi-temporelles. Ces réseaux sont conçus pour extraire des caractéristiques à différentes échelles, ce qui nous permet de capturer une variété de détails dans les images. La deuxième partie de notre architecture est le module d'attention, qui est conçu pour rendre les caractéristiques extraites plus discriminantes. Pour ce faire, nous utilisons le module CBAM . Ce module se concentre sur les aspects spatiaux et les canaux des caractéristiques, ce qui nous permet d'améliorer la discrimination des caractéristiques extraites. En d'autres termes, il nous aide à nous concentrer sur les détails importants et à ignorer les informations moins pertinentes. La troisième et dernière partie de notre architecture est le module de supervision profonde. Ce module utilise les cartes de caractéristiques intermédiaires pour guider l'apprentissage du réseau. Plus précisément, il calcule une perte de Dice à partir de ces cartes intermédiaires, qui est ensuite combinée avec la BCL Loss que nous avons étudiée plus tôt. Cette combinaison de Dice Loss et de BCL Loss constitue notre erreur totale, qui est utilisée pour entraîner le réseau. En combinant ces trois parties, notre architecture est capable de détecter efficacement les changements dans les images bi-temporelles. Elle utilise des caractéristiques multi-échelles, une attention discriminative et une supervision profonde pour obtenir des résultats précis et fiables.

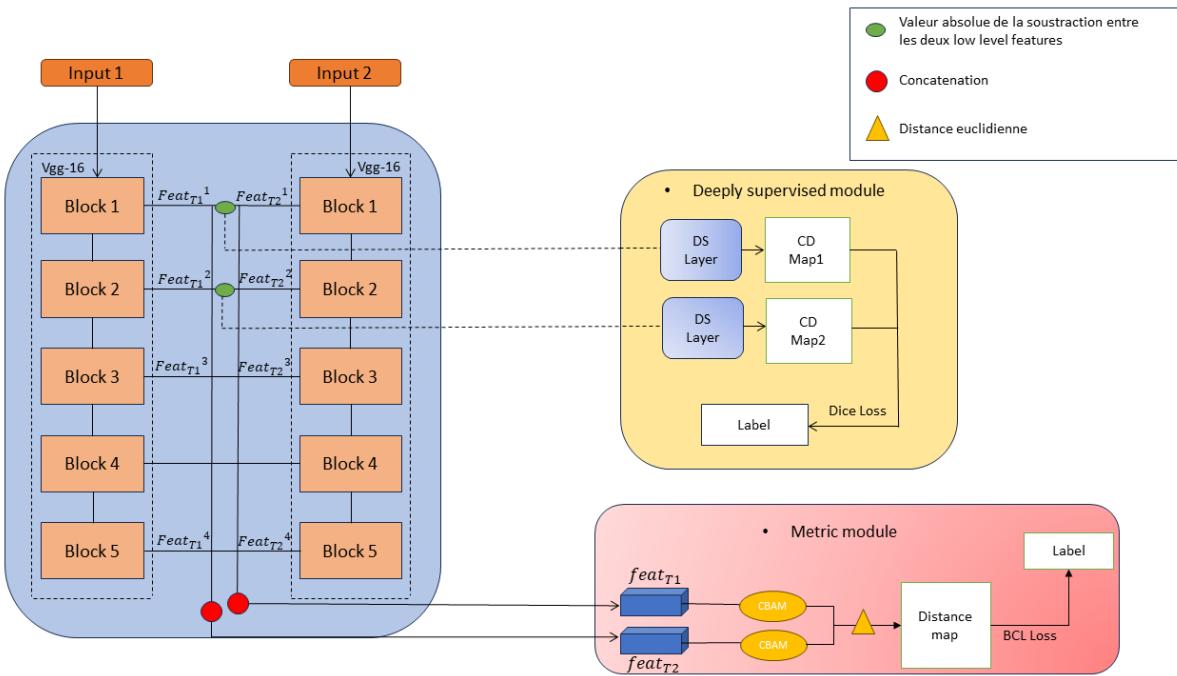


FIGURE 4.4 – Architecture du modèle proposé.

4.2.5 Algorithme du modèle proposé :

Algorithm 2 Entraînement du réseau proposé

Input : Une paire d'images t_1, t_2 et le masque de changement y , Taux d'apprentissage lr , Nombre d'epochs N , Taille du batch B , Ratio de réduction du CBAM r , Taille du noyau de convolution du CBAM k

for $m = 1$ to N **do**

Extraction du vecteur de caractéristiques $feat_{1n}$ de t_1 avec le VGG16.

Extraction du vecteur de caractéristiques $feat_{2n}$ de t_2 avec le VGG16.

$x1_n \leftarrow$ Concaténation des caractéristiques de l'image t_1

$x2_n \leftarrow$ Concaténation des caractéristiques de l'image t_2

$F''_{1t1n} \leftarrow$ CBAM($x1_n, r, k$) // Obtenir les caractéristiques rendues plus discriminantes

$F''_{2t2n} \leftarrow$ CBAM($x2_n, r, k$) //Obtenir les caractéristiques rendues plus discriminantes

Calculer la distance euclidienne entre $x1_n$ et $x2_n$ pour obtenir la carte de changements $Dist$

Comparer $Dist_n$ avec le label y_n grâce à la BCL Loss

$ds2 \leftarrow$ DsLayer(Val_Abs($feat_{1t1} - feat_{1t2}$))

Comparer $ds2$ avec le label y grâce à la DiceLoss

$ds3 \leftarrow$ DsLayer(Val_Abs($feat_{2t1} - feat_{2t2}$))

Comparer $ds3$ avec le label y grâce à la DiceLoss

$Loss \leftarrow$ Somme des deux DiceLoss + BCL Loss

Propagation en arrière pour ajuster les poids

end

Output : Carte de changement $Dist$, Carte de changement intermédiaire $ds2$, Carte de changement intermédiaire $ds3$

Au cours de notre recherche, voici l'algorithme que nous avons implémenté.

L'algorithme va boucler sur le nombre de epoch qu'on va choisir pour l'entraînement. Soit une paire d'images $IT1_n$ et $IT2_n$ et leurs masques de changement Yn (avec n le numéro du n-ième batch). Nous donnons les images en entrée à la partie extraction de caractéristique développée plus haut afin d'en extraire les vecteurs de caractéristiques $feat_{1n}$ et $feat_{2n}$.

Par la suite, ces deux vecteurs de caractéristiques seront concaténés dans les valeurs $x1n$ et $x2n$ afin d'être expédiées vers le module CBAM qui va se charger de les rendre plus discriminantes comme explicité dans le sous chapitre précédent. En sortie, le module CBAM nous donnera $F''1n$ et $F''2n$ qui seront les caractéristiques affinées.

Nous calculons la carte de changement $Dist_n$ en utilisant la distance euclidienne entre $F''1n$ et $F''2n$, puis on calcul la BCL loss de cette carte de changements avec le label Yn .

D'un autre côté, nous reprenons les valeurs de la différences entre les features des du premier niveau $feat1_{T1n}$ et $feat1_{T2n}$ afin de calculer la première carte de changements intermédiaire qu'on appellera $ds2n$, et nous reproduisons le même processus avec $feat2_{T1n}$ et $feat2_{T2n}$ pour créer la seconde carte de changements intermédiaire $ds3n$.

La prochaine étape sera de calculer les distances $ds2n$ et $ds3n$ avec le label Yn avec la Dice Loss et de les sommer ensemble. Pour finir nous calculons la perte totale qui va s'exprimer de cette manière la :

$$\text{Total Loss} = \text{Dice Loss} + \text{BCL Loss} \quad (4.7)$$

qui va être utilisée lors de la backward propagation pour ajuster les poids du réseau de neurones. Pour executer ce code, nous avons décidé d'utiliser un learning rate de 0.0005 qui est une valeur faible qui permettra au modèle d'être plus performant, de plus on a choisi 100 pour le nombre de epoch afin de pouvoir avoir des courbes détaillées après l'apprentissage. Etant équipés d'un calculateur puissant, nous nous sommes permis de choisir 128 comme batchsize.

Après le passage dans l'extracteur de caractéristiques, on ressort avec une image de taille 8x8, d'où le choix du ratio du CBAM qui est de 32 pour un noyau de convolution de taille 3x3.

CHAPITRE 5

IMPLEMENTATION ET RESULTATS

5.1 Présentation du jeu de données

5.1.1 Contexte et motivation

Au cours des dernières décennies, de nombreux efforts ont été déployés pour développer des ensembles de données ouverts pour la détection de changements (CDDs). Cependant, certains problèmes subsistent, tels que la résolution insuffisante, le manque de diversité des types de changements et des volumes de données trop petits pour les méthodes basées sur l'apprentissage profond. Lors de notre recherche de jeu de données, plusieurs datasets ont retenu notre attention et méritent à notre sens d'être présentés et comparés.

Remarque : les datasets présentés sont des datasets composés d'images aériennes, donc chaque image a trois bandes, R, G, B

i SZTAKI Dataset :

Ce dataset comprend 13 paires d'images aériennes de dimensions 952x640 et de résolution 1,5 m/pixel, accompagnées de masques binaires de changement. Les images ont été capturées avec des intervalles de temps de 5, 7 et 23 ans. [51] [52] Le dataset contient des représentations visuelles de divers types de changements, tels que :

- L'apparition de nouvelles zones construites
- La réalisation d'opérations de construction

- La plantation de grands ensembles d'arbres
- La mise en culture de terres récemment labourées
- L'exécution de travaux préparatoires en amont de la construction



FIGURE 5.1 – exmeple d'échantillon du dataset SZTAKI. (Lien du dataset ici)

ii LEVIR-CD :

LEVIR-CD est un ensemble de données à grande échelle pour la détection de changements de bâtiments par télédétection. Il comprend 637 paires d'images Google Earth de très haute résolution (0.5m/pixel) de 1024×1024 pixels, avec des intervalles de temps de 5 à 14 ans. Les images montrent des changements significatifs d'utilisation des terres, en particulier la croissance des bâtiments. [53] Le dataset couvre divers types de bâtiments et se concentre sur les changements liés aux bâtiments, qui incluent :

- Construction et demolition de résidences de type villa.
- Construction et demolition de grands immeubles d'appartements.
- Construction et demolition de petits garages et de grands entrepôts.

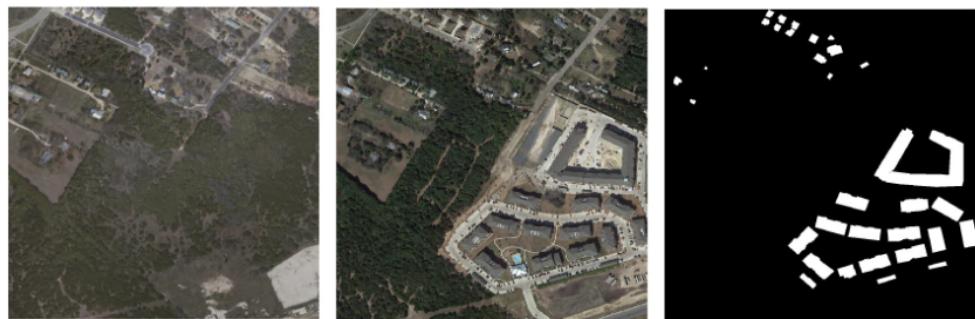


FIGURE 5.2 – exmeple d'échantillon du dataset LEVIR-CD. (Lien du dataset ici)

iii SYSU-CD :

Le dataset SYSU-CD contient 20 000 paires d'images aériennes de 0,5 mètre/pixel prises entre 2007 et 2014 à Hong Kong où la forte densité de population et la construction rapide à

Hong Kong pendant cette période ont entraîné un grand nombre de changements dans les zones urbaines et côtières qui ont été capturés. Il comprend 20.000 paires d'images de taille 256*256. [43] Ce dataset couvre des changements de type :

- Les nouveaux bâtiments urbains construits
- L'expansion des banlieues
- Les travaux préparatoires avant la construction
- Les changements de la végétation
- L'expansion des routes
- La construction en mer

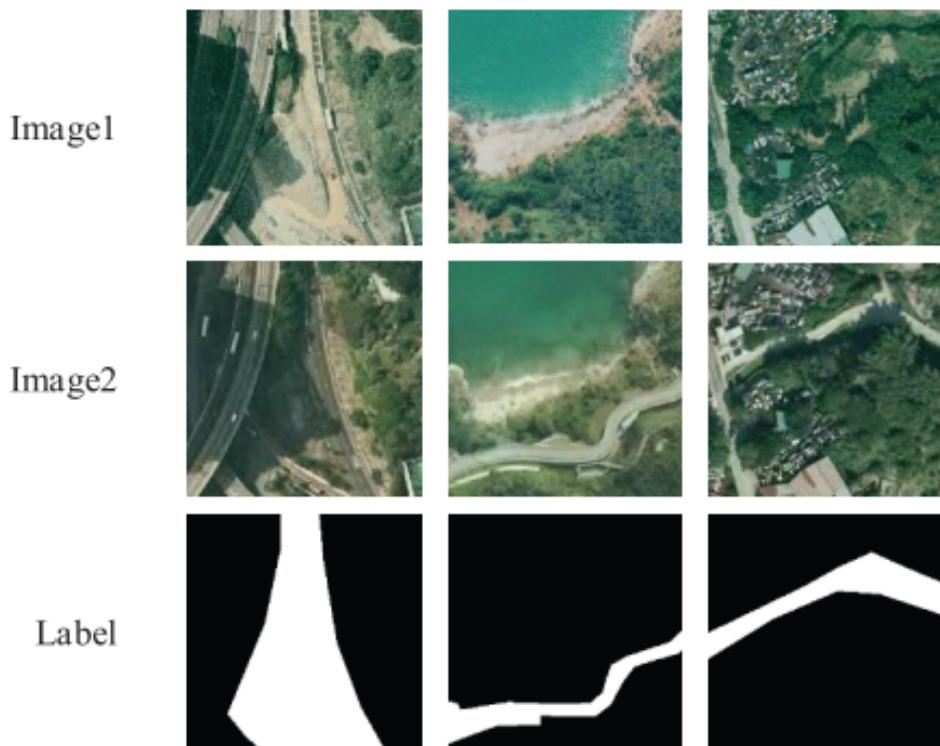


FIGURE 5.3 – exmeple d'échantillon du dataset SYSU-CD (Lien du dataset ici).

iv Etude comparative :

TABLE 5.1 – Etude comparative entre les datasets

Nom du dataset	Nombre de paires	Taille des images	Résolution des images
SZTAKI	13	952x640	1.5m/pixel
LEVIR-CD	673	1024x1024	0.5m/pixel
SYSU-CD	20 000	256*256	0.5m/pixel

Les jeux de données SZTAKI, LEVIR-CD et SYSU-CD partagent des caractéristiques communes pour la détection de changements, avec des images aériennes haute résolution capturant divers types de changements urbains et d'occupation des sols. Cependant, SYSU-CD se démarque par plusieurs aspects.

Tout d'abord, il contient beaucoup plus de paires d'images que les deux autres jeux de données, avec 20 000 paires contre seulement 13 pour SZTAKI et 637 pour LEVIR-CD. Ce plus grand volume de données le rend plus adapté aux méthodes d'apprentissage profond.

Deuxièmement, SYSU-CD couvre le plus de changements spécifiques aux zones urbaines denses, avec la construction de nouveaux gratte-ciel et des changements au niveau des ports, ce qui n'est pas le cas des deux autres jeux de données. Troisièmement, la résolution élevée de 0,5 mètre des images de SYSU-CD permet de détecter des changements fins dans les zones urbaines.

Pour finir, nous pouvons dire que le dataset SYSU-CD complète les ensembles de données existants en termes de résolution d'image, de types de changements et de volume de données. Il offre un nouvel ensemble de données de référence pour la détection de changements.

5.1.2 Préparation et division du dataset

Le dataset SYSU-CD a été divisé en ensembles d'entraînement, de validation et de test selon un ratio de [6 :2 :2]. Pour générer un ensemble de données adapté aux applications d'apprentissage profond, 25 paires d'échantillons de taille 256×256 ont été collectées aléatoirement à partir de chaque paire d'images, avec des opérations de retournement et de rotation aléatoires pour l'augmentation des données. Le prétraitement a abouti à un total de 20 000 paires de patchs d'images aériennes de taille 256×256 .

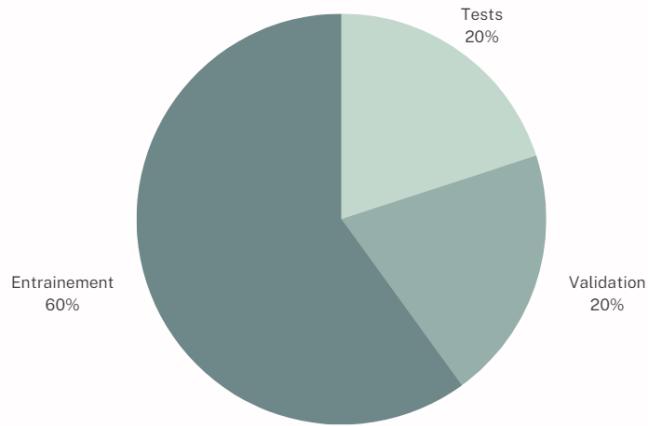


FIGURE 5.4 – Illustration de la séparation des données

5.1.3 Complémentarité du dataset SYSU-CD par rapport aux ensembles de données existants

D'après la comparaison qu'on a vu plus haut, le dataset SYSU-CD complète les ensembles de données existants en termes de résolution d'image, de types de changements et de volume de données. Il offre un nouvel ensemble de données de référence pour la détection de changements, en particulier pour les bâtiments de grande hauteur et les projets portuaires, qui sont difficiles à marquer dans les images haute résolution en raison de l'influence de la déviation et de l'ombre.

5.2 Outils utilisés :

5.2.1 Python

Python est un langage de programmation interprété multiplateforme et multi-paradigme. Python est flexible, indépendant de la plateforme et surtout donne accès à un grand nombre de bibliothèques et framework d'apprentissage automatique et profond. En plus de ça, il a une grande communauté de développeurs et chercheurs ce qui facilite son apprentissage, son utilisation et son évolution.



FIGURE 5.5 – Logo de Python.

5.2.2 Pytorch

PyTorch est une bibliothèque logicielle Python open source d'apprentissage machine qui s'appuie sur Torch développée par Facebook. PyTorch permet d'effectuer les calculs tensoriels nécessaires notamment pour l'apprentissage profond (DP). Ces calculs sont optimisés et effectués soit par le processeur (CPU) soit, lorsque c'est possible, par un processeur graphique (GPU) supportant CUDA. Il est issu des équipes de recherche de Facebook, et avant cela de Ronan Collobert dans l'équipe de Samy Bengio à l'IDIAP.



FIGURE 5.6 – Logo de Pytorch.

5.2.3 Tensorflow :

Tensorflow est une bibliothèque de logiciels open source conçue pour l'apprentissage automatique et les réseaux de neurones profonds. Elle est développée par Google Brain Team en 2011 et s'appelait à l'origine DistBelief. Google la modifie et la renomme Tensorflow en 2015, l'année où elle est rendue ouverte au public. Tensorflow regroupe une multitude d'algorithmes qui permettent de développer et d'exécuter des applications de ML et de DL. Ses outils offrent la possibilité de résoudre des problèmes mathématiques extrêmement complexes et permet donc au développeur de se focaliser sur la logique générale de l'application. cet outil a été utiliser pour lancer la premiere méthode.



FIGURE 5.7 – Logo de Tensorflow.

5.2.4 Google Colab

Colaboratory ,ou « Colab » en abrégé, est un produit de Google Research. Colab permet au développeur données d'écrire et d'exécuter du code Python arbitraire via le navigateur, et est particulièrement bien adapté à l'apprentissage automatique, à l'analyse de données et à l'éducation. Plus techniquement, Colab est un service de notebook Jupyter hébergé qui ne nécessite aucune configuration à utiliser, tout en offrant un accès gratuit aux ressources informatiques, y compris les GPU.

L'apprentissage des modèles profonds nécessite beaucoup de calculs complexes sur de grandes quantités de données donc un matériel puissant est nécessaire pour cette phase.

Pour l'apprentissage de nos modèles on a travaillé avec la plateforme de Google Colaboratory, sur laquelle nous avons exécuté.



FIGURE 5.8 – Logo de google colab.

5.3 Détails d'apprentissage :

5.3.1 Première approche :

Nous entraînons le modèle Xception sur l'ensemble d'entraînement créé dans notre base de données SYSU-CD et dont les images sont augmentées en variant l'éclairage dans un intervalle [0.6, 1.0]. Nous utilisons également la technique de dropout pour éviter le surapprentissage. Nous compilons le modèle à travers un optimiseur ADAM de *learning rate* = 0.001, de *batch size* = 250 et de loss function “*binary cross entropy*” et entamons l'entraînement pour 1000 *epochs* dont les graphes de 'loss' et de 'precision' sont illustrés dans les Figures suivantes.

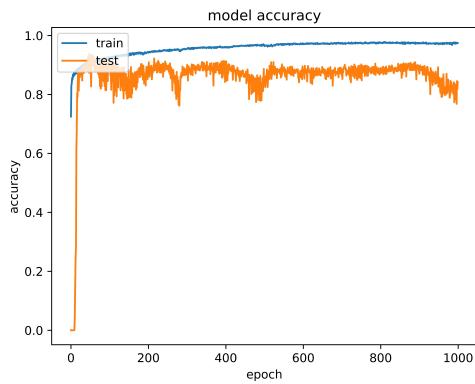


FIGURE 5.9 – Courbe de précision sur 1000 epoch

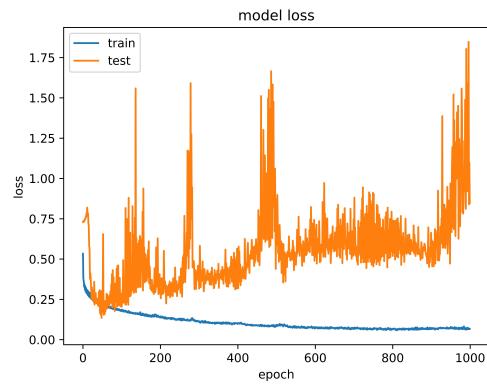


FIGURE 5.10 – Courbe de coût sur 1000 epoch

Au cours des 1000 epochs d’entraînement, le modèle a démontré une bonne performance , atteignant une précision de 97.41%. Cela indique une capacité robuste du modèle à apprendre et à généraliser à partir des données d’entraînement. De plus, la perte finale de 0.06 témoigne de l’efficacité du modèle à minimiser l’erreur pendant l’entraînement, ce qui est un indicateur positif de la qualité du modèle. En ce qui concerne les données de validation, le modèle a atteint une précision de 84.14% sur 1000 epochs. Cela démontre que le modèle a une bonne capacité à généraliser à partir de nouvelles données qui n’ont pas été vues pendant l’entraînement. Cette performance sur les données de validation est un indicateur important de la capacité du modèle à être appliqué à des situations réelles. En somme, ces résultats montrent que le modèle a une performance solide et est capable de faire des prédictions précises à la fois sur les données d’entraînement et de validation.

5.3.2 Deuxième approche :

Nous entraînons le modèle proposé sur l’ensemble d’entraînement de notre base de données SYSU-CD et dont les images sont augmentées en variant l’éclairage dans un intervalle [0.6, 1.0]. Nous utilisons également la technique de dropout pour éviter le surapprentissage. Nous compilons le modèle en utilisant les hyperparamètres affichés dans le tableau ci-dessous et dont les graphes de ‘loss’ et de ‘precision’ sont illustrés dans les Figures suivants.

Hyperparamètre	Valeur
Taux d'apprentissage	0.0001
Nombre d'éPOCHS	50
Taille du batch	16
Ratio CBAM	32
Taille du noyau	3x3

TABLE 5.2 – Hyperparamètres du deuxième modèle

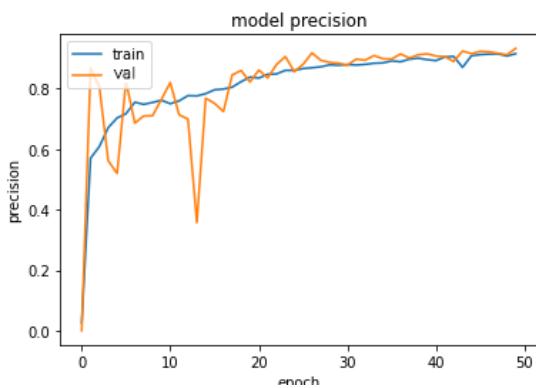


FIGURE 5.11 – Courbe de précision sur 50 epoch

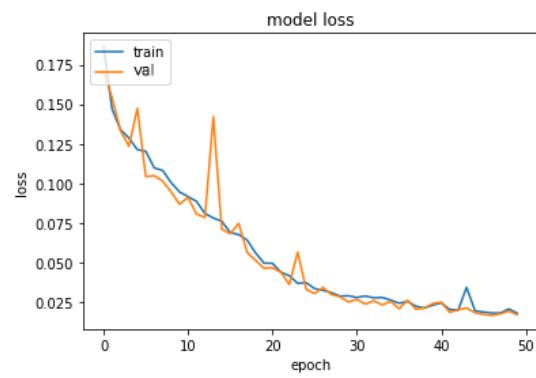


FIGURE 5.12 – Courbe de coût sur 50 epoch

Le modèle a montré une performance solide au cours des 50 epochs d'entraînement, avec une précision atteignant 84.02%. Cela démontre que le modèle a la capacité d'apprendre efficacement à partir des données d'entraînement et de généraliser à de nouvelles données. De plus, la perte finale de 0.025 indique que le modèle a réussi à minimiser l'erreur pendant l'entraînement, ce qui contribue à sa précision globale.

Les résultats sur les données de validation sont également prometteurs, avec une précision de 87.86% sur 50 epochs. Cela démontre que le modèle est capable de généraliser à de nouvelles données qui n'ont pas été utilisées lors de l'entraînement. Cette performance sur les données de validation est un indicateur important de l'aptitude du modèle à être appliquée dans des situations réelles.

En conclusion, ces résultats témoignent de la performance solide du modèle, avec une capacité robuste à faire des prédictions précises à la fois sur les données d'entraînement et de validation.

5.4 Critères d'évaluation

Afin d'évaluer les performances des deux méthodes proposées, nous faisons appel à certains critères standards adaptés aux algorithmes liées à la détection de changements et qui sont la CA, Recall (rappel) et le score F1.

5.4.1 Classification accuracy

Cette métrique nous permet d'avoir une idée générale de la performance d'un modèle de classification[54].

$$CA = \frac{VP + VN}{VP + VN + FP + FN} \quad (5.1)$$

- VP : Vrai positif, est une mesure représentant le nombre de pixels correctement classifiés comme “changés”.
- VN : vrai négatif, est une mesure représentant le nombre de pixels correctement classifiés comme “inchangés”.
- FP : Faux positif, est le nombre de pixels “inchangés” dans le GT mais qui sont classifiés comme “changés”.
- FN : Faux négatif, est le nombre de pixels “changés” dans le GT mais qui sont classifiés comme “inchangés”.

5.4.2 Recall (Sensibilité ou Taux de Vrais Positifs)

i Utilité :

Le recall est une métrique utilisée pour mesurer la proportion de positifs réels qui sont correctement identifiés. Il est également connu sous le nom de sensibilité ou taux de vrais positifs.

ii Formule :

Le recall est calculé à l'aide de la formule suivante :

$$Recall = \frac{VP}{VP + FN} \quad (5.2)$$

Dans cette formule, VP représente le nombre de vrais positifs et FN représente le nombre de faux négatifs. Le recall mesure la capacité du modèle à identifier correctement les positifs réels, en évitant les faux négatifs.

5.4.3 Score F1

i Utilité :

Le score F1 est une métrique qui combine à la fois la précision et le rappel en une seule valeur. Il est utilisé pour évaluer la performance globale d'un modèle de classification, en tenant compte à la fois des vrais positifs et des faux positifs.

ii Formule :

Le score F1 est calculé à l'aide de la formule suivante :

$$\text{ScoreF1} = \frac{2 \cdot (\text{Précision} \cdot \text{Rappel})}{\text{Précision} + \text{Rappel}} \quad (5.3)$$

Dans cette formule, la précision est définie comme suit :

$$\text{Précision} = \frac{VP}{VP + FP} \quad (5.4)$$

Dans ces formules, VP représente le nombre de vrais positifs, FN représente le nombre de faux négatifs et FP représente le nombre de faux positifs. Le score F1 atteint sa meilleure valeur à 1 (précision et rappel parfaits) et la pire valeur à 0.

5.5 Résultats et discussions :

5.5.1 Xception adapté à la détection de changements

i Analyse qualitative :

En premier lieu, nous allons vous présenter l'analyse qualitative des résultats que nous avons obtenu en affichant d'abord les images des résultats sur la partie test de la base de données afin de pouvoir faire une comparaison entre l'image T1, l'image T2, le résultats de la prédiction du modèle ainsi que le masque de changement qui représente la soustraction des deux images T1 et T2.

Image T1	Image T2	Prédiction	Masque de changements

TABLE 5.3 – Analyse qualitive Xception

Le tableau ci-dessus présente une sélection d'images qui jouent un rôle central dans notre étude. Dans la colonne "Image T1", nous avons inclus les images d'origine de notre échantillon. Ces images sont capturées à un instant initial et servent de point de référence pour notre analyse. Dans la colonne "Image T2", nous présentons les images correspondantes acquises à un instant ultérieur. Cela nous permet de comparer les différences et les évolutions au fil du temps. La colonne "Prédiction" représente les résultats de notre modèle d'analyse appliquée aux images T1 et T2.

Enfin, dans la colonne "Masque de changement", nous fournissons les annotations de référence. Ces annotations sont utilisées pour évaluer et valider nos prédictions. Elles servent de base de comparaison pour mesurer l'exactitude et la fiabilité de notre modèle.

En combinant ces différentes colonnes, nous avons l'opportunité de visualiser et d'analyser les résultats de notre étude de manière claire et concise.

ii Analyse quantitative :

Afin d'évaluer objectivement les performances de notre modèle, nous avons appliqué les métriques mentionnées précédemment. Celles-ci nous permettent d'obtenir des mesures

quantitatives et des indicateurs pertinents pour évaluer la précision de notre approche. Les résultats obtenus sont présentés dans le tableau suivant.

Image	Score F1 (%)	CA (%)	Recall (%)
03629	79,2	75,88	45,9
03632	61,4	79,59	95
03633	77,3	75,08	49.25

TABLE 5.4 – Analyse quantitative Xception

Les résultats obtenus pour cette méthode avec le fine tuning du réseau Xception pour la détection de changements indiquent que le modèle a réussi à apprendre des caractéristiques discriminantes, mais n'a peut-être pas atteint des performances exceptionnelles. Cela peut être dû à plusieurs facteurs. Tout d'abord, la complexité des scènes de changement peut rendre la tâche de détection plus difficile, en particulier si les modifications sont subtiles ou si les variations entre les images sont minimes. De plus, la disponibilité d'un ensemble de données d'entraînement adéquat et représentatif est essentielle pour l'apprentissage efficace des modèles de détection de changements. Si l'ensemble de données d'entraînement est limité en termes de quantité ou de diversité des changements, cela peut limiter les performances du modèle. En outre, la sélection des hyperparamètres peut également jouer un rôle crucial dans les performances du modèle. Une optimisation minutieuse des hyperparamètres, tels que le taux d'apprentissage, le *batch size* et le nombre d'époques d'entraînement, peut être nécessaire pour obtenir de meilleurs résultats. En résumé, bien que les résultats obtenus de Xception en détection de changements puissent indiquer une performance satisfaisante, une attention particulière doit être portée à l'amélioration des ensembles de données, à l'optimisation des hyperparamètres et à l'exploration de techniques complémentaires pour augmenter les performances du modèle."

5.5.2 VGG-16 amélioré adapté à la détection de changements

i Analyse qualitative :

Comme nous avons opéré pour le Xception plus haut, nous allons présenter dans cette partie l'analyse qualitative des résultats obtenus sur la partie test de la base de données en affichant les images T1, T2, la soustraction entre les deux, le résultats obtenu par le VGG-16 de base et enfin le résultat du modèle que nous proposons en tant que seconde méthode afin d'avoir une évaluation rigoureuse des performances de ce dernier.

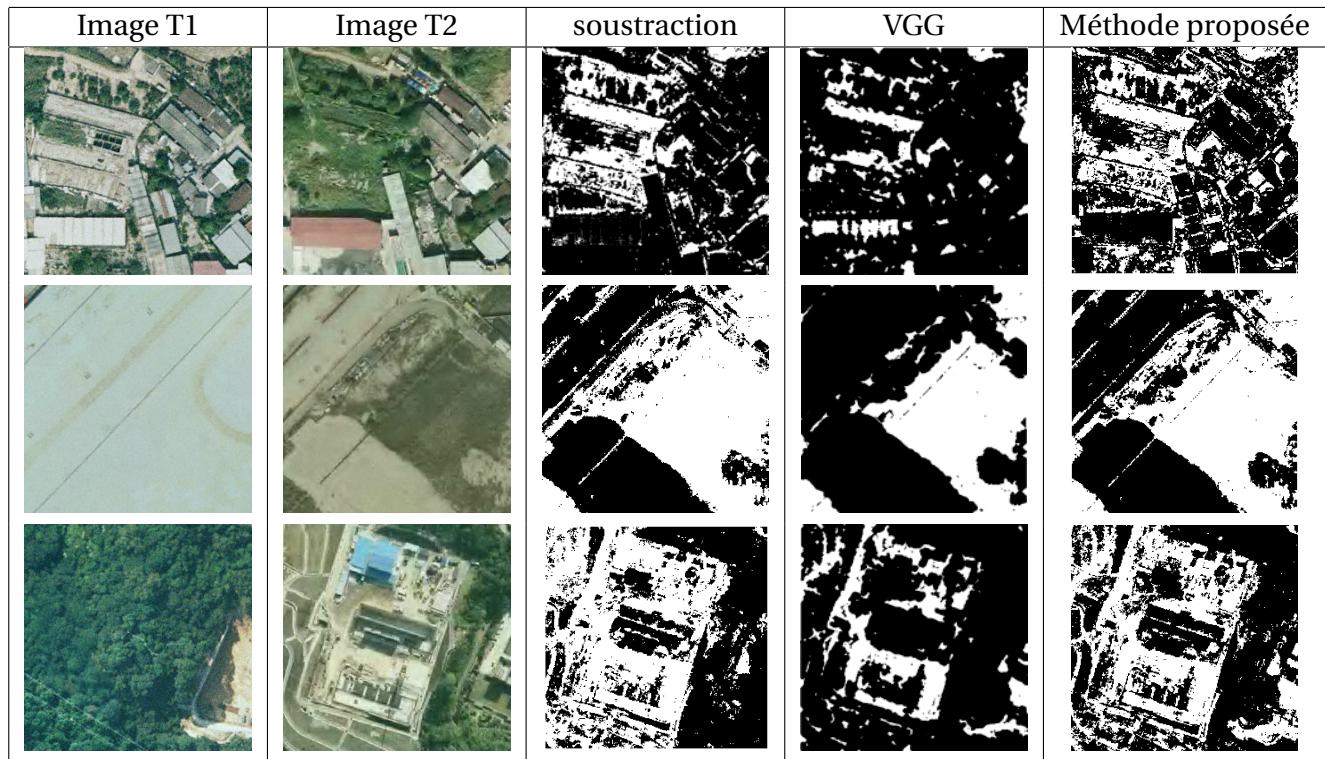


TABLE 5.5 – Analyse qualitative du VGG et de la méthode proposée

ii Analyse quantitative :

	VGG			Methode proposée		
	Precision %	Recall %	F1 %	Precision %	Recall%	F1 %
Test 1	74.216	62.037	70.299	92.003	81.101	82.158
Test 2	79.422	68.256	88.531	94.82	88.163	93.709
Test 3	89.822	56.824	69.611	90.708	75.974	82.69

TABLE 5.6 – Analyse quantitative du VGG simple et de la méthode proposée

Les résultats relatifs à cette deuxième méthode montrent que l'utilisation du module d'attention CBAM suivi de l'ajout du module DSlayer lors de l'entraînement du VGG-16 atteint des résultats satisfaisants, dépassant celle de la variante basée sur le réseau VGG-16 classique utilisé en tant que simple feature extractor et ceci, principalement en terme de la précision et du score F1 mais aussi pour le critère Recall.

L'intégration de CBAM avec VGG-16 peut améliorer la capacité du modèle à extraire des caractéristiques significatives et plus discriminantes et à attirer l'attention sur les parties importantes des images, ce qui peut conduire à de meilleures performances de détection de changements comme le montre les résultats obtenus. De même que l'ajout de DsLayer à VGG-

16 permet au modèle d'apprendre à la fois des caractéristiques denses et dispersées, ce qui peut améliorer sa capacité à détecter une variété de types de changements.

5.5.3 Discussion des résultats :

i Comparaison entre les deux méthodes :

Dans cette dernière partie, nous allons entreprendre une analyse comparative des deux méthodes que nous avons proposé. Notre objectif est de les évaluer tant sur le plan qualitatif que quantitatif, afin d'obtenir une critique objective de chacune d'elles.

Cette démarche vise à mieux cerner les contextes d'utilisation respectifs de ces méthodes et à identifier leurs forces et leurs faiblesses.

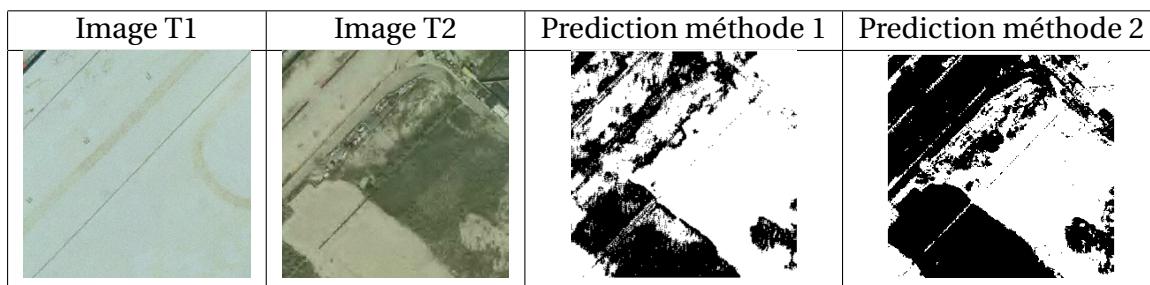


TABLE 5.7 – Analyse qualitative des deux méthodes sur la même paire d'images.

	modèle 1			Modèle 2		
	Precision	Recall	F1	Precision	Recall	F1
Test 2	61,4	79,59	95	94.82	88.163	93.709

TABLE 5.8 – Analyse quantitative de la méthode 1 et de la méthode 2 sur la même paire d'images.

Lorsqu'il s'agit de la détection de changements, deux approches couramment utilisées sont l'utilisation de VGG-16 en tant qu'extracteur de caractéristiques et le fine-tuning de Xception. Examinons les avantages et les inconvénients de chaque méthode.

Utilisation du VGG-16 en tant qu'extracteur de caractéristiques en lui ajoutant les modules CBAM et DSLayer : VGG-16 est un réseau de neurones convolutifs (CNN) pré-entraîné, largement utilisé et bien établi dans la communauté de la vision par ordinateur. Il a été formé sur de grandes bases de données d'images, ce qui lui confère une capacité à extraire des caractéristiques visuelles significatives. Cependant, les couches convolutives de VGG-16 ont été pré-entraînées sur une tâche de classification d'images, ce qui signifie qu'elles sont plus adaptées à extraire des caractéristiques de bas niveau telles que les bords, les textures, etc.

Pour la détection de changements, où l'on souhaite identifier des modifications dans une scène, des caractéristiques de plus haut niveau, telles que la forme des objets, la composition spatiale, etc., peuvent être nécessaires. VGG-16 peut ne pas être aussi performant dans cette tâche, C'est pour ça qu'on a rajouter le module CBAM qui se concentre sur les channel attention et spatial attention, nos caractéristiques deviennent plus discriminantes et plus précises, et aussi le module DSLayer qui aide dans l'apprentissage et qui rend le modèle encore plus précis.

Fine-tuning de Xception : Xception est un réseau CNN plus avancé et complexe que VGG-16, qui a été spécifiquement conçu pour la tâche de classification d'images. En effectuant un fine-tuning de Xception, on peut ajuster les poids du réseau à un nouveau jeu de données spécifique à la détection de changements. Cela permet d'adapter les caractéristiques extraites par Xception aux besoins spécifiques de la tâche de détection de changements, ce qui peut améliorer les performances. Mais le fine-tuning nécessite généralement un ensemble de données d'entraînement important et annoté pour obtenir de bons résultats. Il peut être plus coûteux en termes de temps et de ressources par rapport à l'utilisation d'un extracteur de caractéristiques pré-entraîné tel que VGG-16. Si l'ensemble de données d'entraînement est limité ou non représentatif des variations présentes dans les images de changement, le fine-tuning peut entraîner un surapprentissage.

En conclusion, l'utilisation de VGG-16 en tant qu'extracteur de caractéristiques est plus rapide et nécessite moins de ressources, mais peut être moins performante dans la détection de changements nécessitant des caractéristiques de plus haut niveau, c'est pour ça qu'on a introduit les modules CBAM et DSLayer qui augmenter la précision du VGG-16. Le fine-tuning de Xception permet d'obtenir de meilleures performances en adaptant le réseau à la tâche spécifique de détection de changements, mais il nécessite un ensemble de données d'entraînement suffisamment large et peut être plus coûteux en termes de temps et de ressources. Le choix entre ces deux approches dépendra des contraintes spécifiques du projet et des performances souhaitées.

Conclusion :

Après l'évaluation des deux modèles de détection de changements sur le dataset SYSU-CD. Le premier modèle était basé sur l'architecture Xception, utilisée pour la première fois sur ce dataset. Bien que les résultats obtenus aient démontré une performance satisfaisante, les changements inhérents au dataset ont pu limiter les capacités du modèle. De plus, une optimisation minutieuse des hyperparamètres, tels que le taux d'apprentissage, le batch size et le nombre d'epoch d'entraînement, pourrait s'avérer nécessaire pour améliorer les perfor-

mances.

Le second modèle était basé sur l'architecture VGG16, améliorée par l'ajout de deux modules complémentaires : le CBAM et le DS Layer. Comparativement au VGG16 standard et au modèle Xception, ce VGG16 amélioré a démontré une performance nettement supérieure, tant sur le plan quantitatif que qualitatif. Cette amélioration significative met en évidence l'efficacité des modules ajoutés pour renforcer les capacités d'extraction de caractéristiques du VGG16.

CONCLUSION

La détection de changement dans le milieu urbain est un domaine très important et vaste où plusieurs méthodes sont utilisées dont chacune donne des résultats différents que ce soit pour les méthodes classiques ou les méthodes d'apprentissage automatique qui ont prouvé une performance remarquable comparée aux méthodes classiques. Ce projet de PFE s'inscrit dans le cadre d'un projet de détection de changement dans le milieu urbain par images d'observation de la terre et technologie avancées d'apprentissage profond.

Ensuite, nous avons abordé l'apprentissage automatique et profond dont on met l'accent sur les réseaux de neurones convolutifs.

Dans la première partie de ce rapport, nous nous sommes intéressés à l'état de l'art du domaine de la télédétection ainsi que la détection de changement dans le milieu urbain.

Dans la deuxième partie du projet, nous avons commencé par définir l'apprentissage profond et on a parlé des approches du deep learning utilisées dans la détection de changements, et de manière plus précise aux réseaux de neurones convolutifs.

Enfin la dernière partie du mémoire était consacrée à la conception et l'implémentation des méthodes proposées qui consistaient à implémenter d'un côté une architecture VGG-16, un réseau de neurones peu profond, en introduisant deux modules CBAM et DSlayer qui rendent les caractéristiques plus discriminantes et proposent un meilleur apprentissage, et de l'autre l'implémentation d'un réseau de neurones plus profond à savoir le Xception, tout en l'adaptant à notre domaine d'étude à savoir la détection de changements.

Dans le cadre de cette étude, nous avons entrepris une exploration approfondie de ces deux méthodes pour la détection de changements, en utilisant deux architectures de réseaux de neurones qui se distinguent par leurs profondeurs. Ces méthodes ont été appliquées à un ensemble de données complet, fournissant ainsi une évaluation robuste de leur performance.

La première méthode a impliqué l'adaptation du modèle Xception, un réseau de neurones profond, pour détecter les changements. Le modèle Xception, qui est connu pour sa profondeur et sa complexité, a été choisi dans le but d'obtenir une précision maximale dans la détection de changements.

La seconde méthode a utilisé un réseau moins profond, le VGG16, qui a été amélioré avec l'ajout des modules CBAM et DS Layer. Ces modules ont été intégrés pour augmenter la capacité du modèle à se concentrer sur les caractéristiques pertinentes pour la détection de changements, tout en conservant une architecture moins complexe et plus efficace en termes de temps et de ressources.

Les résultats obtenus ont montré que les deux méthodes offrent de bonnes performances et une bonne généralisation sur l'ensemble de données. Cependant, il a été observé que le modèle Xception, malgré ses performances élevées, a nécessité un temps d'entraînement considérablement plus long en raison de sa profondeur. Cela a mis en évidence l'importance de prendre en compte non seulement la précision du modèle, mais aussi l'efficacité en termes de temps et de ressources lors du choix d'une architecture de réseau de neurones.

En conclusion, le choix de la méthode pour la détection de changements dépend fortement des attentes et des contraintes de l'utilisateur. Si l'objectif principal est d'obtenir la meilleure précision possible, l'utilisation du modèle Xception pourrait être justifiée, malgré son temps d'entraînement plus long. Cependant, si l'efficacité en termes de temps et de ressources est une préoccupation majeure, le modèle VGG16 avec les modules CBAM et DS Layer pourrait être une alternative préférable.

En perspective, il serait intéressant d'explorer les points suivants :

- Introduire des algorithmes de mét-heuristiques qui proposent une optimisation des hyper-paramètres choisis
 - Une autre perspective consiste à appliquer toutes ces méthodes sur autre jeux de données et plus précisément sur des villes algériennes pour enrichir les états de l'art de la détection de changement dans le milieu urbain.
 - On peut aussi utiliser aussi d'autres modules d'optimisation comme les deux utilisés (CBAM et DSlayer) afin d'avoir une meilleure précision.
 - Enfin et pour que notre modèle soit plus robuste et performant il doit être entraîné de zéro et ne pas utiliser le transfert learning cependant cela demande énormément de temps et très coûteux en terme de ressources.

BIBLIOGRAPHIE

- [1] Developpez.net. Forum : Réseaux de convolution et neurones. <https://www.developpez.net/forums/d1834627/general-developpement/algorithme-mathematiques/intelligence-artificielle/reseaux-convolution-neurones/>, YYYY. Accessed : DD Month YYYY.
- [2] Papers with code. <https://paperswithcode.com/method/max-pooling>. Accessed : [Insert Date].
- [3] James B. Campbell. *Introduction to Remote Sensing*. Guilford Press, 2007.
- [4] E. W. Russell. Aerial photography and remote sensing for soil survey. by l. p. white. oxford university press (1977).. *Experimental Agriculture*, 14(4) :400–400, 1978.
- [5] Yuting Zhou, K. Colton Flynn, Prasanna H. Gowda, Pradeep Wagle, Shengfang Ma, Vijaya G. Kakani, and Jean L. Steiner. The potential of active and passive remote sensing to detect frequent harvesting of alfalfa. *International Journal of Applied Earth Observation and Geoinformation*, 104 :102539, 2021.
- [6] Emil TUDOR, Ionuț VASILE, Gabriel POPA, and Marius GHETI. Lidar sensors used for improving safety of electronic-controlled vehicles. In *2021 12th International Symposium on Advanced Topics in Electrical Engineering (ATEE)*, pages 1–5, 2021.
- [7] Youcef Chibani. Selective synthetic aperture radar and panchromatic image fusion by using the $\tilde{A}f$ trous wavelet decomposition. *EURASIP Journal on Advances in Signal Processing*, 2005, 08 2005.
- [8] WeiFa Zheng and ZiXin Xie. Spatial-spectral deep residual network for hyperspectral image super-resolution. *SN Computer Science*, 4, 06 2023.
- [9] Lixing Zhao, Jingjie Jiao, Lan Yang, Wenhao Pan, Fanjun Zeng, Xiaoyan Li, and Fansheng Chen. A cnn-based layer-adaptive gcps extraction method for tir remote sensing images. *Remote Sensing*, 15 :2628, 05 2023.

- [10] Robin Faivre. Introduction à la réalisation de spatio-cartes. ICube-SERTIT Formation Télédétection, UEH, mai 2018. Université de Strasbourg.
- [11] Soufiane Idbraim, Zakaria Mimouni, Mohamed Salah, and Mohamed Dahbi. *CNN Model for Change Detection of Argania Deforestation from Sentinel-2 Remote Sensing Imagery*, pages 716–725. 03 2023.
- [12] Hua Su, Yanan Wei, Wenfang Lu, Xiao-Hai Yan, and Hongsheng Zhang. Unabated global ocean warming revealed by ocean heat content from remote sensing reconstruction. *Remote Sensing*, 15 :566, 01 2023.
- [13] Clement Atzberger. Advances in remote sensing of agriculture : Context description, existing operational monitoring systems and major information needs. *Remote Sensing*, 5(2) :949–981, 2013.
- [14] Mohd Nazip Suratman, Zulkiflee Abd Latif, Tengku Tengku Hashim, Ahmad Mohsin, Nazlin Asari, and Nurul Mohd Zaki. *Remote Sensing for Forest Inventory and Resource Assessment*, pages 3–23. 01 2023.
- [15] Andromachi Chatziantoniou, Nikos Papandroulakis, Orestis Stavrakidis-Zachou, Spyros Spondylidis, Simeon Taskaris, and Konstantinos Topouzelis. Aquasafe : A remote sensing, web-based platform for the support of precision fish farming. *Applied Sciences*, 13 :6122, 05 2023.
- [16] David M. Tralli, Ronald G. Blom, Victor Zlotnicki, Andrea Donnellan, and Diane L. Evans. Satellite remote sensing of earthquake, volcano, flood, landslide and coastal inundation hazards. *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(4) :185–198, January 2005.
- [17] Stefan Voigt, Thomas Kemper, Torsten Riedlinger, Ralph Kiefl, Klaas Scholte, and Harald Mehl. Satellite image analysis for disaster and crisis-management support. *IEEE T. Geoscience and Remote Sensing*, 45 :1520–1528, 06 2007.
- [18] Christopher Small and Robert J. Nicholls. A global analysis of human settlement in coastal zones. *Journal of Coastal Research*, 19(3) :584–599, 2003.
- [19] Shivangi Mishra, Priyanka Shrivastava, and Priyanka Dhurvey. Change detection techniques in remote sensing : A review. *International Journal of Wireless and Mobile Communication for Industrial Systems*, 4 :1–8, 04 2017.
- [20] D. Lu, P. Mausel, E. Brondízio, and E. Moran. Change detection techniques. *International Journal of Remote Sensing*, 25(12) :2365–2401, 2004.
- [21] Jwan Al-doski, Shattri Bin Mansor, and Helmi Zulhaidi Mohd Shafri. Change detection process and techniques. *Civil and environmental research*, 3 :37–45, 2013.

- [22] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [23] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.
<http://www.deeplearningbook.org>.
- [24] F. Rosenblatt. The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6) :386–408, 1958.
- [25] Paul Werbos and Paul John. Beyond regression : new tools for prediction and analysis in the behavioral sciences /. 01 1974.
- [26] Jeffrey L. Elman. Finding structure in time. *Cognitive Science*, 14(2) :179–211, 1990.
- [27] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.
- [28] Su Wang. Generative adversarial networks (gan) : A gentle introduction [updated], 04 2017.
- [29] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4) :541–551, 1989.
- [30] Jiajun Zhang and Chengqing Zong. Deep neural networks in machine translation : An overview. *IEEE Intelligent Systems*, 30 :16–25, 09 2015.
- [31] Shiv Ram Dubey, Satish Kumar Singh, and Bidyut Baran Chaudhuri. Activation functions in deep learning : A comprehensive survey and benchmark, 2021.
- [32] Yann Lecun, Leon Bottou, Y. Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86 :2278 – 2324, 12 1998.
- [33] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.
- [34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.
- [35] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.
- [36] François Chollet. Xception : Deep learning with depthwise separable convolutions, 2017.
- [37] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10) :1345–1359, 2010.

- [38] Dipanjan (DJ) Sarkar. *A Comprehensive Hands-on Guide to Transfer Learning with Real-World Applications in Deep Learning*. 2018.
- [39] Transfer learning : Qu'est-ce que c'est? (datascientest web article).
- [40] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks?, 2014.
- [41] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf : an astounding baseline for recognition, 2014.
- [42] Chris Kawatsu, Frank Koss, Andy Gillies, Aaron Zhao, Jacob Crossman, Benjamin Purman, David Stone, and Dawn Dahn. Gesture recognition for robotic control using deep learning. 08 2017.
- [43] Qian Shi, Mengxi Liu, Shengchen Li, Xiaoping Liu, Fei Wang, and Liangpei Zhang. A deeply supervised attention metric-based network and an open aerial image dataset for remote sensing change detection. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–16, 2021.
- [44] Chenxiao Zhang, Peng Yue, Deodato Tapete, Liangcun Jiang, Boyi Shangguan, Li Huang, and Guangchao Liu. A deeply supervised image fusion network for change detection in high resolution bi-temporal remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 166 :183–200, 2020.
- [45] Jie Chen, Ziyang Yuan, Jian Peng, Li Chen, Haozhe Huang, Jiawei Zhu, Yu Liu, and Haifeng Li. DASNet : Dual attentive fully convolutional siamese networks for change detection in high-resolution satellite images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14 :1194–1206, 2021.
- [46] Meziane Iftene, Mohammed El Amin Larabi, and Moussa Sofiane Karoui. End-to-end change detection in satellite remote sensing imagery. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pages 4356–4359, 2021.
- [47] Yi Long, Heng-Chao Li, Turgay Celik, Nathan Longbotham, and William J. Emery. Pairwise-distance-analysis-driven dimensionality reduction model with double mappings for hyperspectral image visualization. *Remote Sensing*, 7(6) :7785–7808, 2015.
- [48] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In-So Kweon. Cbam : Convolutional block attention module. In *European Conference on Computer Vision*, 2018.
- [49] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742, 2006.

- [50] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net : Fully convolutional neural networks for volumetric medical image segmentation. 2016.
- [51] Csaba Benedek and Tamás Szirányi. Change detection in optical aerial images by a multilayer conditional mixed markov model. *IEEE Transactions on Geoscience and Remote Sensing*, 47(10) :3416–3430, 2009.
- [52] Csaba Benedek and Tamas Sziranyi. A mixed markov model for change detection in aerial photos with large time differences. In *2008 19th International Conference on Pattern Recognition*, pages 1–4, 2008.
- [53] Hao Chen and Zhenwei Shi. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection, remote sensing journal. volume 12, 2020.
- [54] Ouadah Cylia and Hamadache Rachika ElHassna. Détection de changement par images d’observation de la terre et techniques avancées d’apprentissage profond, projet de fin d’étude master ecole nationale polytechnique. September 2020.