

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Master

Mention Électronique
Spécialité Traitement de l'Information et Systèmes Électroniques

présenté par

HADJI AHMED

&

CHABANE SAFIA

EXTRACTION DES RESEAUX ROUTIERS PAR DEEP LEARNING

Proposée Par : Mme N.Bougherira

Année Universitaire 2020-2021

Remerciements

Au terme de cette étude, qu'il nous soit permis de témoigner toute notre reconnaissance et nos sincères remerciements à tous ceux qui ont participé d'une manière ou une autre à l'élaboration de ce travail.

*Nous tenons tout d'abord adresser toute notre reconnaissance et notre profond respect à **Mme BOUGHERIRA KHETIB NADIA**, pour avoir dirigé ce travail, pour la confiance et l'intérêt qu'elle nous a donné, et pour l'ensemble des informations qu'elle nous a offert. Nous tenons à lui exprimer notre gratitude la plus sincère, et notre plus grand respect à son égard, en espérant qu'aujourd'hui, elle est fière de nous autant que nous sommes fiers d'avoir été ses étudiants.*

*Notre gratitude va aussi vers notre chef de spécialité **Mme NACEUR DJAMILA** pour son encouragement.*

Un grand merci également à tous les enseignants du département d'électronique de l'université Saad Dahleb Blida pour tout le savoir qu'ils ont su nous transmettre durant notre cursus universitaire.

Nous tenons aussi à remercier les membres du jury pour l'intérêt qu'ils ont porté à notre projet en acceptant d'examiner notre travail et de l'enrichir par leurs propositions.

Nous tenons à exprimer nos plus profondes reconnaissances et respect a tous nos enseignants pour leurs encouragements et leurs aides tout au long du chemin.

Dédicaces

Je dédie ce travail

A MON TRES CHER PERE Omar

Je tiens à honorer l'homme que tu es. Grace à toi papa j'ai appris le sens du travail et responsabilité.

Je voudrais te remercier pour ton amour, ta générosité, ta compréhension...ton soutien fut une lumière dans tout mon parcours. Ce modeste travail est le fruit de tous les sacrifices que tu as déployés pour mon éducation et ma formation. Que dieu te préserve et te procure longue vie.

À MA CHERE MAMAN Houria

Autant de phrases aussi expressives soient-elles ne sauraient montrer le degré d'amour et d'affection que j'éprouve pour toi. Tu m'as comblé avec ta tendresse et affection tout au long de mon parcours. Tu n'as cessé de me soutenir et de m'encourager, tu as été toujours présente à mes côtés pour me consoler quand il fallait. Que dieu, le tout puissant vous préserve, vous accorde santé, bonheur, quiétude de l'esprit et vous protège de tout mal.

À MA SŒUR Khadidja ET MON FRERE Walid

Merci d'être toujours à mes côtés, par votre amour dévoué, pour donner de gout et du sens à ma vie. Je vous prie de trouver dans ce travail l'expression de mon estime et mon sincère attachement.

À ma nièce et mon neveu Ryme, Badis que Dieu les protège.

A MES MEILLEURS AMIS Johnny, Ihcene, Zoulikha

Je ne peux trouver les mots justes et sincères pour vous exprimer mon affection, vous êtes pour moi des amis sur qui je peux compter. Je vous dédie ce travail je vous souhaite une vie pleine de santé et de bonheur.

À MES CHERES AMIES

Fatma-Zohra, Sabrina, Aya, Zaara ma vie ne serait pas aussi magique sans votre présence.

A TOUTE MA FAMILLE

Aucun langage ne saurait exprimer mon respect et ma considération pour votre soutien et encouragements.

Safia

Dédicaces

Je dédie ce travail avec beaucoup d'amour à toute ma famille pour leur soutien inconditionnel, pour me pousser toujours à être meilleur et réussir ma carrière a conduit à ce jour prévu.

À MON CHER PERE

pour m'avoir apporté son soutien dès le premier jour, tant monétaire que moral, ses conseils et son éducation ont été parmi les meilleurs

À MA CHERE MAMAN

aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être. Je vous remercie pour tous le soutien et l'amour que vous me portez depuis mon enfance. C'est pourquoi je vous donne ce travail en offrande pour votre patience.

À MA FAMILLE

d'avoir toujours été là pour moi, vous savez que cette réalisation est aussi pour vous.

À MES CHERES AMIES

merci énormément pour vos soutiens plus que précieux.

Ahmed

ملخص:

الهدف من هذا المشروع في خطوته الاولى حول مفهوم الذكاء الاصطناعي (intelligence artificielle) و التعليم الآلي (machine learning) وكذلك التعليم العميق (deep learning) ثم التعريف على برنامج U.Net و في الخطوة الثانية، انشاء و تدريب هذا الأخير (U.Net) باستخدام الأداة (Google colab) الموجهة المجالات التعرف الخاضعة تحت اشراف شبكة CNN. النموذج الذي تم انشاءه سيكون قادرا على استخراج شبكة طرق في خرائط للأقمار الصناعية. الكلمات المفتاحية: شبكة طرق، خرائط للأقمار الصناعية، شبكات CNN ، التعليم الآلي، التعلم العميق، الذكاء الاصطناعي.

Résumé :

L'objectif de ce projet est consiste à faire dans une première étape une présentation de l'Intelligence Artificiel et le Machine Learning, ainsi que le Deep Learning, puis définir le modèle U.NET, et, dans une seconde étape, à créer et entrainer ce dernier (modèle U.net), en utilisant des outils (Google Colab....) destinés aux domaines de la reconnaissance supervisée a base des réseaux CNN. Le modèle créé va être capable de faire une extraction d'un réseau voirie dans des cartes satellitaires.

Mots clés : réseau voirie , cartes satellitaires, réseaux CNN, Google Colab, U.net, Machine Learning, Deep Learning Intelligence Artificiel.

Abstract:

The present project consists of a presentation of the Artificial Intelligence and the Machine Learning and the Deep Learning , then define the U.net model, and, in a second step, to create and train the (U.net model), using tools like (Google Colab, ...),designed for the fields of supervised recognition based on CNN networks. The model created will be able to extract road network in Satellite maps.

Keywords: road network, urban, satellites maps, networks CNN, Google Colab,U.net, Machine Learning, Deep Learning, Artificial Intelligence.

Liste des acronymes et abréviations

BP : Back propagation.

CNN : Convolutional Neural Network (Réseau neuronal convolutif).

CPU : Central Processing Unit (unités centrales de traitement).

CUDA: Compute Unified Device Architecture

DL : Deep Learning.

TPU : Tensor Processing Unit.

IA: Intelligence Artificielle.

IH : Intelligence Humaine.

ML : Machine learning.

MLP : Multilayer perceptron (Réseaux à plusieurs neurones).

NN: Region with Convolutional Neural Network (Réseau neuronal convolutif régionale).

RNN : Recurrent Neural Network (Réseau à connexions récurrentes).

ReLU : Rectified Linear Unit.

U.net : Réseau de neurone de computer vision.

Table des matières

Introduction générale	01
Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain.	
I.1 Introduction	04
I.2 La méthode de Muratori	04
I.3 Le modèle interprétatif de Gianfranco Caniggia	06
I.3.1 Notion de modularité	06
I.3.2 Notion de nodalité	07
I.3.3 Notion de hiérarchie	07
I.4 Définition de paramètres issus du modèle interprétatif	08
I.4.1 Définition d'un Réseau de voirie	08
I.4.2 Définition d'une Parcelle	09
I.4.3 Définition d'un Ilot	09
I.5 Cas d'étude centre ancien de blida	10
I.5.1 Situation géographique	10
I.5.2 Croissance urbaine du centre ancien de Blida	11
a) Période pré turque	11
b) Période médiévale	11
c) Période coloniale	12
I.6 Présentation de la carte urbaine	12
a) Les cartes topographiques	12
b) Les cartes spéciales	13
c) Les cartes thématiques	14
d) Les cartes météorologiques	14
e) Les cartes géologiques	15
I.6.1 Cartes urbaines cas d'étude	16
I.7 Conclusion	17
Chapitre II : Généralité sur le Deep Learning : les outils et le modèle.	
II.1 Introduction	19
II.2 L'intelligence artificielle (IA)	19
II.3 Machine learning (ML)	19
II.4 Deep Learning (DL)	20
II.4.1 Neurone biologique	21
II.4.2 Neurone artificiel	22
II.4.3 Le perceptron	23
II.4.4 Perceptron multicouches	24
II.5 L'Apprentissage	25
II.5.1 L'apprentissage supervisé	25
II.5.2 L'apprentissage non- supervisé	26
II.5.3 L'apprentissage semi supervisé	27
II.5.4 L'apprentissage renforcé	27
II.6 Les fonctions d'activation	27
II.6.1 La fonction ReLu (Rectified Linear Unit)	27
II.6.2 la fonction sigmoïde	27

II.7	Les modèles d'interconnexion des nerones	28
	II.7.1 Réseau à connexion complète	28
	II.7.2 Réseau a connexion locales	29
	II.7.3 Réseau à connexion récurrentes (RNN)	29
II.8	Les réseaux de neurones convolutifs (CNN)	29
	II.8.1 Définitions CNN	30
	II.8.2 Les couches utilisées dans un CNN	30
	a) La couche de convolution (Convolution Layer)	30
	b) La couche de mise en commun (pooling layer)	31
	c) La couche entièrement connectée (Fully-connected)	32
	d) la couche de perte (LOSS)	32
II.9	Réseaux de Neurones Convolutifs Régionaux (R-CNN)	32
II.10	Fast R-CNN	33
II.11	Faster R-CNN	33
II.12	Mask R-CNN	34
II.13	Réseau de neurones de computer vision U-NET	34
	II.13.1 la segmentation sémantique	35
	II.13.2 Architecture de U-NET	35
	a) Encodeur (côté gauche)	35
	b) Décodeur (côté droite)	36
	II.13.3 Les avantages de U-NET	36
II.14	Conclusion	37
Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.		
III.1	Introduction	39
III.2	Les outils utilises	39
	III.2.1 Langage de programmation python	39
	III.2.2 Jupiter Notebook	39
	III.2.3 Les bibliothèques Python	40
	a)Numpy	40
	b)Matplotip	40
	c)OpenCv	40
	d)Tensorflow	40
	e)Keras	40
	III.2.4 Google Colab	41
	III.2.5 Google Drive	42
III.3	Schématisation des étapes du DL	42
	III.3.1 Chargement des données	43
	III.3.2 Importation des bibliothèques et modules	44
	III.3.3 Préparation des donnés	44
	III.3.4 Prétraitement des donnés	45
	III.3.5 Configuration de modèle	45
	a)Chemin de de contrat	45
	b)Chemin de d'extension	46
	III.3.6 Entraînement	47

III.3.7 Test	50
III.3.8 Les résultats	51
III.4 Interface graphique	52
III.5 Application sur une carte satellitaire de centre de blida	53
III.6 Conclusion	56
Conclusion générale	57
Bibliographie	58
Annexe	60

Liste des figures

Chapitre I : L’aspect urbain de la ville et extraction d’objets urbain.	
Figure I.1 organigramme du système d’aide à la reconnaissance des tissus urbains	08
Figure I.2 <i>réseau routiers</i>	09
Figure I.3 <i>Exemple de placement des parcelles dans un ilot</i>	09
Figure I.4 <i>Représentation d’un ilot dans un bloc urbanisme</i>	10
Figure I.5 Carte topographies	13
Figure I.6 Carte maritime	13
Figure I.7 Carte thématique de sol	14
Figure I.8 Carte de météo	15
Figure I.9 Carte géologique de Algérie	16
Figure I.10 Image satellitaire d’une zone urbaine	16
Chapitre II : Généralité sur le Deep Learning : les outils et le modèle.	
Figure II.1 La relation entre l’intelligence artificielle, ML et le Deep Learning	19
Figure II.2 Neurone biologique	22
Figure II.3 les elements d’un neurone formel	22
Figure II.4 Représentation du perceptron	23
Figure II.5 Réseau neurone (MLP)	24
Figure II.6 schéma synoptique des types d’apprentissage	25
Figure II.7 Apprentissage supervise	26
Figure II.8 Apprentissage non supervise	26
Figure II.9 la fonction Relu et sa dérivée	27
Figure II.10 Fonction d’activation Sigmoidé.	28
Figure II.11 Réseau à connexion complète	28
Figure II.12 <i>Réseau a connexion locales</i>	29
Figure II.13 <i>Réseaux à connexion récurrente</i>	29
Figure II.14 <i>Différentes couches CNN</i>	30
Figure II.15 Exemple de produit de convolution	31
Figure II.16 Les types de pooling	32
Figure II.17 L'architecture R-CNN d'origine	33
Figure II.18 L'architecture Fast R-CNN.	33
Figure II.19 L'architecture plus rapide R-CNN « Faster R-CNN ».	34
Figure II.20 Mask R-CNN. La sortie des couches CONV est le masque lui- même.	34
Figure II.21 image et étiquette de pixel.	35
Figure II.22 architecture de U-NET	36
Chapitre III : Extraction des Réseaux routiers à partir des cartes satellitaires.	
Figure III.1 Interface de Google Colab	41
Figure III.2 <i>Les étapes de détection par DL</i>	42
Figure III.3 Relier kaggle avec google drive	43
Figure III.4 téléchargement dataset vers google drive	43
Figure III.5 décompression du fichier	43
Figure III.6 Importation des bibliothèques	44
Figure III.7 tri des paramètres des images.	44
Figure III.8 Redimensionnement des images	45

Figure III.9 <i>chemin de contrat</i>	46
Figure III.10 <i>chemin d'extension</i>	46
Figure III.11 <i>les couches de notre modèle</i>	46
Figure III.12 <i>Phase d'entraînement.</i>	47
Figure III.13 <i>Les étapes d'activation TPU</i>	48
Figure III.14 <i>Début de l'entraînement</i>	48
Figure III.15 <i>la fin de l'entraînement</i>	49
Figure III.16 <i>Graphique de la perte/précision en fonction du nombre d'itération</i>	49
Figure III.17 <i>le nombre de paramètre</i>	50
Figure III.18 <i>Les images testées (à gauche :images originales, à droite : masque originaux, au milieu : images générées)</i>	50
Figure III.19 <i>Le schéma synoptique de la prédiction</i>	51
Figure III.20 <i>Les résultats obtenus</i>	52
Figure III.21a <i>lancement de l'interface.</i>	53
Figure III.2b <i>exécution du modèle</i>	53
Figure III.21c <i>prédiction.</i>	54
Figure III.22 <i>les cartes de Blida après la segmentation</i>	55

Introduction générale

Introduction générale

Le domaine de l'intelligence artificielle (le deep learning) est utilisé pour extraire par segmentation a partir des images satellitaire des réseaux routiers.

Notre objectif est d'entraîner un modèle à base de (DP) afin qu'il puisse reconnaître le réseau voirie dans le centre-ville de Blida (ville cas d'étude) sur les cartes satellitaires, car ce dernier (réseau routier) est un facteur très important pour l'étude de la typo-morphologie des villes car les urbanistes passe entre autre par l'étude de ce réseau voirie.

La reconnaissance des réseaux routiers est importante car elle permet d'analyser le tissu urbain selon la théorie de Muratori et Caniggia en contribuant à la détermination les axes structurants dans une ville.

Il est noter que l'automatisation de la lecture des cartes de villes est d'une grande aide pour les urbanistes pour traiter un nombre important de cartes [en un temps réduit].

Ce travail peut être également exploité dans d'autres applications telle que l'application d'optimisation utilisée dans les GPS (par exemple).

Pour réaliser ce projet, on a choisi comme modèle de segmentation d'objet le modèle U.net construit à base de réseaux CNN, qui a la capacité de traiter les données en temps réel, aussion utilise une base de données composée d'une variations des réseaux routiers pour le type de carte satellitaires, sur la plateforme Colab et à l'aide du langage de programmation Python et ses bibliothèques (notamment Keras) et bien beaucoup d'autres outils, Ceci va nous permettre d'avoir un modèle qui peut reconnaître les réseaux voirie.

Pour décrire notre projet nous avons organisé ce mémoire comme suit :

Dans le Chapitre I

Dans le premier chapitre on présente quelque notion sur la morphologie de ville, et aussi on va présenter la méthode d'interprétation Muratori et Caniggia, les aspects historique

Introduction générale

sur le centre de Blida et les types des cartes .

Ensuite

Dans le deuxième chapitre on s'approfondit dans l'Intelligence Artificielle notamment dans le Machine Learning et le Deep Learning et définir notre modèle U.net.

Finissons par

Le dernier chapitre est consacré à l'application de plusieurs outils (Google Colab, Google Drive...), pour la création et l'entraînement du modèle U.net sous Python, pour qu'il puisse segmenter les images satellitaires et extraire les réseaux voirie.

Chapitre I :

L'aspect urbain de la ville et extraction d'objets urbain.

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

I.1 Introduction

Dans le cadre de ce travail on procède de former un system d'aide à la reconnaissance d'un réseau de voirie inclus dans une carte urbaine en utilisant les techniques du Deep Learning.

Les system d'aide contribué à aider les urbanistes à situer rapidement et automatiquement les parcours et les centres ancien.

Le but ce chapitre est donner les grandes lignes de la méthode Muratorienne et son développement à travers le modèle interprétatif de Caniggia.

I.2 La méthode de Muratori [1] [2]

La méthode de lecture du tissu urbain examiné dans le but de calculer les paramètres (calcul du nombre de parcelles et mesure des côtés de l'îlot) pour la modélisation et la méthode typo- morphologique développée par l'école Muratorienne. Cette structure remonte à la naissance d'anthropologue italien l'architecte Saverio Muratori. Contrairement à la structure considérant l'architecture comme art pur, celui-ci ne peut pas être définie par des lois, car elle dépend essentiellement de l'individualité et de l'inspiration de l'architecte. La méthode Muratorienne propose de définir des disciplines propres à l'architecture à partir d'observations permanentes fondées sur des observations objectives de l'évolution des phénomènes urbains.

1-il n'y a pas de réalité non structurée.

2-la genèse de l'habitat est une genèse logique, ou logiquement restructurable.

Depuis les années 1950, Muratori s'intéresse à la critique du mouvement moderne où il observe les carences et dérives du mouvement moderne et les qualités qui manquent aux villes contemporaines : racines, mémoire collective, génie de lieu.

Cette hypothèse s'appuie sur deux postulats :

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

Si Muratori a pris position contre les objectifs antihistoriques et anti-urbains du mouvement moderne, c'est parce qu'il a saisi les intérêts culturels fondamentaux de la continuité des établissements humains, et qu'il doit également utiliser les vieux bâtiments comme infrastructure pour en obtenir de nouveaux bâtiments puissent apparaître et se développer.

Par conséquent, les attitudes individualistes des architectes, des dirigeants et des créateurs solitaires ont causé la perte du sens complexe architectural. La production urbaine de banlieues sans vie et de zones résidentielles composées de dortoirs sont un exemple.

Dans le but d'éviter cette situation, il est revenu sur les réalités concrètes des villes historiques, et a commencé à comprendre ces réalités pendant sa transition.

Muratori et ses élèves ont mis en évidence le point commun entre la typologie des bâtiments et la forme urbaine pour comprendre les phénomènes urbains. Ils traitent ces problèmes à travers une méthode structurée (basée sur la reconnaissance des structures), à travers laquelle le niveau structurel global de la ville peut être observé.

D'après cette approche, à la première lecture, les bâtiments ne sont pas considérés comme des objets isolés, mais comme des objets non-bâti (terrains, parcelle, lieux publics). La parcelle sera considérée comme l'élément de base de notre reconnaissance de forme.

Au autre niveau de lecture, le regroupement des parcelles permet de considérer la structure caractéristique des éléments du tissu en fonction de leur localisation dans la ville, de leur période de formation et de leur mode de développement.

De ces analyses Muratori tire trois leçons essentielles :

- a) Le type bâti ne se caractérise pas en dehors de son tissu.
- b) Le tissu urbain à son tour ne se caractérise pas en dehors de l'ensemble de la structure urbaine.
- c) L'étude de la structure urbaine ne se conçoit que dans sa dimension historique car sa réalité se fonde dans le temps sur une succession de réactions et de croissances à partir d'un état antérieur.

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

Ces constats largement révélateurs des idées architecturales contemporaines et ont contribué à la reconstruction des valeurs effacées par le mouvement moderne : Le rapport entre l'architecture et les villes et l'histoire.

I.3 Le modèle interprétatif de Gianfranco Caniggia [2]

Gianfranco Caniggia a été assistant de Muratori au département d'architecture de l'Université de Rome, dans la continuité de ses idées. Il a développé un modèle explicatif pour lire l'analyse de la structure urbaine. C'est sur ce modèle que nous pouvons développer nos projets.

L'hypothèse globale de Caniggia est de considérer l'organisation interne de l'environnement bâti comme une influence de son processus de formation.

Il a construit sa théorie sur la base de différentes échelles d'observation. Après avoir examiné la relation entre les emplacements des villes, commencez par le noyau d'origine. Il a commencé par une comparaison de divers cas de mutation dans des types de bâtiments italiens et a approfondi ses recherches sur les structures urbaines de base. Il a observé la formation et la rénovation des maisons de Florence de la basilique romaine.

La ville peut être considérée comme un immense laboratoire, qui expérimente depuis des siècles les formes urbaines. Le processus de construction progressive de la structure urbaine est appelé processus de transformation structurelle.

Pendant des centaines d'années, en raison des changements la dynamique de la vie humaine, il a connu une série de mouvements continus. Cela se traduit par une dynamique de forme urbaine.

Surtout dans les anciens centres, cette dynamique apparaît au niveau des trois lois fondamentales de la structure urbaine :

- La modularité.
- La nodalité.
- La hiérarchie.

I.3.1 Notion de modularité

Le concept de modularité ou d'agrégation modulaire de la structure de l'habitat est très important car il peut tirer plusieurs ordres de grandeur à partir du multiple ou de la division du module de base. Ensuite, une opération simple peut être utilisée pour décrire le

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

processus de construction de la croissance et de la diversification : répétition continue, séries unifiées d'agrégation et fusion.

Dans la terminologie de Caniggia le module de base a la double signification de module élémentaire et de module origine.

I.3.2 Notion de nodalité

Un nœud fait référence à tout point qui représentant un objet continu, qui est généralement déterminé par l'intersection de deux objets continus ou le bijou d'un objet continu et d'un autre objet : un nœud de corde ou deux nœuds de corde, qui expriment bien Cette fonctionnalité. Par conséquent, si une personne comprend un itinéraire à travers des objets continus, le nœud sera l'intersection entre les deux itinéraires, ou la collision entre deux objets continus, par exemple, un pont ou Ford, qui est liée à un parcours

I.3.3 Notion de hiérarchie :

Des itinéraires dans la ville permettent de connecter les différents points de cette dernière. On peut les catégoriser selon leur localisation et l'ordre d'importance qui leur est donné par les fonctions qu'il desservent : on aura donc des vois principales, des voies secondaire et des voies tertiaires ; dans les terminologie de Caniggia, ces voies sont appelé : parcours matrice , qui est à l'origine de la création de la ville ; puis parcours d'implantation, qui positionnera les nouvelles extensions perpendiculairement à la ville ; puis parcours de liaison, qui relie les parcours d'implantation entre eux.

Et enfin desserte, qui sont les voies et impasse menant aux maisons dans les zones résidentielles.

Pour notre travail, nous en retenons a partie de la théorie de G. Caniggia :

- a) Que la structure du tissu soit modulaire et que le module de base est la parcelle.
- b) Que les parcours sont hiérarchisés.

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

- c) La relation entre les parcours et les parcelles nous permet de définir les parcours structurants suivant la loi qui stipule que le parcours structurant présente un feu grand nombre de parcelles.

I.4 Définition de paramètres issus du modèle interprétatif

On va définir les principales parties du système d'aide à la reconnaissance des tissus urbains (fig.I.1).

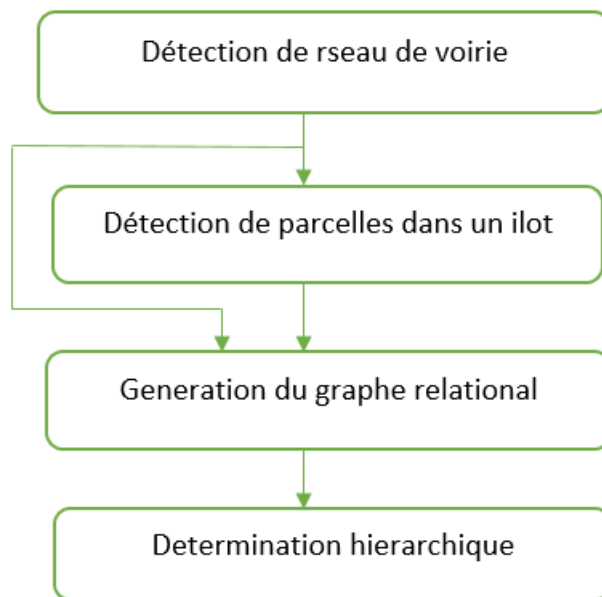


Figure I.1 : organigramme du système d'aide à la reconnaissance des tissus urbains.

I.4.1 Définition d'un Réseau de voirie

Un réseau routier (ou de voirie) est un ensemble de voies et d'intersections, dont les côtés sont composés par les frontières des ilots. Ainsi il sépare les défèrent bloc urbain que les parcelles sont incluses dans elle, Il relie les différentes régions de la ville entre elles la figure (fig I.2) montre un réseau de voirie indique par le gris.

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbains



Figure I.2 : réseau routiers.

I.4.2 Définition d'une Parcelle

C'est la composante de base d'un îlot, qui peut prendre plusieurs formes. Le développement de la structure urbaine peut être déduit selon leur quantité et leur taille. Un ensemble des parcelles peut être considéré comme un « parcellaire » (fig I.3).

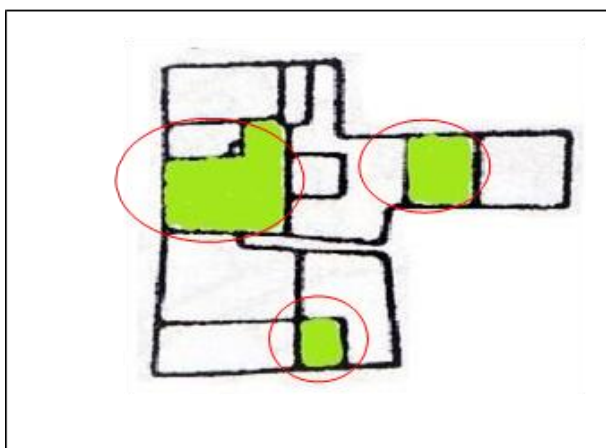


Figure I.3 : Exemple de placement des parcelles dans un îlot.

I.4.3 Définition d'un Ilot

Ilot est un bloc de maison, ensemble de parcelles, bâties ou non, constitue l'unité de base de l'espace urbain, il possède trois frontières (des cotés) ou plus, et chaque limite est délimitée par un réseau de routier (des voiries). Elle fait donc partie du tissu urbain limitée par l'intersection des voies autour de ses cotés (fig I.4).

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

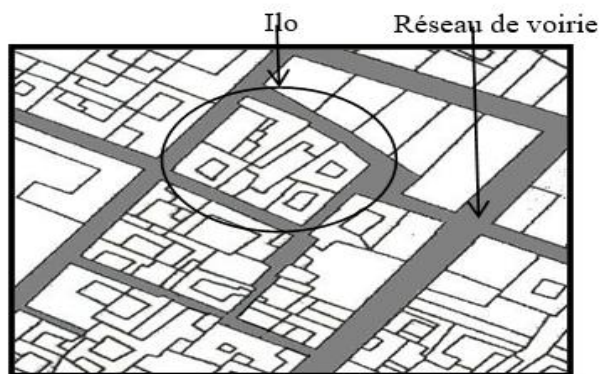


Figure I.4 : Représentation d'un îlot dans un bloc urbanisme.

I.5 Cas d'étude «centre ancien de Blida » [3] [4]

I.5.1 Situation géographique :

La ville de Blida est située au sud-ouest d'Alger et s'est développée en fonction de sa position stratégique dans l'aménagement du territoire de la région. En fait, c'est une ville carrefour, vitale pour les échanges est-ouest de l'Algérie.

C'est la capitale de la plaine de la Mitidja, et l'agriculture a toujours été à la base de son développement

Blida a été fondée par Sidi Ahmed EL Kébir sous l'Empire ottoman, c'est une ville de garnison et représente la puissance turque dans la laine de Mitidja. C'est aussi une importante station relais entre Alger et Titrerai (quartier de Médéa). Une petite ville appelée « el Blidah » est entourée d'un mur d'enceinte, qui est entouré d'un mur d'enceinte de 3 à 4 mètres de haut avec 06 portes.

Quelques années plus tard, la ville se développe rapidement et la place de d'Arme devient le centre des puissances européennes (théâtre, mairie, poste, banque, architecture, etc.), qui symbolise le nouvel ordre administratif et économique, c'est-à-dire la façade du bâtiment a remplacé le bâtiment du XIe siècle Les façades aveugles des maisons mauresques (andalouses) le long de la route goudronnée, la ville enferme finalement toutes les surfaces libres à l'intérieur de la clôture (sauf installations militaires, barrières de croissance).

Actuellement Blida ville des roses est un grand centre administratif, industriel, ses offres de services elle dépasse ses limites administratives pour atteindre ou dépasser les wilayets limitrophes.

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbains

I.5.2 Croissance urbaine du centre ancien de Blida

a) Période pré turque

Selon le colonel Trumelet près de Blida, vivent ici des tribus berbères dont les plus importantes sont les Beni-Khalil au sud et Hajar Sidi Ali au nord. Il y a aussi des tribus comme les Beni. Les gens vivent aussi dans les montagnes. La tribu Salah est divisée en plusieurs parties, divisées en villages. Le sultan de l'Oued et Hedjar Sidi Ali vivaient dans les Gulbis de pierre, entourés de cultures maraîchères, tandis que les montagnards étaient installés dans de petits villages sur les pentes de la vallée.

b) Période médiévale

Blida a été fondée par un vieux Marabout de l'Est. Il a beaucoup voyagé dans les pays islamiques, notamment en Andalousie (Espagne). Il s'appelle Sidi Ahmed · Kibir (sidi Ahmed elkébir), il a vécu dans l'oued Taberkachent et le chaàbet Ar Roman vers 1519, dans nos jours appelé "oued sidi El Kabîr".

L'ermitage de Sidi Ahmed El kebir fut bientôt entouré de mosquées et devint rapidement un lieu de pèlerinage pour de nombreux citoyens, attirés par les enseignements du saint et sa réputation de vertus et de miracles.

Vers 1533, après la capitulation de la Grenade, Le Pacha Kheir Eddine déporta des milliers de Maures d'Espagne et l'Algérie fut dispersée entre Alger et Cherchell.

La ville se protège de remparts en prisé de 3 ou 4 mètres de hauteur et percés de 4 pins, de 6 portes :

- Au sud : Bab Errahba (la porte du marché).
- A l'est : Bab Eddzair (porte d'Alger).
- A l'ouest : Bab el kébour (porte des tombeaux).
- Au nord : ab Essebt (porte du marché du samedi).
- Au nord-est : Bab Ezaouia (ouvrant sur la route de la zaouïa de sidi madjébeur).
- Au sud-est : Bab el khouikha (la poterne).

La structure résidentielle très dense se compose de petites maisons avec une cour intérieure et un toit en terrasse. À cette époque, la ville traversait 30 rues larges qui se rattachaient aux artères principales.

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

c) Période coloniale

Le 7 février 1839, l'armée française occupa la ville et occupa fermement le château.

En 1842, à la fin de la guerre de Mitidja, les premières familles européennes s'installent dans la ville peuplée. Des maisons de style européen ont commencé à être construites dans le centre, juxtaposées à des maisons de style mauresque. Les constructions coloniales, édifiées dans les villes européennes primitives se résument en trois signes de ville : L'ingénierie de défense, L'ingénierie religieuse et les espaces vides.

En 1846, le mur de pierre a remplacé l'ancien entrepôt a gravement endommagé par la guerre et a été remplacé par le mur de pierre, percé par la porte et situé bien au-delà du tour d'origine. En conséquence, la périphérie de la ville a presque étendu agrandie au profit de l'armée. Il représente la moitié de la zone urbaine et 20 hectares de 48 hectares.

Par la suite, l'occupation du centre-ville de Blida a également été caractérisée par des immeubles à forte densité, des bâtiments commerciaux, des bâtiments administratifs ou religieux (marchés, hôtels de ville, mosquées), qui collectent avec le premier noyau de la ville.

I.6 Présentation de la carte urbaine [5]

La cartographie est à la fois la science, la technique et l'art de réaliser et d'utiliser les cartes. La carte gère deux informations fondamentales : la position et l'information qui y est rattachée. Les informations rattachées (attributs), peuvent être un nombre, une activité, un taux, une quantité et elles évoluent avec le temps.

Différentes types de cartes fournissent des échantillons de cette large sélection, de sorte que ces cartes remplissent ainsi la fonction de présenter ces faits d'une façon réalisable.

Les cartes varient selon l'échelle, l'objectif ou le contenu peuvent être regroupées selon la typologie suivante :

a) Les cartes topographiques

Qui montrent les relations spatiales entre différents objets géographiques, tel que les bâtiments, les routes, les frontières et les rivières.

La plupart des communes produisent également des plans de ville. Ils existent également des cartes topographiques pour la randonnée ou les sports nautiques. De nombreux

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

systèmes embarqués de navigation routière et services internet fournissent également des cartes topographiques (fig I.5).



Figure I.5 : Carte topographies.

b) Les cartes spéciales

Ces cartes sont à usage professionnel et suivent les règles définies par l'Organisation des Nations Unies par exemple, les cartes maritimes ou aéronautiques. Il existe aussi des cartographies spécifiques à la navigation de plaisance ainsi que des cartes dédiées à l'orientation définies par la Fédération internationale de course d'orientation (fig I.6).

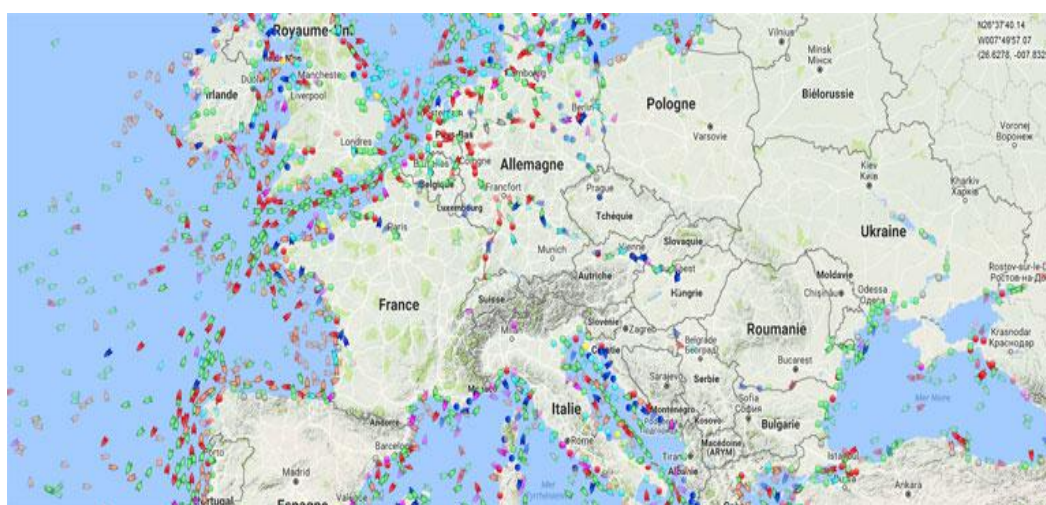


Figure I.6 : Carte maritime.

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

c) Les cartes thématiques

Utilisée pour présenter des données sur un thème spécifique décrivent des phénomènes géographiques tels que la géologie (sol et substrat rocheux essentiellement), à l'occupation du sol ou à la végétation. Les cartes statistiques font également partie des cartes thématiques. Elles montrent la répartition géographique d'une variable statistique (fig I.7).

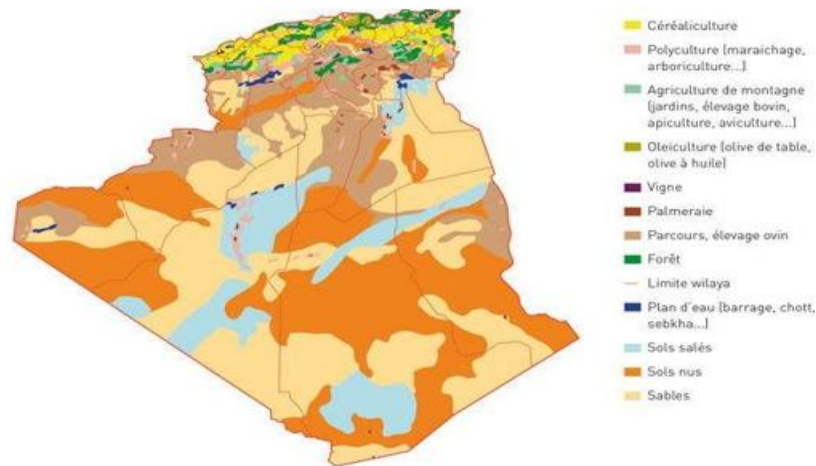


Figure I.7 : Carte thématique de sol.

d) Les cartes météorologiques

C'est les cartes thématiques les plus utilisées. Elles sont diffusées quotidiennement à la télévision tous les jours pour montrer les conditions météorologiques du jour et les prévisions pour les prochains jours (fig I.8).

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbains

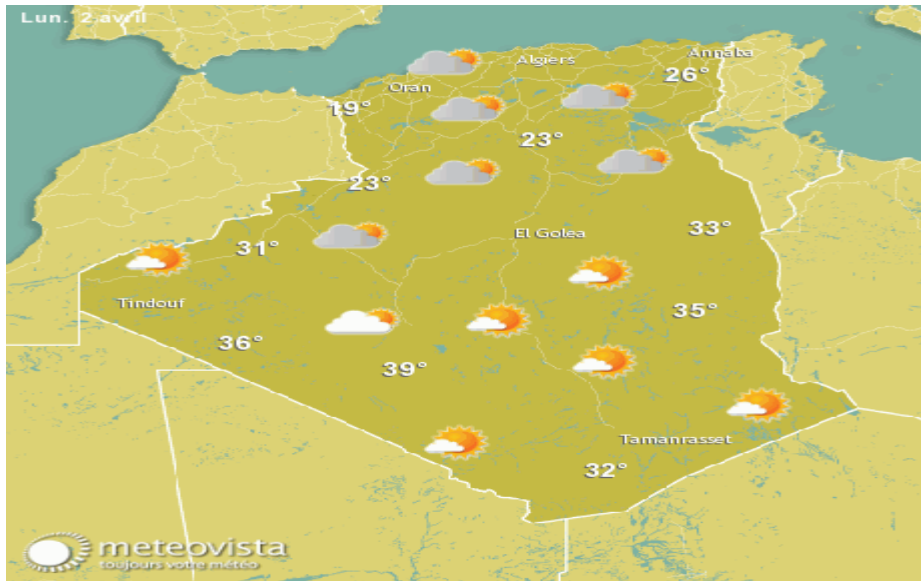


Figure I.8 : Carte de météo.

e) Les cartes géologiques

C'est une partie de la carte thématique et sont inestimables pour la prospection de minerais ou de pétrole, ou pour analyser le potentiel agricole. Elles fournissent de nombreuses informations compliquées et les résultats d'études universitaires en géologie sont accompagnés de plusieurs planches cartographiques, très utiles lorsqu'il faut prendre des décisions géotechniques ou ayant une incidence sur l'environnement. (fig I.9).

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

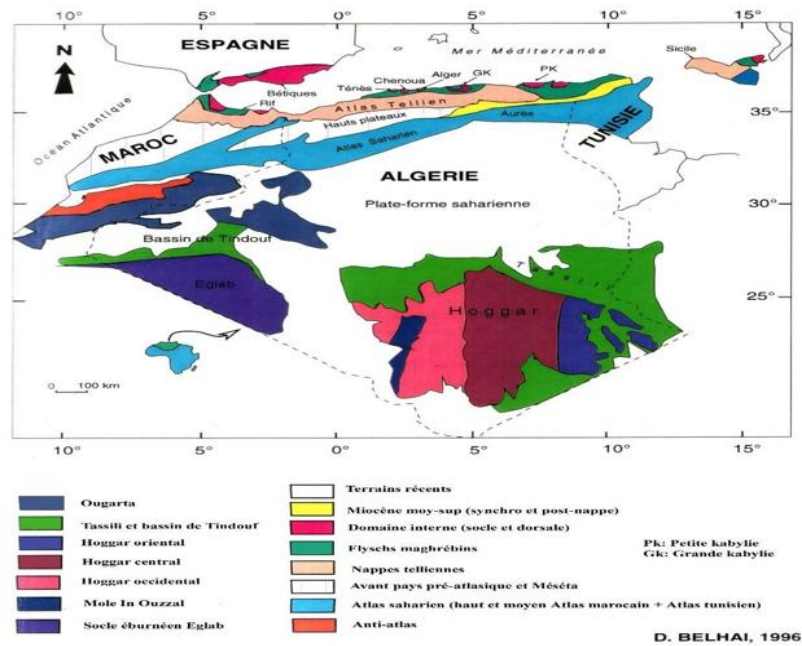


Figure I.9 : Carte géologique de l'Algérie.

I.6.1 Cartes urbaines cas d'étude

Les cartes urbaines peuvent être utilisées pour faire de la reconnaissance d'objets urbains, le but est extraire réseaux de rues à partir des images satellitaires pour la réalisation d'un système d'aide de centres anciens.

Dans notre travail on s'intéresse par les images satellitaires, on essaiera de d'extraire des réseaux routiers en utilisant le Deep Learning (Fig I.10).



Figure I.10 : Image satellitaire d'une zone urbaine.

Chapitre I : L'aspect urbain de la ville et extraction d'objets urbain

I.7 Conclusion

Après toutes les études faites sur la morphologie urbaine et les critères de connaissance. Il est possible de créer un programme capable d'extraire réseau routier urbain à partir des cartes satellitaires.

Notre étude représente seulement un aspect de lecture d'un tissu urbaine afin de créer un system supervisée à l'aide de Deep Learning qui reconnais les réseaux voirie.

Ce projet s'intéresse à la détection du réseau routier urbain pour son utilité et la diversité de ses applications.

Chapitre II

Généralités sur le Deep Learning : les outils et le modèle.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

II.1 Introduction

Le Deep Learning est un nouveau domaine de recherche du ML, qui a été introduit dans le but de rapprocher le ML de son objectif principal. Le but de ce travail d'utiliser le deep learning pour former des données. On va présenter tout d'abord les notions en relation avec l'apprentissage profond, le réseau de neurones et sa topologie, CNN et ses différentes couches et algorithmes de détection

II.2 L'intelligence artificielle (IA) [6]

L'IA est un ensemble de techniques et théories qui cherchent à développer des modèles capables de **simuler** le comportement humain (raisonnement, apprentissage...).

Le pouvoir de simuler l'intelligence humaine et le comportement d'un être humain doté de conscience et de sentiments c'est le but des scientifiques dans le domaine de l'IA

Parmi les techniques de l'IA, on trouve le **Machine Learning** (L'apprentissage machine).

II.3 Machine Learning (ML)

Le ML désigne la science de la création et de l'étude d'algorithmes qui améliorent leur propre comportement de manière itérative par la conception (figure II.1).

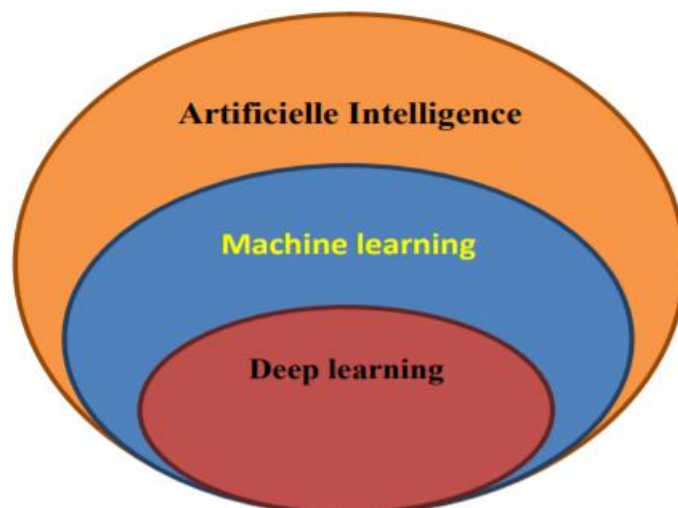


Figure II.1 La relation entre l'intelligence artificielle, le ML et le deep learning.

Le domaine était dédié au développement de l'intelligence artificielle, mais en raison des

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

limitations théoriques et technologiques existantes, il est devenu plus logique de concentrer ces algorithmes sur des tâches spécifiques. La plupart des algorithmes (ML) existants aujourd'hui se concentrent sur les fonctions d'optimisation.

Par conséquent, l'utilisation d'algorithmes (ML) devient souvent un processus répétitif d'essais et d'erreur, ou le choix d'un algorithme parmi les problèmes produira des résultats de performance différents. C'est très bien dans certains cas, mais dans le cas de la modélisation du langage et de la vision par ordinateur, cela devient problématique.

Suite à l'avancement des compétences théoriques et technologiques dont nous disposons aujourd'hui, l'apprentissage approfondi (Deep Learning) est apparu et évaluer rapidement pour devenir l'un des domaines scientifiques les plus passionnants. Il est utilisé dans des technologies telles que la reconnaissance d'images sur les plateformes de médias sociaux et la traduction de textes d'une langue à une autre.

II.4 Deep Learning (DL) [7] [8]

Deep Learning est un ensemble de méthodes d'apprentissage automatique (Machine Learning) tentant de modéliser avec un haut niveau d'abstraction des données grâce à des architectures articulées de différentes transformations non linéaires. Avec tels techniques ont provoqué des progrès importants et rapides dans les domaines de :

- l'analyse du signal sonore ou visuel et la reconnaissance faciale.
- la reconnaissance vocale et la vision par ordinateur.
- traitement automatisé du langage.

L'apprentissage en profondeur est une branche particulière du l'apprentissage automatique. Le processus de Machine Learning commence par extraction des caractéristiques pertinentes des images. En s'appuyant sur ces caractéristiques, un modèle qui classe les objets de l'image est ensuite créé. Dans le processus de Deep Learning, l'extraction des

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

caractéristiques pertinentes à partir d'images est automatique. De plus, le Deep Learning effectue un apprentissage « de bout en bout » : à partir de données brutes, un réseau se voit attribuer des tâches à accomplir (une classification, par exemple) et apprend à automatiser ces tâches.

Le Deep Learning est un domaine à croissance rapide, et de nouvelles architectures, variantes ou algorithmes apparaissent toutes les semaines. Basant sur les neurones, nous allons montrer deux notions, le neurone biologique, et le neurone l'artificiel.

II.4.1 Neurone biologique

Un neurone comprend un corps cellulaire ou cellule somatique, En fait, à cause de ses dendrites, les neurones recevront des informations d'autres neurones. Ensuite, le corps cellulaire agrège toutes les informations. Puis il traite les informations et renvoie le résultat sous forme de signaux électriques l'entrée des autres neurones.

Le Neurone biologique est constitué par :

- a. Dendrites : ce sont des prolongements du corps cellulaire des neurones. Elles se divisent créant une arborescence du neurone sous forme de filaments courts et ramifiés, selon leur longueur, ils affectent la quantité d'influx nerveux qui se rend au noyau, ils représentent les entrées du neurone.
- b. Axones : ils relient les neurones entre eux jouent donc un rôle important dans le comportement logique de l'ensemble. Les synapses du neurone quant à eux reçoivent les informations des autres neurones via l'axone et permettent donc aux neurones de communiquer entre eux.
- c. Noyau : Le noyau est le centre des réactions électrochimiques. Si les stimulations externes sont suffisantes, le noyau provoque l'envoi d'un influx nerveux électrique à travers l'axone.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

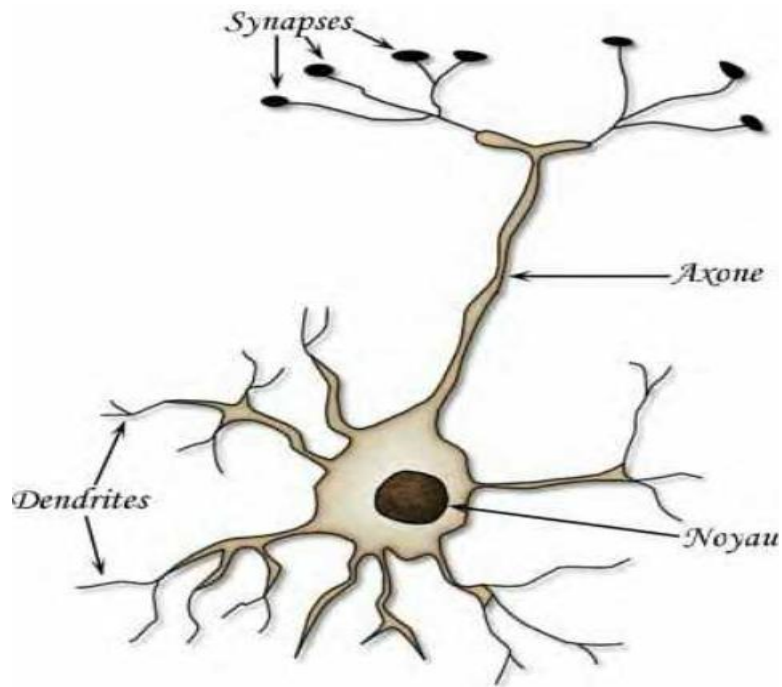


Figure II.2 : Neurone biologique.

II.4.2 Neurone artificiel [9]

Le neurone formel est l'unité de base du réseau de neurones artificiels (ARN) et se compose de deux parties :

- Évaluer le stimulus reçu (Fonction E)
- Evaluer son activation (fonction f) dont les caractéristiques sont :
- Son état X (binaire, discret, continu)
- Niveau d'activation reçu à l'entrée U
- W Saisir le poids de la connexion

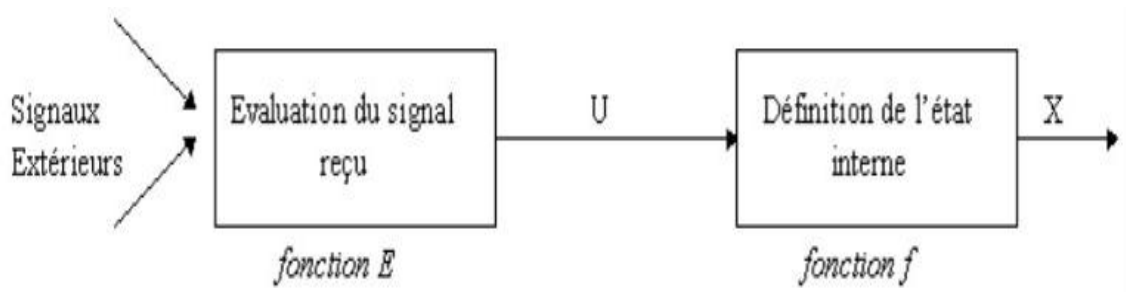


Figure II.3 : les éléments d'un neurone formel.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

II.4.3 Le perceptron

Un modèle utilisé pour la classification binaire avec une simple relation entrée-sortie. Les entrées au perceptron sont indépendantes, donc il suffit d'étudier la sortie.

1. Le neurone calcule la somme z de toutes les entrées $z = \sum x\theta$. un calcul linéaire.
2. Le neurone introduit z dans la fonction d'activation qui est la fonction sigmoïde. un calcul non-linéaire qui génère la sortie y

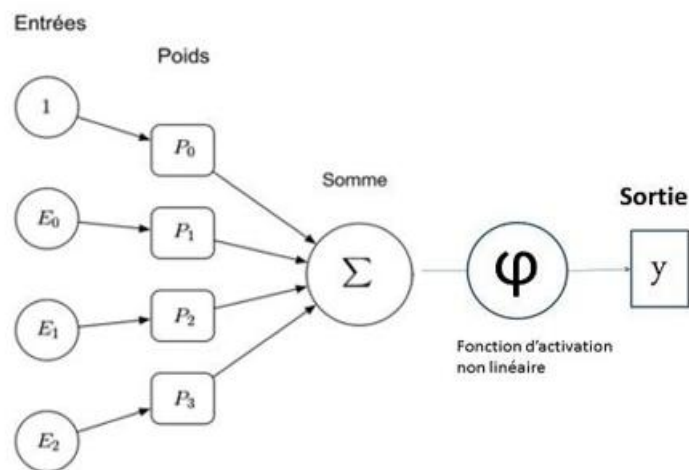


Figure II.4 : Représentation du perceptron.

- Chaque entrée possède un poids.
- La sortie est une fonction du poids et des entrées
- Il est doté d'une couche dédiée à la perception et une couche dédiée à la prise de décision.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

II.4.4 Perceptron multicouche

Le PMC est un réseau de neurones de plusieurs perceptrons connecter un aux autres d'une façon particulière dans laquelle une information circule uniquement de la couche d'entrée vers la couche de sortie.

- On regroupe les neurones en colonne (on dit qu'on les réunit en couche, en *layer*). dans leur colonne, les neurones ne sont pas connectés les uns aux autres.
- On connecte toutes les sorties des neurones d'une colonne à gauche aux entrées dans tous les neurones de la colonne de droite qui suit.

La figure (II.5) représente un réseau de neurones. Tous les *layers* entre la couche d'entrée et la couche de sortie nous n'avons pas accès à leur entrées/sortie donc appelés cachés, qui sont utilisées par les *layers* suivants.

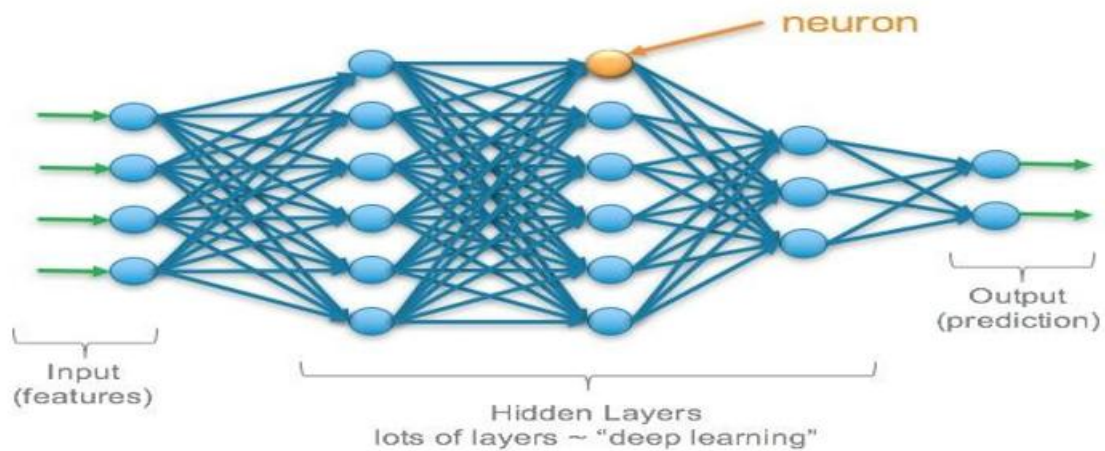


Figure II.5 : Réseau neurone (MLP).

On peut construire un réseau avec un nombre quelconque de couches et de neurones. Plus il y a de couches, on dit que plus le réseau est profond (profond), plus le modèle devient riche, et difficile à entraîner. C'est un apprentissage en profondeur (deep learning).

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

II.5 L'Apprentissage [10] [11]

Comme le cerveau humain, les réseaux de neurones artificiels peuvent apprendre par expérience. Dans cette section, nous verrons comment les réseaux de neurones peuvent apprendre est minimiser les erreurs grâce à des techniques d'apprentissage.

L'apprentissage est une étape du développement du réseau de neurones au cours de laquelle le comportement du réseau est modifié jusqu'à ce que le comportement souhaité soit atteint. On distingue quatre grandes classes d'apprentissage :

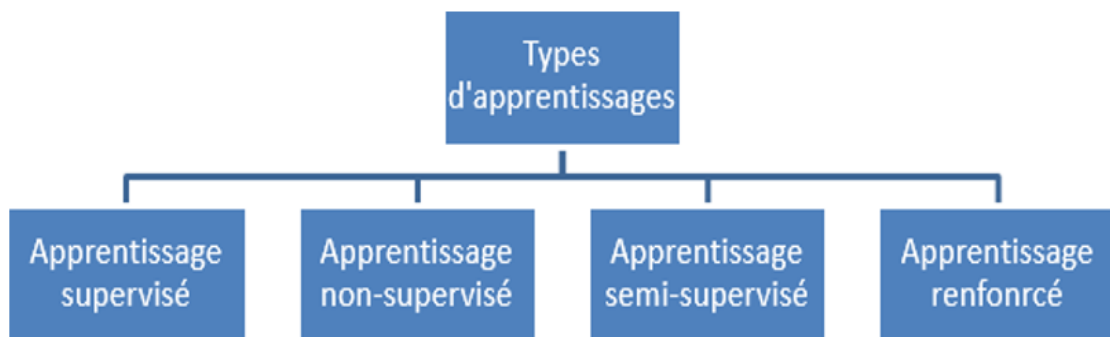


Figure II.6 : Schéma synoptique des types d'apprentissage.

II.5.1. L'apprentissage supervisé

On leur fournit des données dites annotées pour entraîner le modèle, ce qui signifie qu'elles sont déjà associées à un label ou une classe cible et on veut que l'algorithme puisse prédire cette cible après l'entraînement des données non annotées.

L'algorithme le plus connu pour l'apprentissage supervisé est le gradient descendant. Ce dernier est l'erreur relative à ce neurone.

Il peut en quelque sorte être vu comme la contribution du neurone à l'erreur globale.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

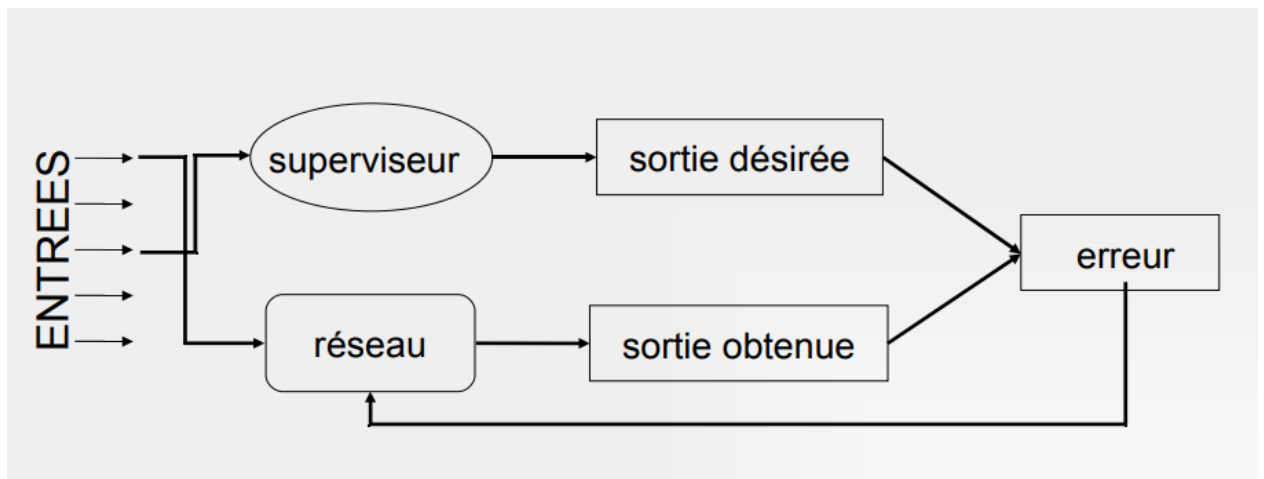


Figure II.7 : Apprentissage supervisé.

Puisque nous devons reconnaître des objets et classer des images, nous avons besoin d'un modèle d'apprentissage supervisé.

Etant donné que l'apprentissage supervisé est une méthode d'utilisation des données d'entraînement étiquetées pour former un modèle, il peut ensuite être généralisée à de nouveaux exemples.

II.5.2. L'apprentissage non supervisé

Appelé apprentissage à partir d'observation, Pour des entrée données. Il n'y pas de connaissances préalable pour des sorties désirées. En fait, il s'agit d'un apprentissage par exploration dans lequel l'algorithme d'apprentissage ajuste les poids des liens entre neurones pour maximiser la qualité de classification des entrées (fig II.8).

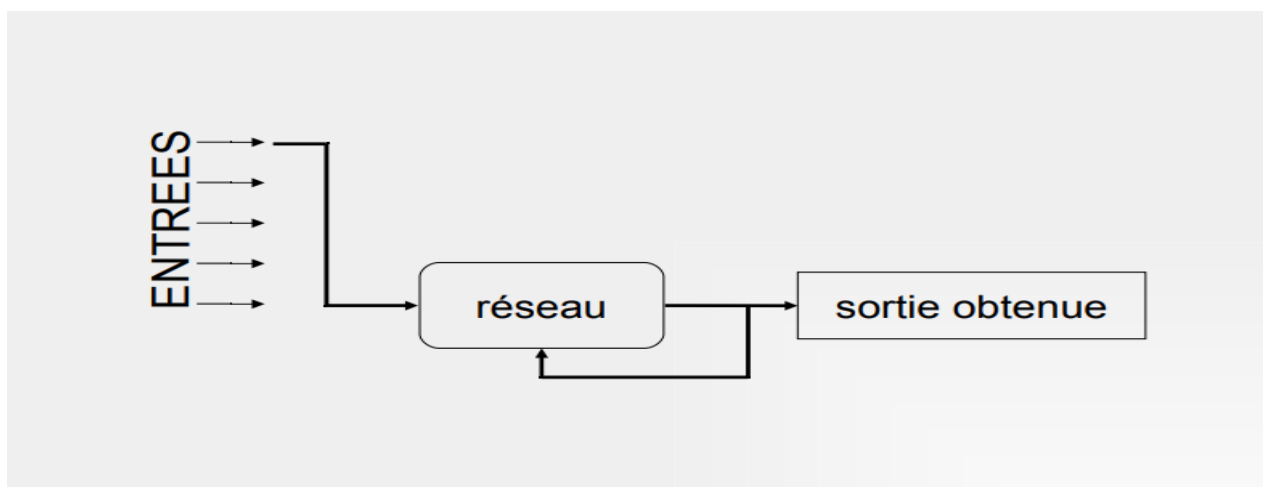


Figure II.8 : Apprentissage non supervisé.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

II.5.3. L'apprentissage semi supervisé

Il prend en entrée certaines données annotées et d'autres non. Ce sont des méthodes très intéressantes qui tirent parti des deux mondes (supervisé et non-supervisé), mais apportent leur lot de difficultés.

II.5.4. L'apprentissage renforcé :

Il s'appuie sur un cycle d'expérience, récompense et améliore les performances à chaque itération. Une analogie souvent citée est celle du cycle de dopamine, une "bonne" expérience augmente la dopamine et augmente ainsi la probabilité que l'agent répète l'expérience.

II.6 Les fonctions d'activation

II.6.1 La fonction ReLU (Rectified Linear Unit)

Cette fonction, appelée aussi "fonction d'activation non saturante", pour convertir tout ce qui est inférieur à zéro en zéro. Ainsi la plage devient 0 à l'infini. La fonction d'activation est appliquée après la convolution et laisse la taille volume inchangé qui sert comme une couche de correction

La correction ReLU : $f(x) = \max(0, x)$

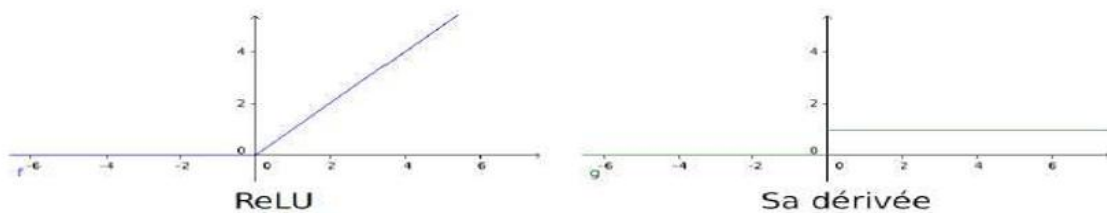


Figure II.9 : la fonction Relu et sa dérivée.

II.6.2 La fonction sigmoïde

Elle réduit la sortie entre 0 et 1. Le problème principal est que le gradient sur les queues est en train de disparaître. Ainsi, l'algorithme de rétro propagation modifie à peine les paramètres et l'apprentissage est ralenti.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.



Figure II.10 : Fonction d'activation Sigmoïde.

II.7 Les Modèles d'interconnexion des neurones [12]

Les connexions entre les neurones qui composent le réseau décrivent la topologie du modèle. Elle peut être quelconque, mais le plus souvent il est possible de distinguer une certaine régularité.

On a mentionné les deux topologies les plus simples (MLP, perceptron), on va présenter d'autres types.

II.8.1 Réseau à connexion complète

C'est la structure d'interconnexion la plus générale. Chaque neurone est connecté à tous les neurones du réseau à double sens (fig II.11)

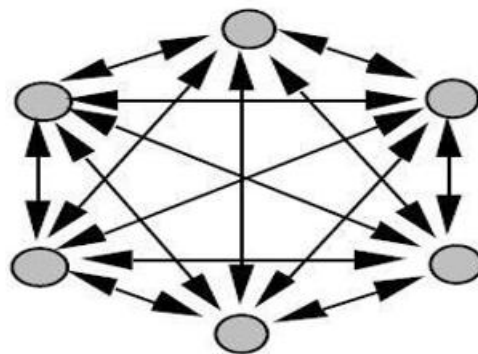


Figure II.11 : Réseau à connexion complète.

II.7.2 Réseau a connexion locales

C'est une structure multicouche, comme la rétine elle conserve une certaine structure topologique. Chaque neurone entretient une relation avec un petit nombre de neurones

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

locaux dans la couche aval (fig.12). Par conséquent, le nombre de connexions est inférieur à celui des réseaux multicouches classique.

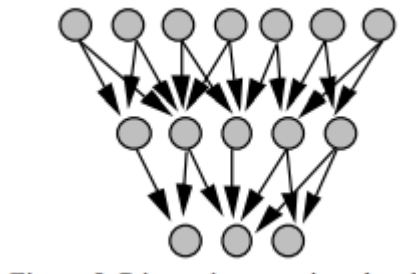


Figure II.12 Réseau a connexion locales.

II.7.3 Réseau à connexions récurrentes (RNN)

Les connexions récurrentes ramènent l'information en arrière par rapport au sens de propagation défini dans un réseau multicouche. Ces connexions sont le plus souvent locales (figure II.13).

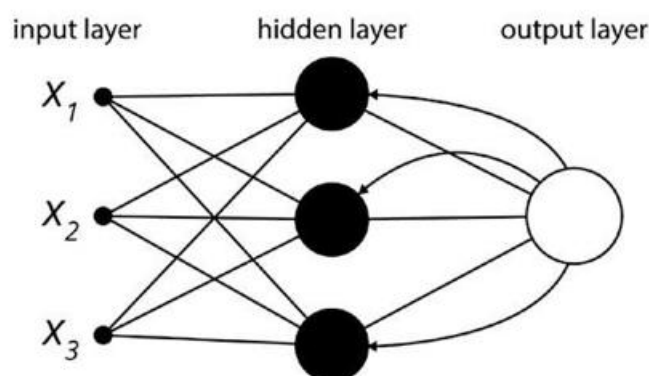


Figure II.13 : Réseaux à connexions récurrente.

II.8 Les Réseaux de neurone convolutifs (CNN) [13]

Les réseaux de neurones convolutés peuvent apprendre des fonctions de cartographie extrêmement complexes lorsqu'ils sont formés sur suffisamment de données. Au niveau de base, les poids d'un CNN sont constitués de filtres. Considérez un filtre comme une matrice ($n * n$) composée de certains nombres. Maintenant, ce filtre est alambiqué (fait glisser et multipliez) à travers l'image fournie.(FigII.14)

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

- Apprentissage des caractéristiques : contient deux couches (convolution et pooling).
- Classification : contient une couche entièrement connectée.

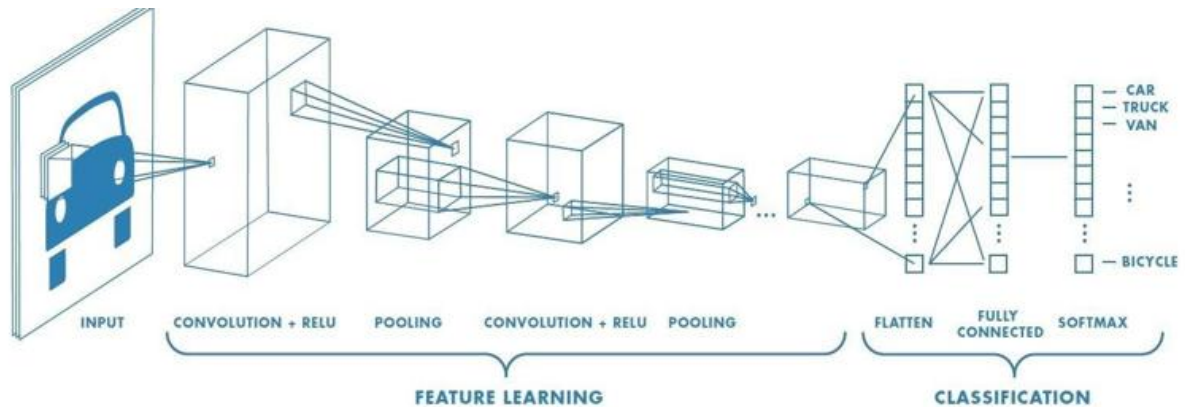


Figure II.14 : Différentes couches CNN.

Fondamentalement, la formation d'un CNN implique de trouver la valeur correcte sur chaque filtre de sorte que lorsque l'image d'entrée traverse plusieurs couches, elle active certains neurones de la dernière couche pour prédire la bonne catégorie.

II.8.1 Définitions de CNN

Un réseau neuronal convolutif ou (Convolutionel Neural Networks en anglais) un des types de réseaux de neurones d'apprentissage approfondi qui peut prendre en compte une image d'entrée, attribuer de l'importance (poids et biais appelables) à divers aspects ou objets de l'image et être capable de les différencier les uns des autres. Le prétraitement requis dans un CNN est beaucoup moins important à celui d'autres algorithmes de classification.

II.8.2 les couches utilisées dans un CNN

Ils sont constitués de plusieurs couches, cette dernière constituée de plusieurs neurones.

a) La couche de convolution (Convolution Layer) [14]

L'objectif de cette couche est de détecter les caractéristiques telles que les bords, les taches de couleur et d'autres éléments visuels, avec le filtre la convolution d'image crée des

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

images appelées cartes de caractéristiques de sortie, plus de filtres dans les couches de convolution plus des caractéristiques ont détecté.

CNN ont la capacité de mis a jour les filtres d'une manière systématique. Il n'y a aucune conception manuelle.(figII.15)

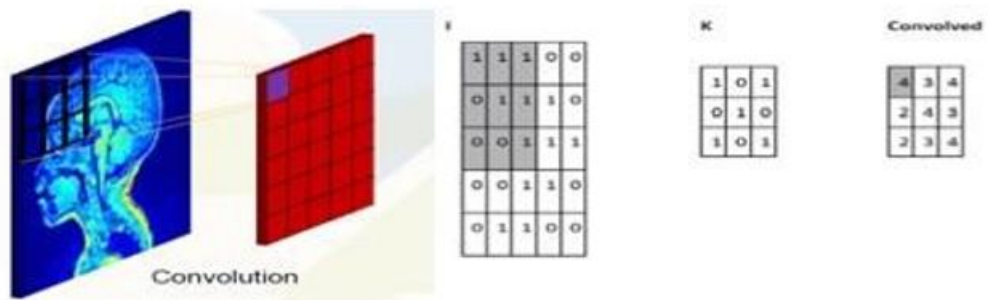


Figure II.15 : Exemple de produit de convolution.

b) La couche de mise en commun (pooling layer) [15]

Le pooling est un sous-échantillon de l'image, permettant à la couche suivante d'une zone spatiale plus grande de réduire la dimension (nombre de variables à traiter) mais elle rend également le réseau moins sensible à la traduction éventuelle de l'image sans affecter la profondeur.

Par conséquent, il examine les réponses des filtres à différents emplacements, permettant de pointer les emplacements des filtres qui donnent les mêmes réponses.

Il existe deux types de mise en commun (Figure II.16) :

- Mise en commun maximale (Max Pooling) : renvoie la valeur maximale de la partie de l'image couverte par le filtre.
- Mise en commun moyenne (average Pooling) : renvoie la moyenne de toutes les valeurs de la partie de l'image couverte par le filtre.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

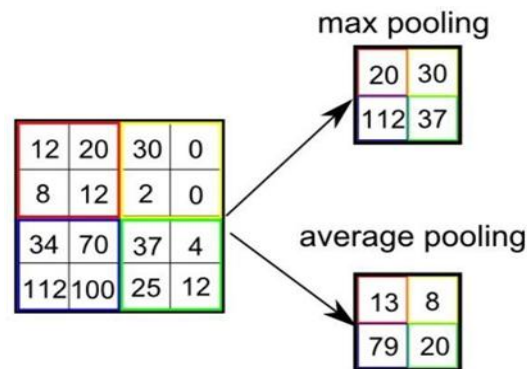


Figure II.16 : Les types de pooling.

c) Couche entièrement connectée (fully-connected) [16]

Après plusieurs couches de convolution de max-pooling, le raisonnement de haut niveau dans le réseau neuronal se fait via des couches entièrement connectées. Le neurone dans une couche entièrement connectées ont des connexions vers toutes les sorties de la couche précédentes, leur fonctions d'activation peuvent donc être calculés avec multiplication matricielles suivis d'un décalage de polarisation..

d) Couche de perte (LOSS)

La couche de perte spécifie comment l'entraînement du réseau pénalise l'écart entre les signaux attendus et réels. Il s'agit généralement de la dernière couche du réseau. Différentes fonctions de perte adaptées à différentes tâches peuvent être utilisées. La fonction « softmax » permet de calculer la distribution de probabilité sur les classes de sortie.

II.9 Réseaux de Neurones Convolutifs Régionaux (R-CNN) [17]

R-CNN est un modèle de détection d'objets son algorithme origine est un processus en quatre étapes :

- **Étape 1** : Saisir une image sur le réseau.
- **Étape 2** : Extraire des propositions de région (Des régions d'une image pouvant contenir des objets) à l'aide d'un algorithme.
- **Étape 3** : Utilisez l'apprentissage par transfert, en particulier l'extraction de fonctionnalités, pour calculer les fonctionnalités de chaque proposition à l'aide du CNN.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

- **Étape 4** : Classifiez chaque proposition à l'aide des fonctionnalités extraites avec une machine à vecteur de support (SVM).

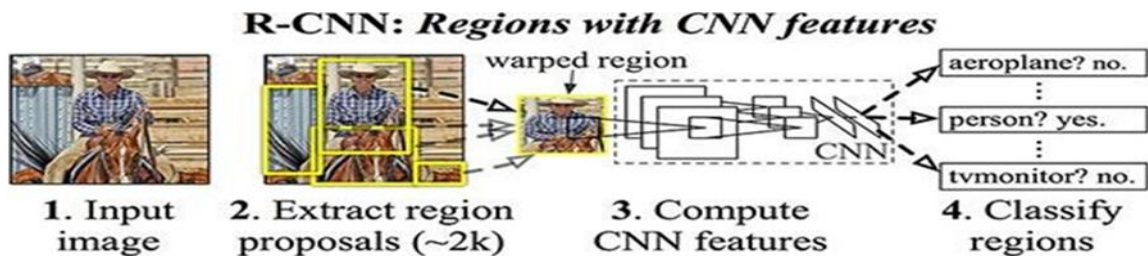


Figure II.17 : L'architecture R-CNN d'origine.

II.10. Fast R-CNN [18]

R-CNN rapide traite l'image entière contrairement au détecteur de R-CNN qui recadre et redimensionne les régions proposées, l'algorithme Fast R-CNN a été publié

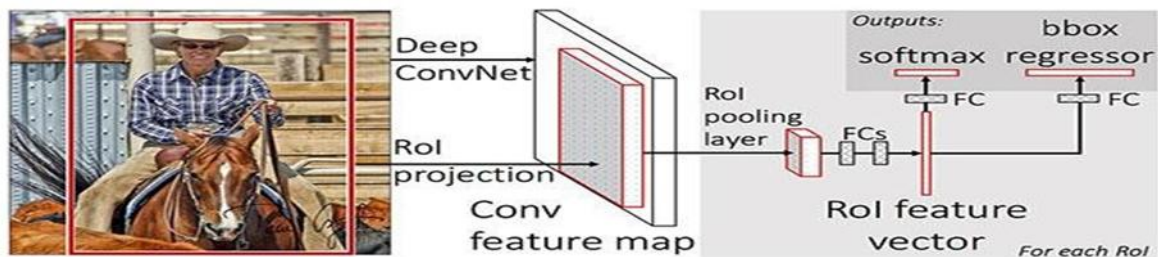


Figure II.18 : L'architecture Fast R-CNN.

Le ROI Pooling fonctionne en extrayant une fenêtre de taille fixe de « Features Map » et en utilisant ces entités pour obtenir l'étiquette de classe finale et le cadre de délimitation.

II.11. Faster R-CNN

Faster R-CNN s'appuie sur le modèle de bas Fast-R-CNN ce réseau de neurones convolutés supplémentaire permet de générer directement les propositions de zones.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

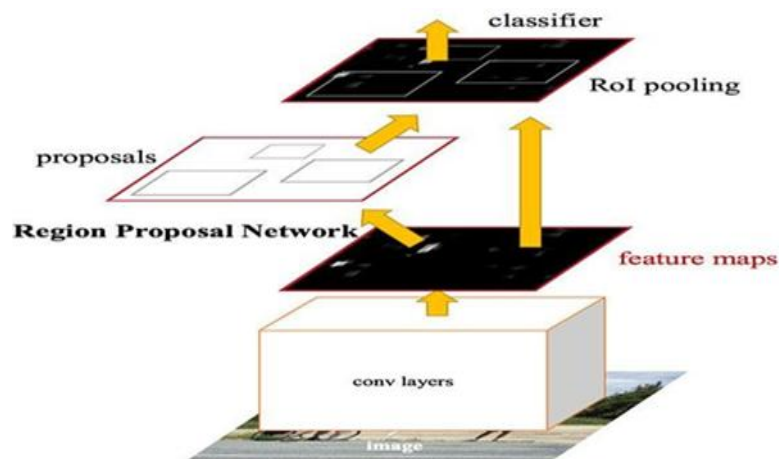


Figure II.19 : L'architecture plus rapide R-CNN « Faster R-CNN ».

II.12. Mask R-CNN

Le modèle Le masque R-CNN comme les modèles précédents permet de détecter et classer les objets, sa particularité est que la segmentation d'instance s'ajoute à cette tâche de détection, donc chaque pixel de l'image sera classé. Par conséquent, par rapport au modèle de détection, elle représente un avantage, qui améliore les résultats recommandés, de plus contrairement à la segmentation sémantique qui permet d'associer des étiquettes à chaque pixel, la segmentation d'instance associer des masques et des étiquettes à chaque objet.

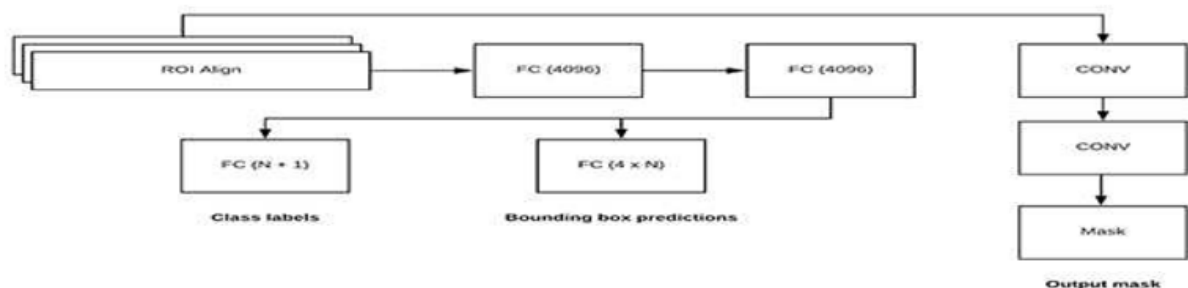


Figure II.20 : Mask R-CNN. La sortie des couches CONV est le masque lui-même.

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

II.13. Réseau de neurones de computer vision U-NET [19]

U-Net est un modèle de réseau de neurones spécialement utilisé pour les tâches de vision par ordinateur, en particulier les problèmes de segmentation sémantique. Le réseau est entièrement convolutionnel. C'est le modèle de neurones qu'on a utilisé dans notre projet

II.13.1 La segmentation sémantique [20]

L'objectif de la segmentation sémantique d'image est d'étiqueter chaque pixel de l'image avec une classe correspondante. Puisque nous faisons des prédictions pour chaque pixel de l'image, cette tâche est généralement appelée prédiction dense.



Figure II.21 : image et étiquette de pixel.

La segmentation sémantique d'image utile dans des applications de divers domaines (conduite autonome, imagerie médicale, contrôle industriels, imagerie satellite et aérienne...).

II.13.2 Architecture de U-NET

L'architecture de U-NET est composée de deux chemins, le premier appelé chemin de contraction (encodeur), le deuxième est le chemin d'expansion symétrique (décodeur).

a) Encodeur (côté gauche)

Il consiste en des applications répétées de deux convolutions 3x3. Après chaque convolution, il y a une ReLU et une normalisation de lot. Appliquer ensuite l'opération de pooling 2x2 max. pour réduire la dimension spatiale. De même, à chaque étape de sous-

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

échantillonnage, nous doublons le nombre de canaux de caractéristiques tout en réduisant de moitié la dimension spatiale.

b) Décodeur (côté droite)

Chaque étape de chemin d'extension comprend le sur-échantillonnage de la carte de caractéristiques, suivi d'une convolution transposée 2x2, qui divise par deux le nombre de canaux de caractéristiques. Nous avons également une concaténation avec la carte de caractéristiques correspondante du chemin de contraction, généralement une convolution 3x3 (chaque convolution suivie d'un ReLU). Dans la dernière couche, une convolution 1x1 est utilisée pour mapper les canaux au nombre de classes.

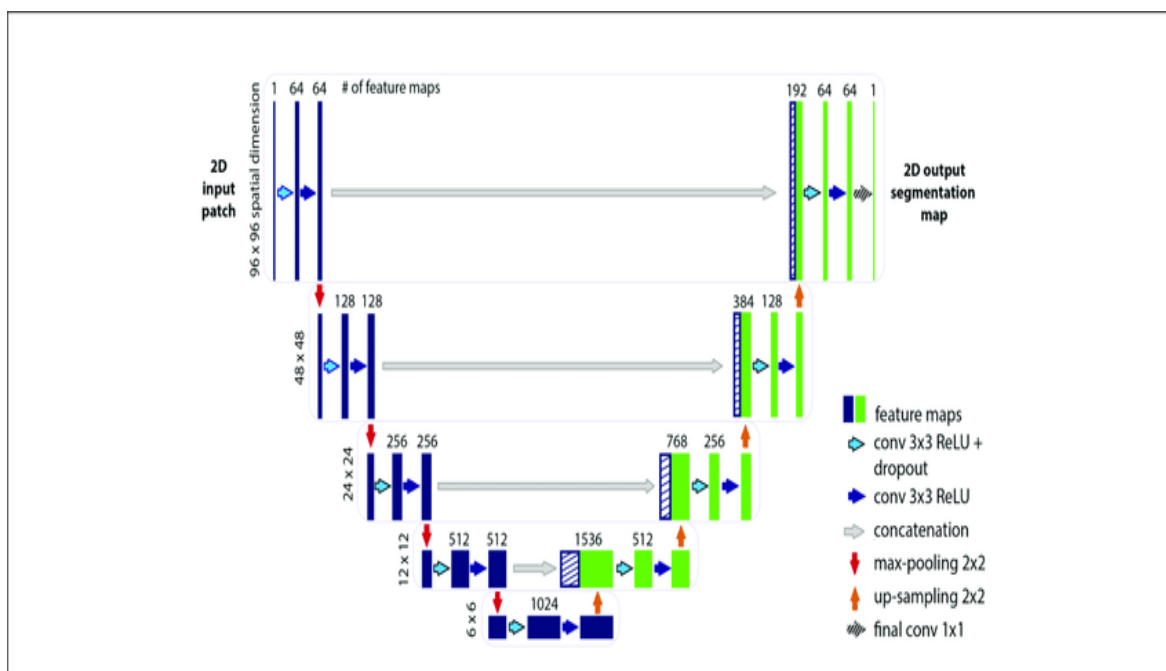


Figure II.22 : architecture de U-NET.

II.13.3 Les avantages de U-NET

Dans le domaine du Deep Learning, de larges ensembles de données est nécessaire pour entraîner les modèles. En termes de temps, de budget et de ressources matérielles, Il peut être difficile d'assembler un tel volume de données pour résoudre un problème de

Chapitre II : Généralités sur le Deep Learning : les outils et le modèle.

classification d'images. L'étiquetage des données, particulièrement est le cas pour des domaines hautement spécialisés comme les diagnostics médicaux.

U-NET résout ces problèmes, car il est efficace même avec un ensemble de données limité.

Il offre également une précision plus grande que les modèles conventionnels.

Grâce à cette architecture de réseau de neurones en forme de U, la déconvolution est effectuée côté décodeur, évitant le problème de goulot d'étranglement rencontré par l'architecture d'auto encodeur, évitant ainsi la perte de fonctionnalités.

II.14 Conclusion

Dans ce deuxième chapitre nous avons présenté les différents reconnaissances de base et modèles de deep learning, ainsi le model choisis dans notre projet.

L'apprentissage profond est le domaine le plus émergent de l'apprentissage automatique.

Dans le chapitre suivant, nous allons présenter les résultats de ces outils qu'on a définis après les traiter sous python pour reconstruire un modèle de reconnaissance des réseaux routiers.

Chapitre III

Extraction des Réseaux routiers à partir des cartes

Satellitaires.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

III.1. Introduction

Dans les chapitres précédents, on a donné les notions de bases, qui sont des connaissances nécessaires pour entamer le but de ce chapitre qui sert à réalisation du projet.

Notre objectif est basé sur l'utilisation des outils Google Drive, Google Colab qui nous permettons de gérer et traiter les données et les Dataset jusqu'aux résultats obtenus.

III.2. Les outils utilisés

Il est nécessaire de présenter les outils informatiques utilisés avant d'entamer la présentation des différentes parties qui constituent notre système.



III.2.1 Langage de programmation python

Python est un langage programmation open source le plus employé par les informaticiens.

Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels.

Malgré qu'il ne soit pas le plus puissant du point de vue calcul et rapidité d'exécution, mais dans le domaine AI est le plus utilisé car sa syntaxe est facile et pratique, il permet d'y progresser très certainement plus aisément que dans d'autres langages.

Enfin, concernant le domaine de (DL) Python se distingue tout particulièrement en offrant une pléthore de bibliothèques de très grande qualité, couvrant tous les types d'apprentissages disponibles sur le marché.



III.2.2 Jupiter Notebook

Jupyter Notebook est une application Web Open Source permettant de créer et de partager des documents contenant du code (écrit par langage de programmation python), des équations, des images et du texte. Avec cette application il est possible de faire du traitement de données de la visualisation de données, du Deep Learning.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

III.2.3 Les bibliothèques Python

a) NumPy

C'est l'une des bibliothèques les plus essentielles lorsqu'on travaille sur des projets en Python. Elle permet des manipulations de tableaux de grande dimension comme les opérations arithmétiques, les conversions, etc.

b) Matplotlib

On utilise cette bibliothèque pour une représentation visuelle des résultats. Elle peut faire plusieurs de choses avec les données, comme les fonctions de tracé, les graphiques à barres, etc. Nous l'utilisons quand nous devons tracer quelques images pour faire des comparaisons

c) OpenCV

C'est l'une des meilleures bibliothèques de vision par ordinateur et de traitement d'image utilisée pour faire des tonnes de choses sur des images et des vidéos. Elle met à disposition de nombreuses fonctionnalités très diversifiées pour le traitement.

d) Tensorflow

TensorFlow est une bibliothèque de Deep Learning développée par l'équipe Google Brain de Google, il s'agit d'une boîte à outils permettant de résoudre des problèmes mathématiques extrêmement complexes avec aisance. Elle permet de développer des architectures d'apprentissage expérimentales.

e) Keras

Keras est un API de réseaux neuronaux de haut niveau, écrite en Python et capable de s'exécuter sur TensorFlow, Il a été développé dans le but de permettre une expérimentation rapide.

Elle prend en charge les réseaux convolutionnels et les réseaux récurrents, ainsi que les combinaisons des deux.

Keras est compatible avec les versions de Python 2.7 / 3.6 et autres.

Keras a été initialement développé dans le cadre de l'effort de recherche du projet ONEIROS (système d'exploitation de robot intelligent neuro-électronique à composition open-ended).

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

III.2.4 Google Colab

Google Colab ou Colaboratory est un service Cloud, offert par Google (gratuit), basé sur Jupiter Notebook et destiné à l'entraînement et à la recherche dans l'apprentissage automatique. Cette plateforme permet d'importer un ensemble de données d'images, d'entraîner un classificateur d'images sur cet ensemble et d'évaluer le modèle de Deep Learning directement dans le Cloud (fig III.1).

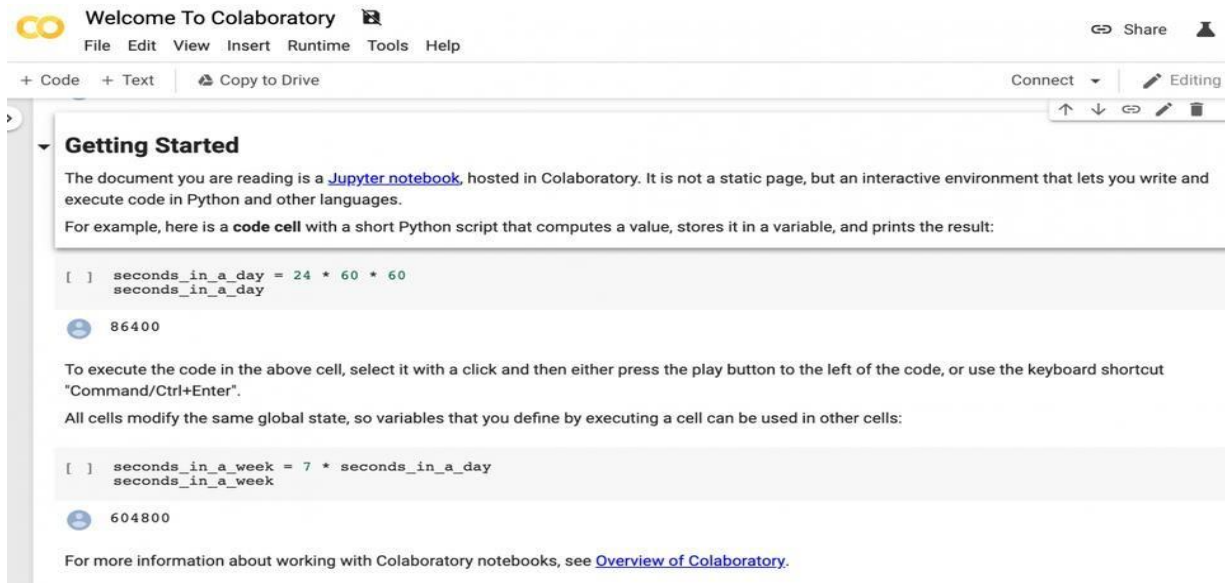


Figure III.1 : Interface de Google Colab.

On a utilisé Google Colab, car il présente les avantages suivants :

❖ Aucune configuration requise

On n'a pas besoin d'installer quoi que ce soit sur notre ordinateur à l'exception d'un navigateur (Google chrome ou Firefox).

❖ Accès gratuit aux TPU

Les Notebooks Colab exécutent les codes sur les serveurs Cloud de Google, nous avons donc à notre disposition toute la puissance du matériel Google.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

III.2.5 Google Drive

Google Drive est un service stockage et de partage de fichiers en ligne lancé par Google en avril 2012, il permet aux utilisateurs de stocker, partager, modifier et visualiser différents types de fichiers, et de les synchroniser à distance avec des terminaux fixes (PC, Mac) ou mobiles.

Google Drive est utilisé car :

- ❖ Il Permet de réserver un espace de stockage qui communique avec les autres services de Google (Google Docs, Google Colab).
- ❖ Il permet de Transfer des données entre Colab et Drive et de les synchroniser plus vite qu'avec une machine locale (PC, téléphone).

Il permet une sauvegarde rapide et automatique de l'état d'avancement dans le Drive.

III.3 Les démarches (étapes) du DL

La schématisation employée par la majorité des projets de vision utilisant le DL est donnée dans le synoptique ci-dessous :

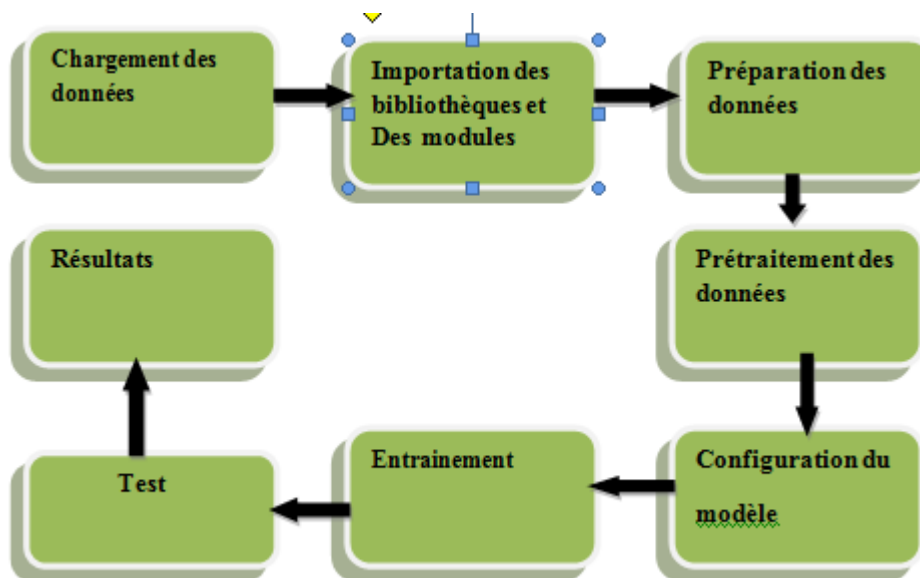


Figure III.2 : Les étapes de détection par DL.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

III.3.1 Chargement des données [21]

Le chargement des données se fait a partir du site « kaggle » (site gratuit)

Nous avons importé la dataset « roadextraction datasetfrom deep globe challenge » vers notre drive suivant trois étapes :

- Relier le compte Kaggle avec Google drive :

```
!pip install -q kaggle  
  
!mkdir ~/.kaggle/  
  
!cp '/content/drive/MyDrive/kaggle/kaggle.json' ~/.kaggle/  
  
!chmod 600 ~/.kaggle/kaggle.json
```

Figure III.3 relier kaggle avec google drive.

- Télécharger dataset vers Google drive :

```
! kaggle datasets download -d balraj98/deepglobe-road-extraction-dataset  
  
Downloading deepglobe-road-extraction-dataset.zip to /content  
100% 3.79G/3.79G [01:08<00:00, 25.2MB/s]  
100% 3.79G/3.79G [01:08<00:00, 59.1MB/s]
```

Figure III.4 importation dataset vers google drive.

- décompresser le fichier :

```
! unzip /content/deepglobe-road-extraction-dataset.zip -d deepglobe-road-extraction-dataset
```

Figure III.5 Décompression du fichier.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

III.3.2 Importation des bibliothèques et modules

Cette phase consiste à installer et importer des bibliothèques comme (Tensorflow,opencv...), elles contiennent des fonctions de (DL), pour notre cas toutes les fonctions préinstaller et incluse dans Google Colab.

```
import os
import sys
import numpy as np
import re
import cv2
import matplotlib.pyplot as plt
import tensorflow as tf
```

Figure III.6 Importation des bibliothèques.

III.3.3 Préparation des données

Un tri de paramètres des images données est effectué pour avoir deux classes différentes image(x), masque (y) (fig III.7).

```
framObjTrain = {'img' : [],
                'mask' : []
               }

def LoadData( frameObj = None, imgPath = None, maskPath = None, shape = 128):
    imgNames = os.listdir(imgPath)
    maskNames = []

    for mem in imgNames:
        mem = mem.split('_')[0]
        if mem not in maskNames:
            maskNames.append(mem)

    imgAddr = imgPath + '/'
    maskAddr = maskPath + '/'

    for i in range (1250):
        try:
            img = plt.imread(imgAddr + maskNames[i] + '_sat.jpg')
            mask = plt.imread(maskAddr + maskNames[i] + '_mask.png')
```

(figureIII.7) tri des paramètres des images.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

III.3.4 Prétraitement des données

Cette phase a pour but de régler des paramètres qui nous permet de redimensionner les images de notre dataset pour l'adapter avec notre input de modèle. (fig III.8)

```
for i in range (1250):
    try:
        img = plt.imread(imgAddr + maskNames[i] + '_sat.jpg')
        mask = plt.imread(maskAddr + maskNames[i] + '_mask.png')

    except:
        continue
    img = cv2.resize(img, (shape, shape))
    mask = cv2.resize(mask, (shape, shape))
    frameObj['img'].append(img)
    frameObj['mask'].append(mask[:, :, 0])

return frameObj
ramObjTrain = LoadData( framObjTrain, imgPath = '/content/drive/MyDrive/deepglobe-road-extraction-dataset/train',
                        maskPath = '/content/drive/MyDrive/deepglobe-road-extraction-dataset/train'
                        , shape = 128)
```

FigureIII.8 Redimensionnement des images.

III.3.5 Configuration de modèle

Pour la configuration du modèle, nous avons opter pour l'architecture U.net déjà présentée dans le chapitre 2 .

a) Chemin de contraction

Le chemin contractant est composé de quatre blocs où chaque bloc est composé de:

- Couche de convolution 3x3 + fonction d'activation (relu) [La suppression est facultative]
- Couche de convolution 3x3 + fonction d'activation (relu) [La suppression est facultative]
- Couche de mise en commun 2x2 Max.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

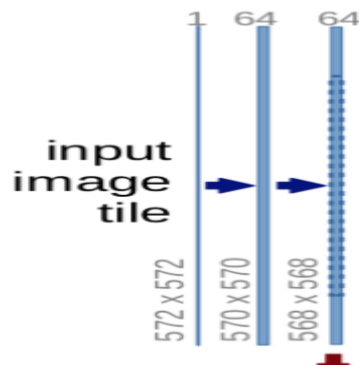


Figure III.9 chemin de contrat.

b) Chemin d'extension

Le chemin en expansion est également composé de 4 blocs. Chacun de ces blocs est composé de:

- Couche De convolution ou UpSampling2D avec stride 2
- L'image est concaténée avec l'image correspondante du chemin contractant
- Couche de convolution 3x3 + fonction d'activation [La suppression est facultative]
- Couche de convolution 3x3 + fonction d'activation [La suppression est facultative]

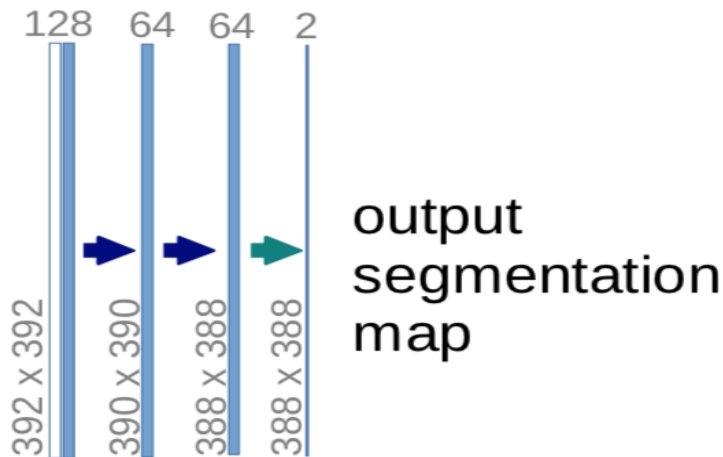


Figure III.10 chemin d'extension.

Le modèle se compose de 76 couches :

```
len(unet.layers)
```

76

Figure III.11 les couches de notre modèle.

On a obtenu structure u.net dans l'annexe 1.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

III.3.6 Entraînement

le modèle va être entraîné pour qu'il puisse segmenter les réseaux routiers.

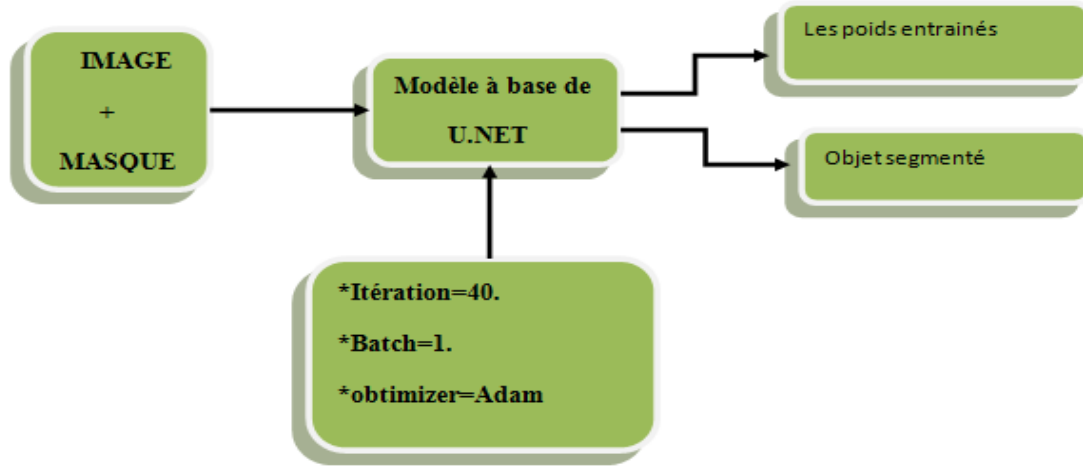


Figure III.12 Phase d'entraînement.

le paramètre [batch] : représente La taille du lot est le nombre d'images que nous traitons avant de mettre à jour les poids du réseau.

le paramètre [Le nombre d'époques] : représente un hyper paramètre qui définit le nombre de fois que l'algorithme d'apprentissage fonctionnera sur l'ensemble des données de l'entraînement.

l'optimizer « Adam » :est une méthode de descente de gradient stochastique basée sur l'estimation adaptative des moments du premier et du second ordre.

Le nombre d'époques est traditionnellement important, souvent des centaines ou des milliers, ce qui permet à l'algorithme d'apprentissage de fonctionner jusqu'à ce que l'erreur du modèle ait été suffisamment minimisée.

Pour notre cas, les paramètres sont:

*batch=1 ; epoch=40;

Avant de lancer l'entraînement il faut d'abord modifier le type d'exécution et le mettre sur le TPU pour accélérer l'entraînement.(figIII.13)

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

Paramètres du notebook

Accélérateur matériel

TPU ▼ ?

Pour tirer le meilleur parti de Colab, évitez d'utiliser un TPU si vous n'en avez pas besoin. [En savoir plus](#)

Omettre l'élément de sortie des cellules de code lors de l'enregistrement de ce notebook

Annuler Enregistrer

Figure III.13 Les étapes d'activation TPU.

Une fois l'entraînement lancé, nous obtenus les 10 premières itérations.(figIII.14).

```
Epoch 1/40
230/230 [=====] - 225s 891ms/step - loss: 0.3534 - accuracy: 0.9015
Epoch 2/40
230/230 [=====] - 206s 898ms/step - loss: 0.1854 - accuracy: 0.9546
Epoch 3/40
230/230 [=====] - 204s 886ms/step - loss: 0.1629 - accuracy: 0.9546
Epoch 4/40
230/230 [=====] - 204s 888ms/step - loss: 0.1524 - accuracy: 0.9546
Epoch 5/40
230/230 [=====] - 204s 887ms/step - loss: 0.1421 - accuracy: 0.9546
Epoch 6/40
230/230 [=====] - 205s 890ms/step - loss: 0.1328 - accuracy: 0.9547
Epoch 7/40
230/230 [=====] - 204s 889ms/step - loss: 0.1253 - accuracy: 0.9561
Epoch 8/40
230/230 [=====] - 204s 889ms/step - loss: 0.1197 - accuracy: 0.9579
Epoch 9/40
230/230 [=====] - 204s 888ms/step - loss: 0.1144 - accuracy: 0.9589
Epoch 10/40
230/230 [=====] - 204s 889ms/step - loss: 0.1093 - accuracy: 0.9597
```

Figure III.14 Début de l'entraînement.

la fonction de perte passe de 0.3534 à 0.1093, et la fonction de précision passe de 0.9015 à 0.9597.

pour rester dans environnement Colab, on exécute une procédure (annexe 2).

Les dernières itérations nous permettent d'avoir une fonction de perte à 0.0331,et une fonction de précision à 0.9815 (figIII.15).

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

```
230/230 [=====] - 205s 891ms/step - loss: 0.0531 - accuracy: 0.9745
Epoch 28/40
230/230 [=====] - 205s 890ms/step - loss: 0.0527 - accuracy: 0.9747
Epoch 29/40
230/230 [=====] - 205s 890ms/step - loss: 0.0493 - accuracy: 0.9758
Epoch 30/40
230/230 [=====] - 205s 890ms/step - loss: 0.0470 - accuracy: 0.9765
Epoch 31/40
230/230 [=====] - 205s 889ms/step - loss: 0.0449 - accuracy: 0.9772
Epoch 32/40
230/230 [=====] - 205s 892ms/step - loss: 0.0436 - accuracy: 0.9777
Epoch 33/40
230/230 [=====] - 204s 888ms/step - loss: 0.0426 - accuracy: 0.9781
Epoch 34/40
230/230 [=====] - 205s 891ms/step - loss: 0.0417 - accuracy: 0.9784
Epoch 35/40
230/230 [=====] - 200s 869ms/step - loss: 0.0386 - accuracy: 0.9795
Epoch 36/40
230/230 [=====] - 200s 870ms/step - loss: 0.0370 - accuracy: 0.9801
Epoch 37/40
230/230 [=====] - 201s 872ms/step - loss: 0.0372 - accuracy: 0.9800
Epoch 38/40
230/230 [=====] - 201s 875ms/step - loss: 0.0372 - accuracy: 0.9800
Epoch 39/40
230/230 [=====] - 201s 874ms/step - loss: 0.0346 - accuracy: 0.9809
Epoch 40/40
230/230 [=====] - 201s 874ms/step - loss: 0.0331 - accuracy: 0.9815
```

Figure III.15 la fin de l'entraînement.

Ce résultat est obtenu après itération 40.

On a obtenu un modèle précis étant donné que la perte obtenue est inférieure 4 % et une précision de 98,15 %.

Le graphe de la performance de notre modèle tout au long du processus de l'entraînement est donné dans la (fig III.16).

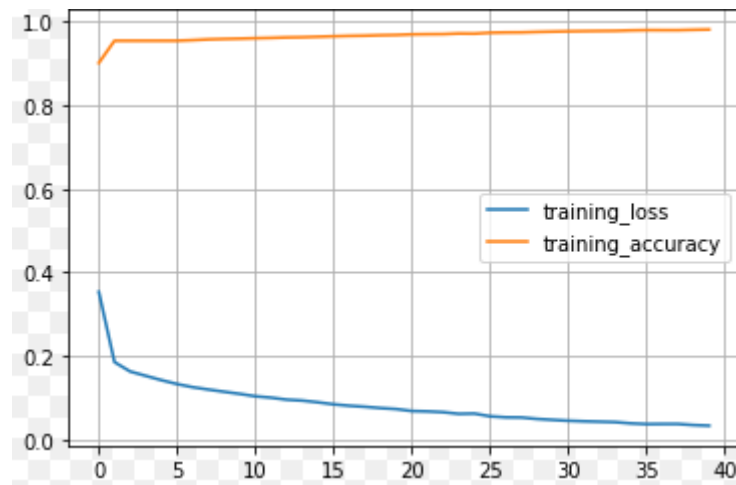


Figure III.16 Graphique de la perte/précision en fonction du nombre d'itération.

Interprétation :

D'après la figure, on remarque que a partir d'un certains nombre d'itération l'erreur diminue et la précision augmente.

Après l'entraînement notre modèle a eu plus de deux millions de paramètres :

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

Total params: 2,164,593
Trainable params: 2,161,649
Non-trainable params: 2,944

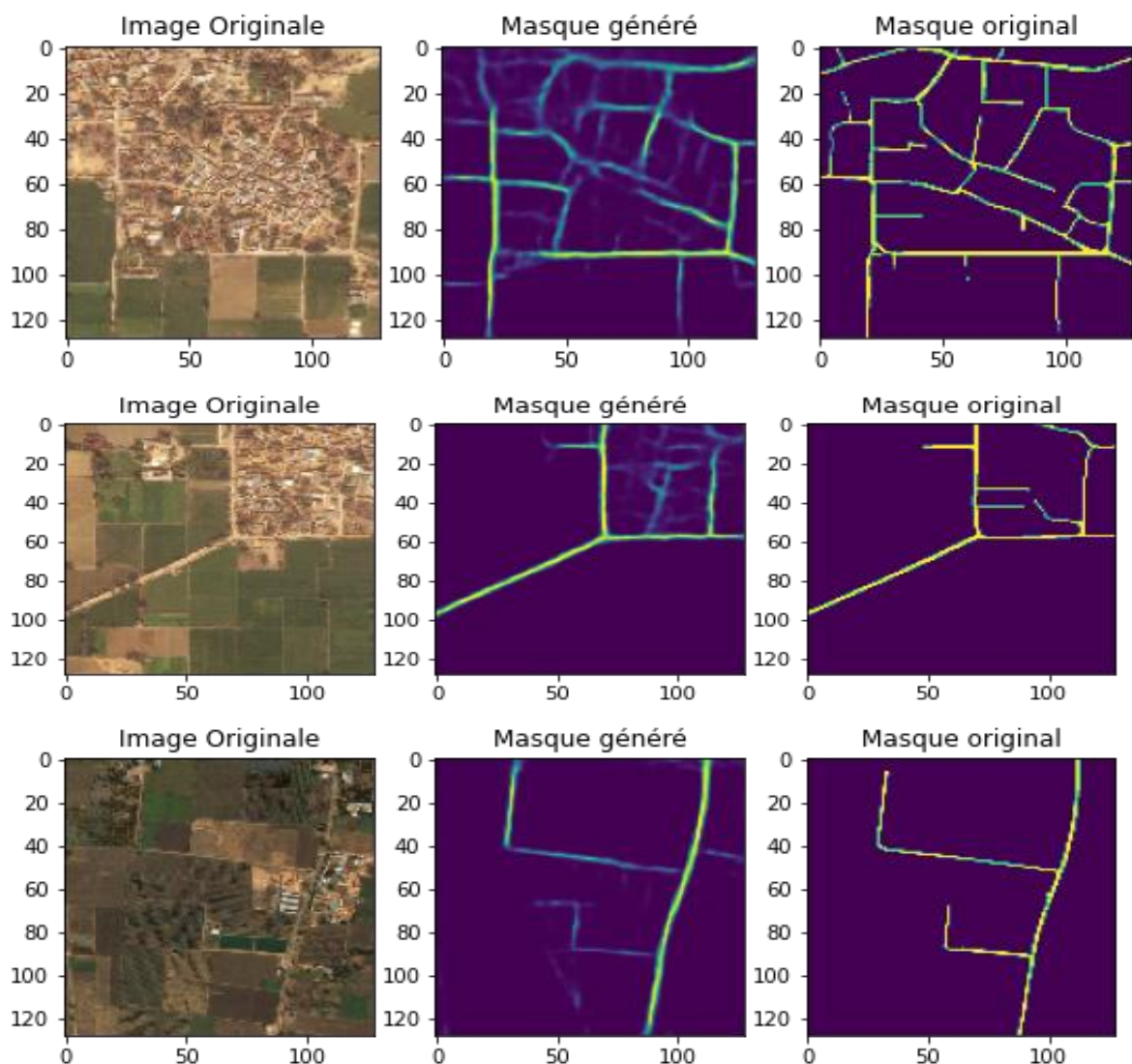
Figure III.17 le nombre de paramètre.

III.3.7 Test

Après l'entraînement, on teste notre modèle avec des images ayant déjà un masque, puis on effectue la comparaison de ces derniers avec les masques générés par notre modèle.

Durant l'étape du test on évalue les performances du modèle.

Nous avons utilisé trois images avec leur masque (figIII.18)



Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

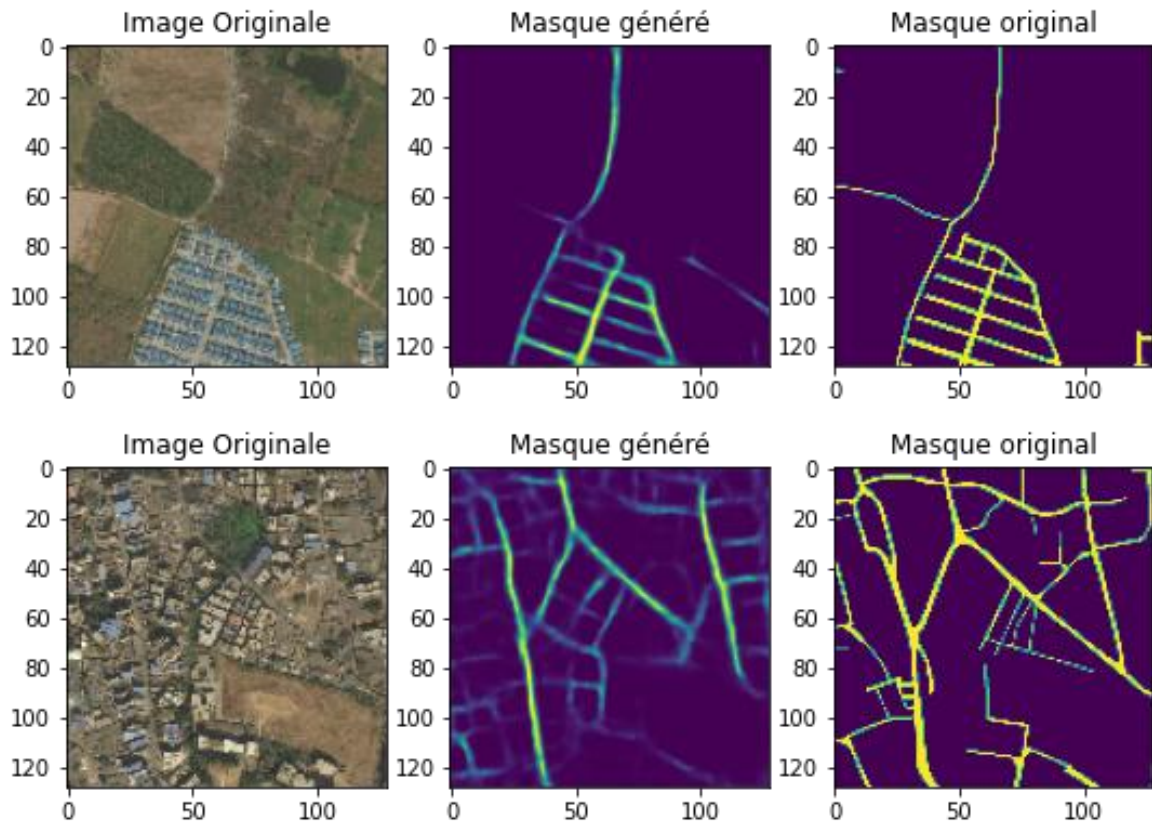


Figure III.18 Les images testées (à gauche :images originales, à droite : masque originaux, au milieu : images générées) .

À partir de ces résultats, on peut dire que les masques générés présentent un taux de ressemblance très important aux masques originaux.

III.3.8 Prédiction

On lui donne des images contenant des réseaux routiers, il va les générer une image qui contient des réseaux routiers.(fig III.19)

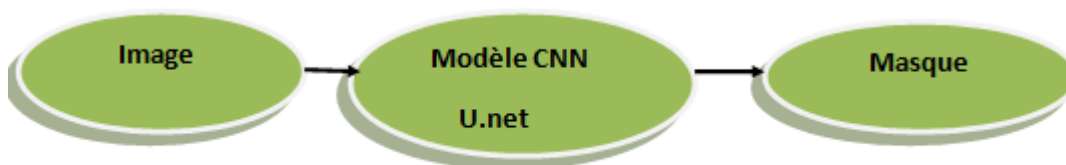
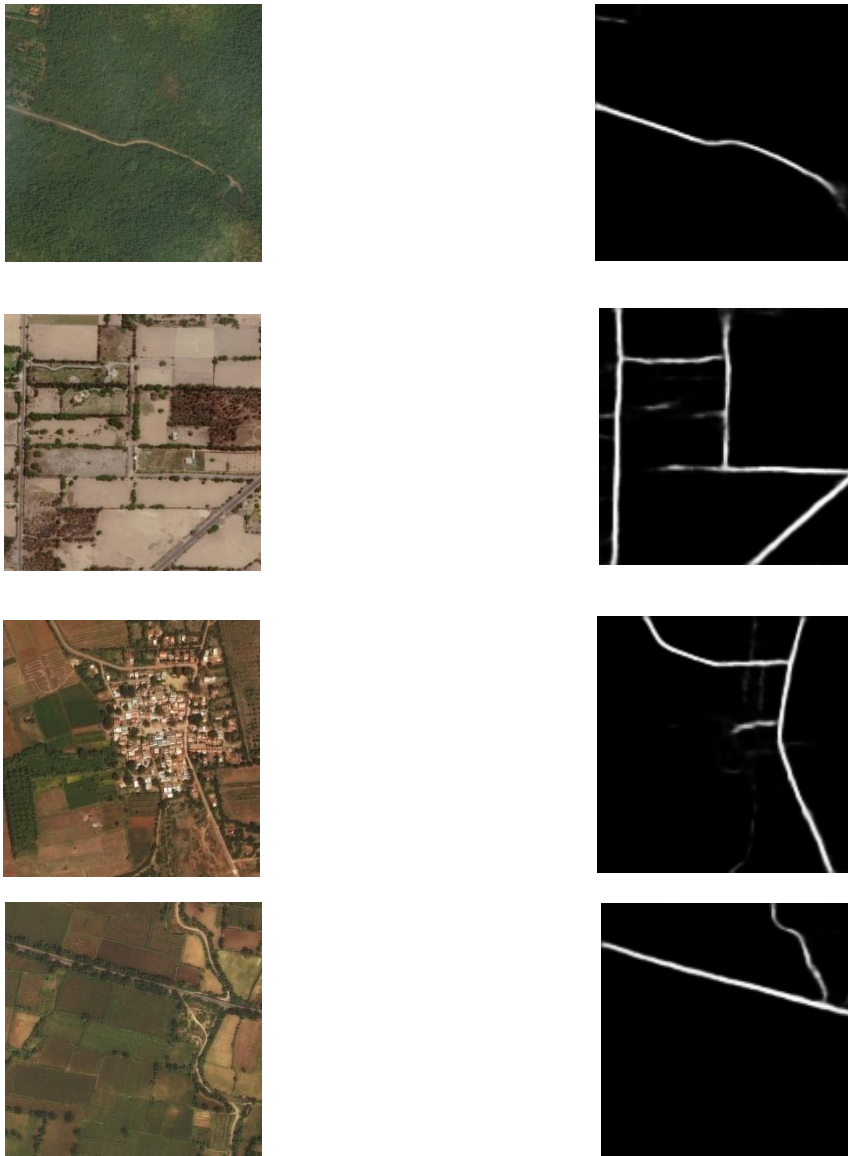


Figure III.19 Le schéma synoptique de la prédiction.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

Les résultats obtenus montrent l'efficacité du modèle (figIII.20.)



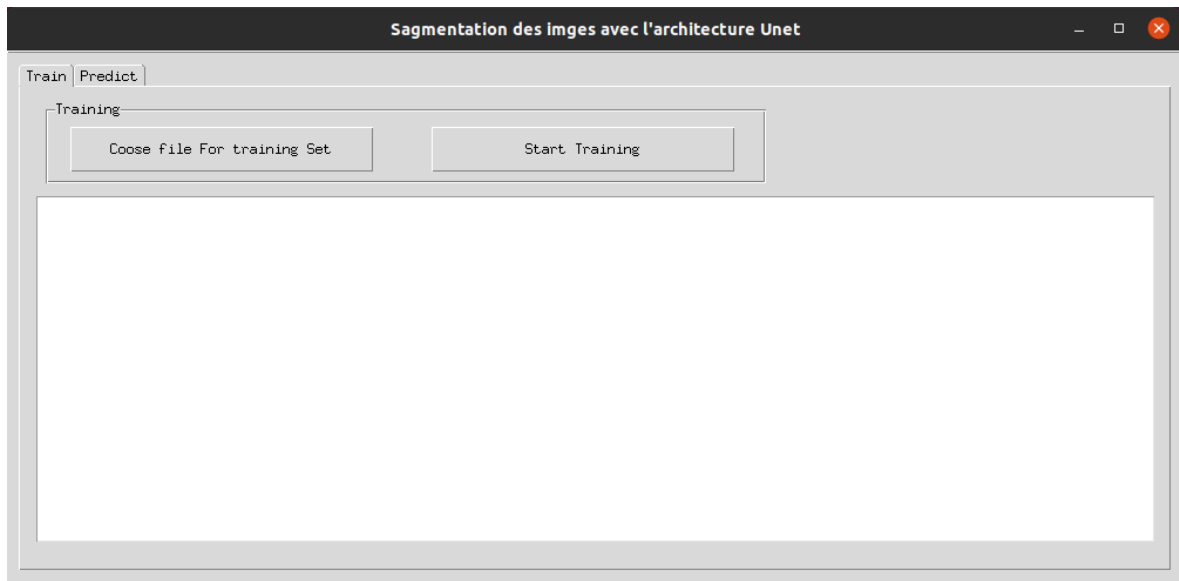
FigureIII.20 Les résultats obtenus.

III.4 interface graphique

Nous avons finalisé ce travail dans une interface graphique, on la programmer avec python mais on a utilisé la bibliothèque « Tkinter » qu'est dédié pour construire les interfaces graphiques (figIII.21).

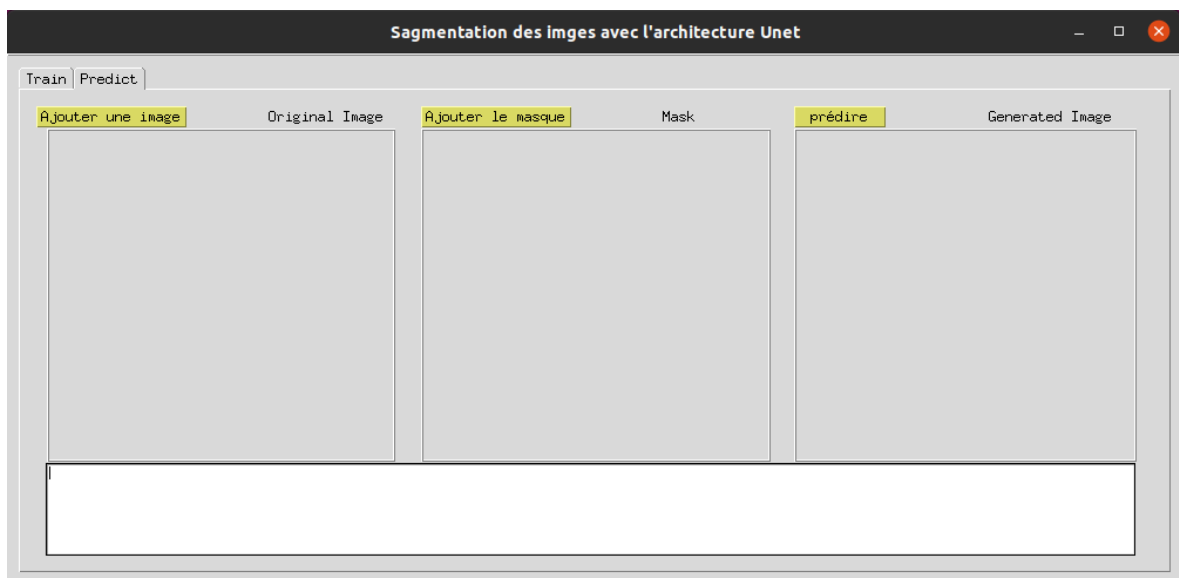
Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

Premièrement on lance l'interface :



FigureIII.21a lancement de l'interface.

Deuxièmement faire un click sur le bouton « prédire » à gauche :



FigureIII.21b exécution du modèle.

Troisièmement nous avons choisie une image pour la segmenter et nous avons fait un click sur le bouton jaune à droit « prédire » :

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.



FigureIII.21c prédiction.

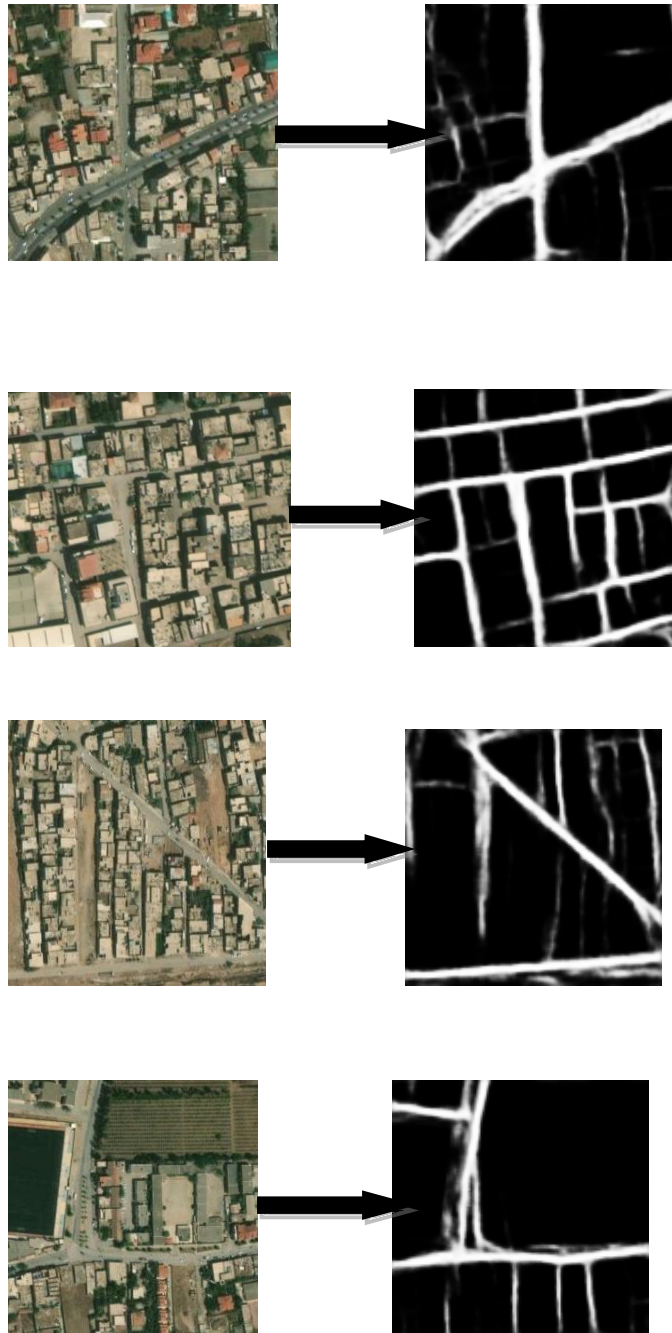
III.5 Application du modèle sur une carte satellitaire du centre de Blida

Notre modèle a été entraîné et personnalisé pour qu'il puisse segmenter n'importe quel réseau routier, dans ce qui suit nous allons donner les résultats de l'application du modèle suivant un critère d'étude, et basé sur la performance de détection du modèle.

Nous avons travaillé sur la carte de centre de ville de Blida, et nous avons fait l'étude sur types de carte satellitaire.

On applique notre modèle sur les cartes ci-dessous du centre de Blida nous avons pris les images satellitaires de Blida de « zoom.earth ».

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.



FigureIII.22 les cartes de Blida après la segmentation.

COMMENTAIRE :

Les résultats obtenus s'avèrent très satisfaisants, vue que nous avons fait l'extraction des réseaux routier a partir de cartes satellitaires du centre ville de Blida avec succès.

Chapitre III : Extraction des Réseaux routiers à partir des cartes Satellitaires.

III.6 Conclusion

Dans ce chapitre, nous avons suivi la schématisation générale employée par la majorité des projets de vision utilisant le DL, nous avons par le biais de cette dernière entrainer notre modèle U.net et procurer à ce dernier des données en entrée afin qu'il puisse nous segmenter des images pour extraire le réseau routier.

Les résultats obtenus sont très satisfaisants pour des itérations à partir de 40.

L'avantage que nous procure U.net est qu'il puisse faire des extractions d'objets à partir des cartes satellitaires.

Conclusion générale

Conclusion générale

Notre projet, a été proposé dans le but de participer à l'élaboration d'une partie d'un système d'aide à la reconnaissance des parcours structurants dans les centres urbains anciens selon le modèle interprétatif de G.Caniggia issu de la méthode Muratorienne. En utilisant le Deep Learning et en développant un modèle d'extraction d'objets d'intérêt dans une scène graphique, dans notre cas nous devions préparer une reconnaissance des parcours structurants en se basant sur une méthode d'extraction des réseaux routiers par la segmentation des images utilisant le modèle U.net.

Durant ce travail, on a essayé de suivre le cahier des charges qui nous a été confié, à savoir, la tâche demandée est d'extraire les réseaux routiers à partir de carte satellitaire, particulièrement les réseaux routiers du Blida, par la méthode (DL).

Les résultats obtenus, nous permettent d'afficher une grande satisfaction car le modèle employé a pu extraire les réseaux voirie pour les cartes satellitaires, ce qui nous permet de dire que les objectifs visés par le cahier des charges ont été correctement atteints.

Nous avons rencontré quelques problèmes dans le début de notre étude, car les bibliothèques voulus, affichaient des erreurs, après l'installation (Il n'y a pas une compatibilité des versions entre elles), mais on a pu contourner cet obstacle par l'utilisation Google Colab qui utilise des bibliothèques préinstallées. En utilisant le TPU qu'il fournit, la durée d'entraînement a été écourtée. (Quelques heures au lieu des jours par CPU).

Enfin ce projet nous a permis de développer nos connaissances en touchants à plusieurs domaines tels que la typomorphologie des villes, le Deep Learning, surtout le langage de programmation python, en espérant qu'il va nous permettre d'ouvrir des opportunités dans le future.

Bibliographie

- [1] Jean cartex, « lecteur d'une ville : Versailles ».
- [2] Traduction de l'ouvrage de canggia, « lecteur de bâti de base ».
- [3] Andi, « wilaya de Blida », 2016.
- [4] Ouadah Sofiane, 'requalification du quartier ramoul a Blida et conception d'une gare multimodale', mémoire de master, université de Blida département d'architecture, 2014.
- [5] Benchabane leila, Zouggari Zakaria et Benkara omar, 'recomposition d'un ilot du 19eme siècle dans l'hyper-centre d'Alger, conception d'un immeuble d'habitat et d'une résidence pour chercheurs', université de Blida département d'architecture, 2018.
- [6] T. Beysolow II, "Introduction to Deep Learning Using R", 2017.
- [7] Guillaume Saint-Cirgue, 'Apprendre le machine learning en une semaine', 2019.
- [8] Kifadji Amir, « segmentation des images de drone par deep learning, Mémoire de master, université de Badji Mokhtar Annaba département d'électronique, 2020.
- [9] <http://www-lisic.univ-littoral.fr/verel/TEACHING/08-09/sac-M1/cRdnv9.pdf>
- [10] http://conf.laas.fr/ignotus/archives/Doncescu_reseaux_neuronnes.pdf
- [11] <https://openclassrooms.com/fr/courses/4011851-initiez-vous-au-machine-learning/4020611-identifiez-les-differents-types-dapprentissage-automatiques>
- [12] Claude Touzet, ' les réseaux de neurones artificiels, introduction au connexionnisme, Editions la Machotte, 2016.
- [13] <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks- the-eli5-way-3bd2b1164a53>.
- [14] yacine ouassar : réseaux d'ondelettes et réseaux de neurones pour la modélisation statique et dynamique de processus, université pierre et marie curie paris, le 23 avril 2004
- [15] https://software.intel.com/content/www/us/en/develop/training/course-deep-learning.html#linklist_1231887423?multiplayer=5680465479001
- [16] Benjamin Graham, fractional Max-Pooling, 12 Mai 2015.
- [17] Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick - Mask R-CNN - modifié le 24 janvier 2015.
- [18] Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick - Mask R-CNN modifié

Bibliographie

le 24 janvier 2015.

[19] <https://datascientest.com/U-net>.

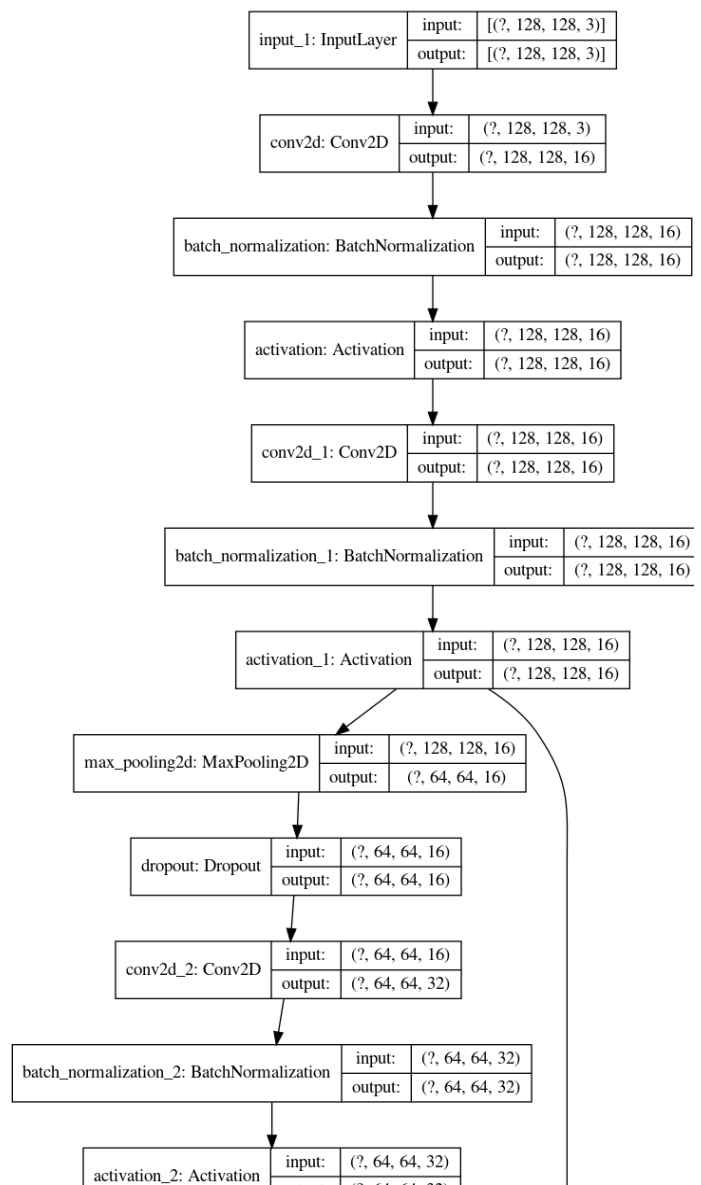
[20] <https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47>

[21] [https://www.kaggle.com/?fbclid=IwAR0iMsy-](https://www.kaggle.com/?fbclid=IwAR0iMsy-EIaXiP2cRb415T6IBDXWkje4eOp6xT1W0wmv7KDvV4ZEm19UC88)

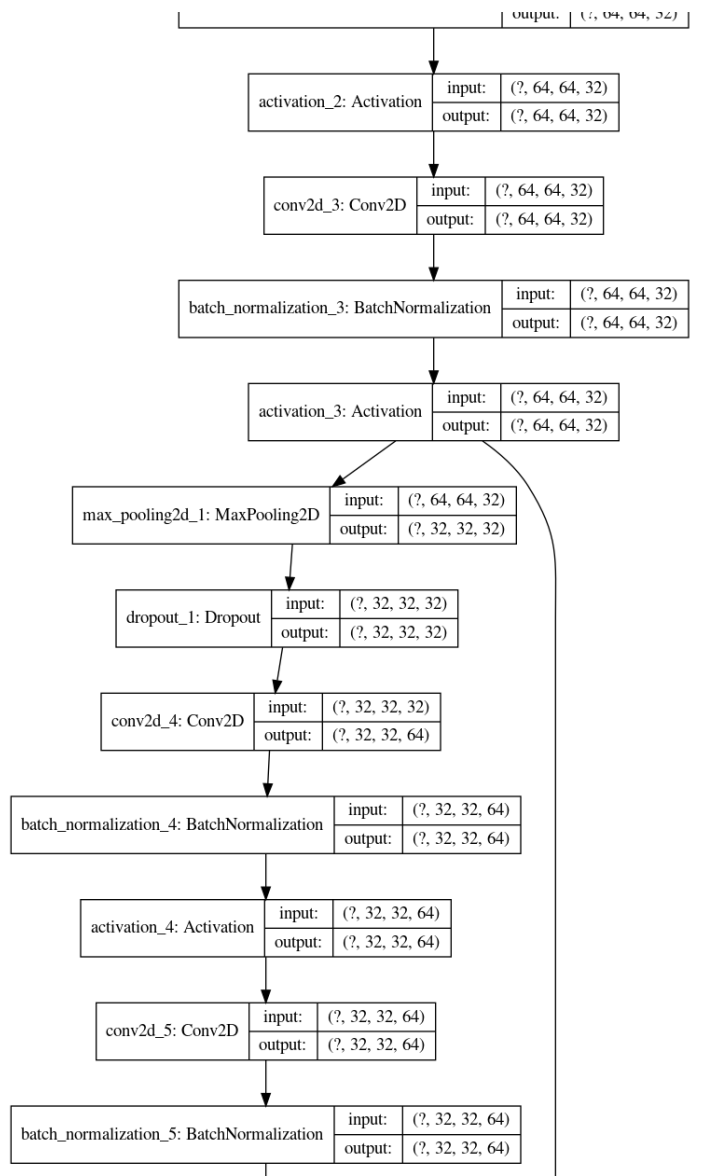
[ElaXiP2cRb415T6IBDXWkje4eOp6xT1W0wmv7KDvV4ZEm19UC88](https://www.kaggle.com/?fbclid=IwAR0iMsy-EIaXiP2cRb415T6IBDXWkje4eOp6xT1W0wmv7KDvV4ZEm19UC88)

Annexe

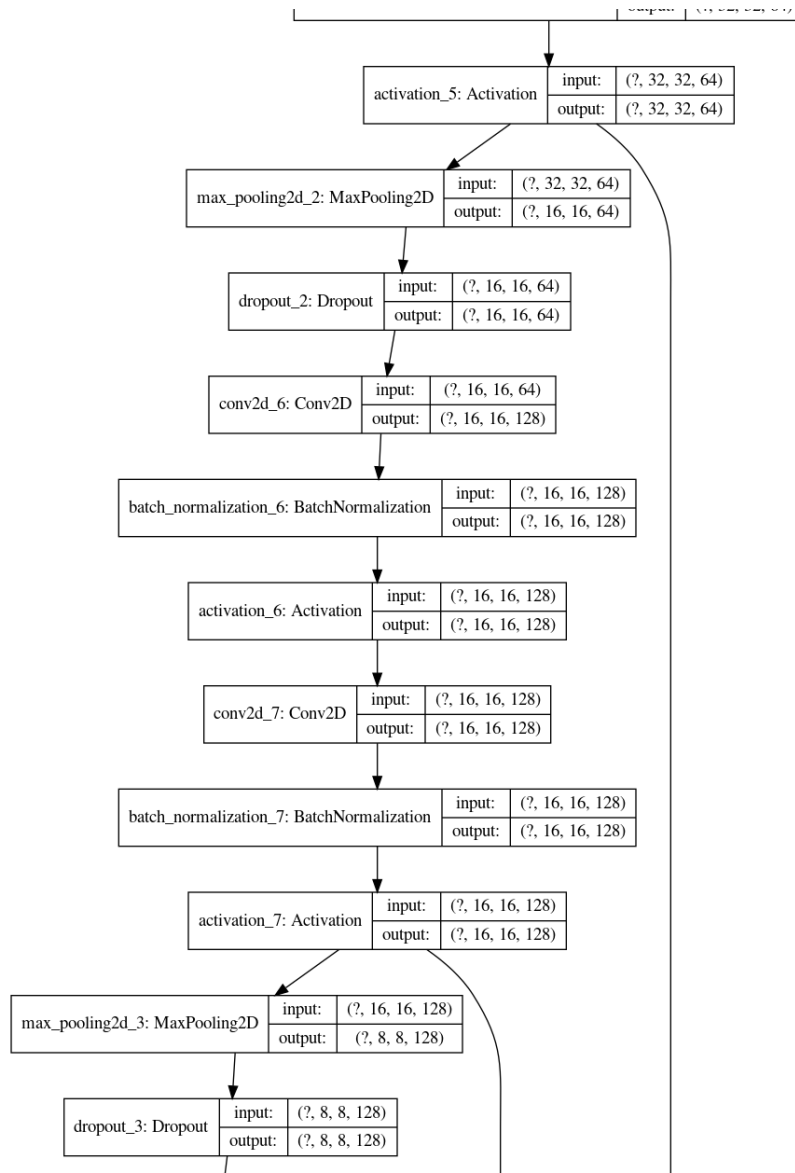
ANNEXE1:



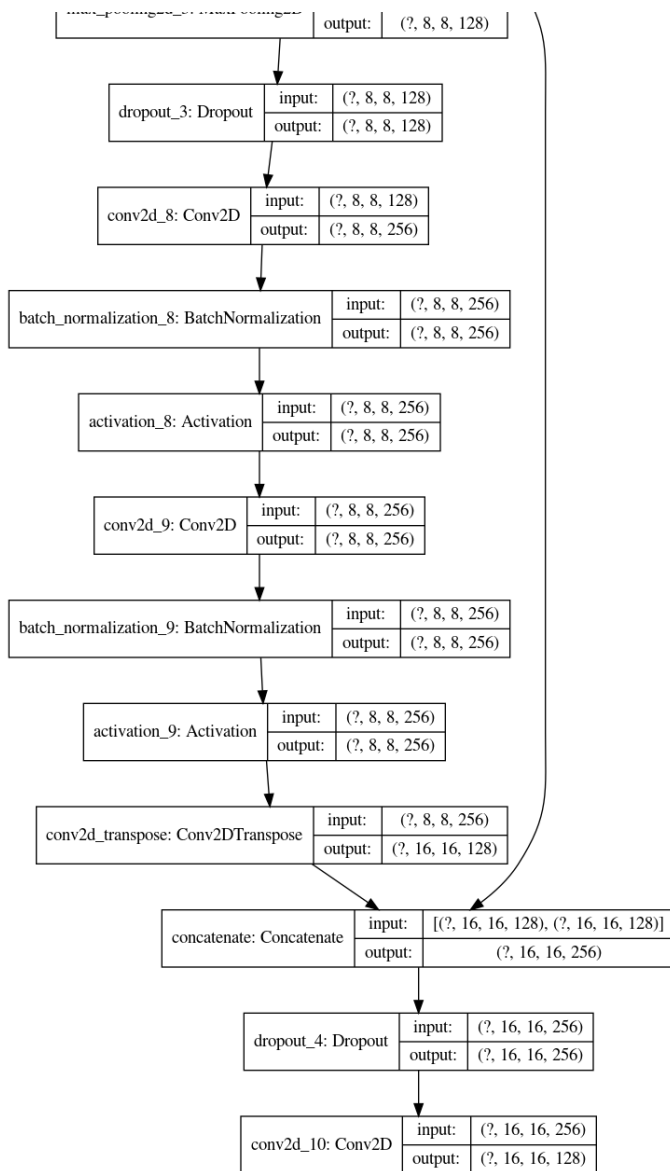
Annexe



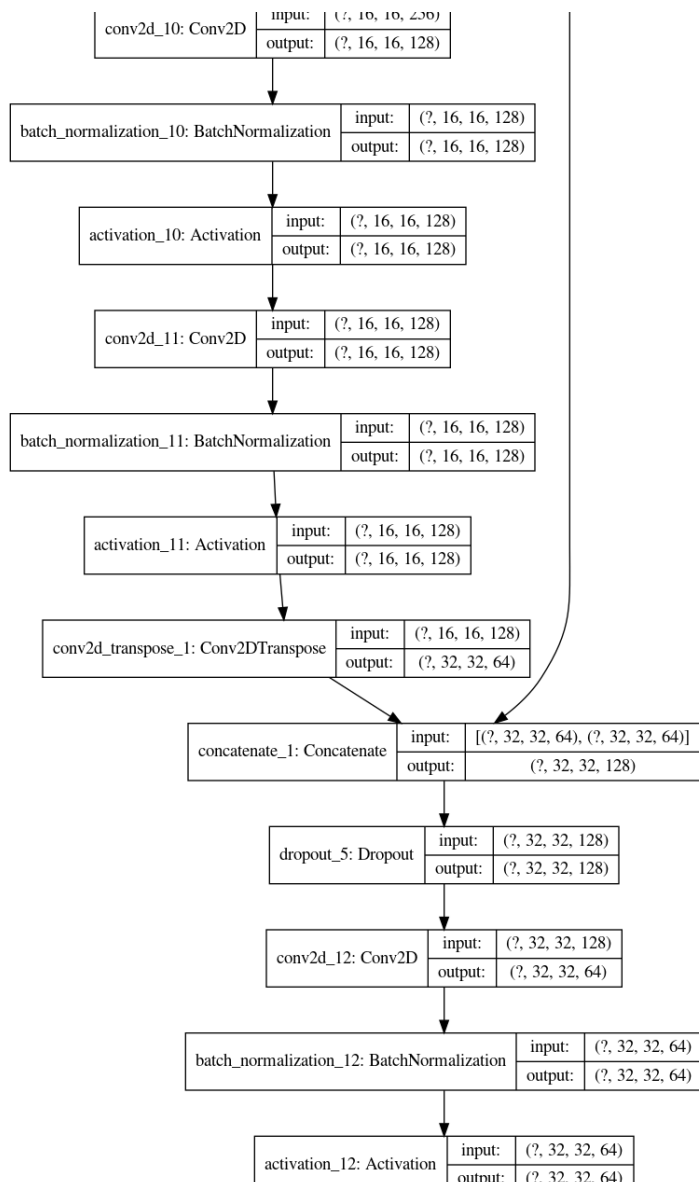
Annexe



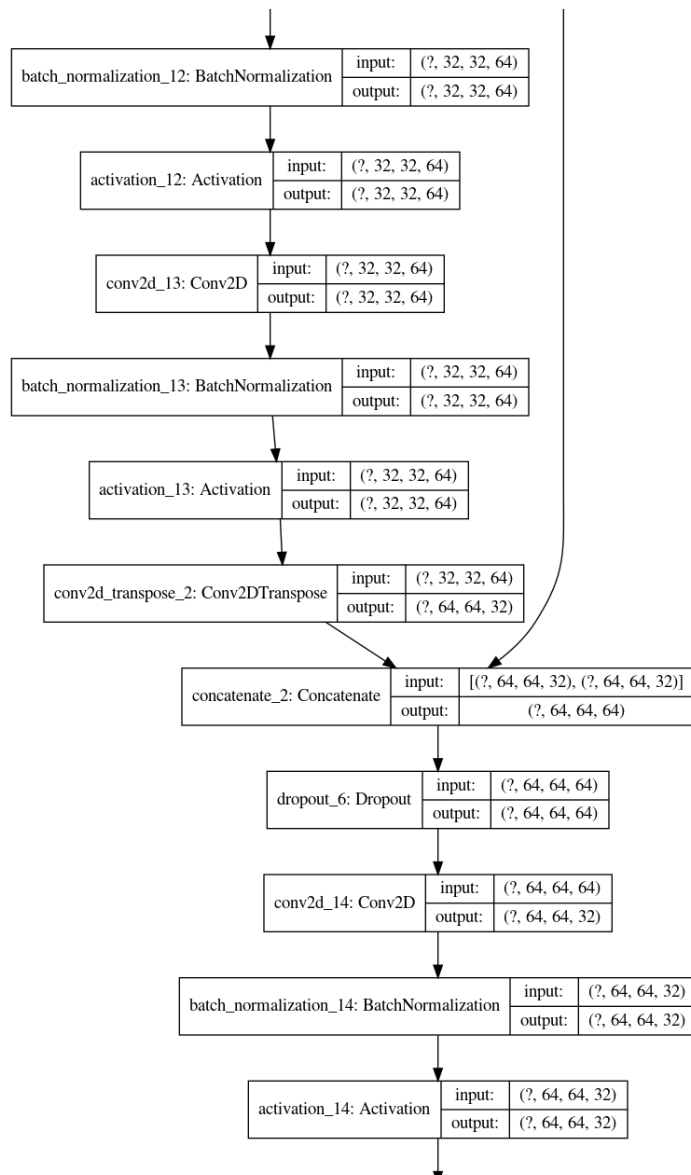
Annexe



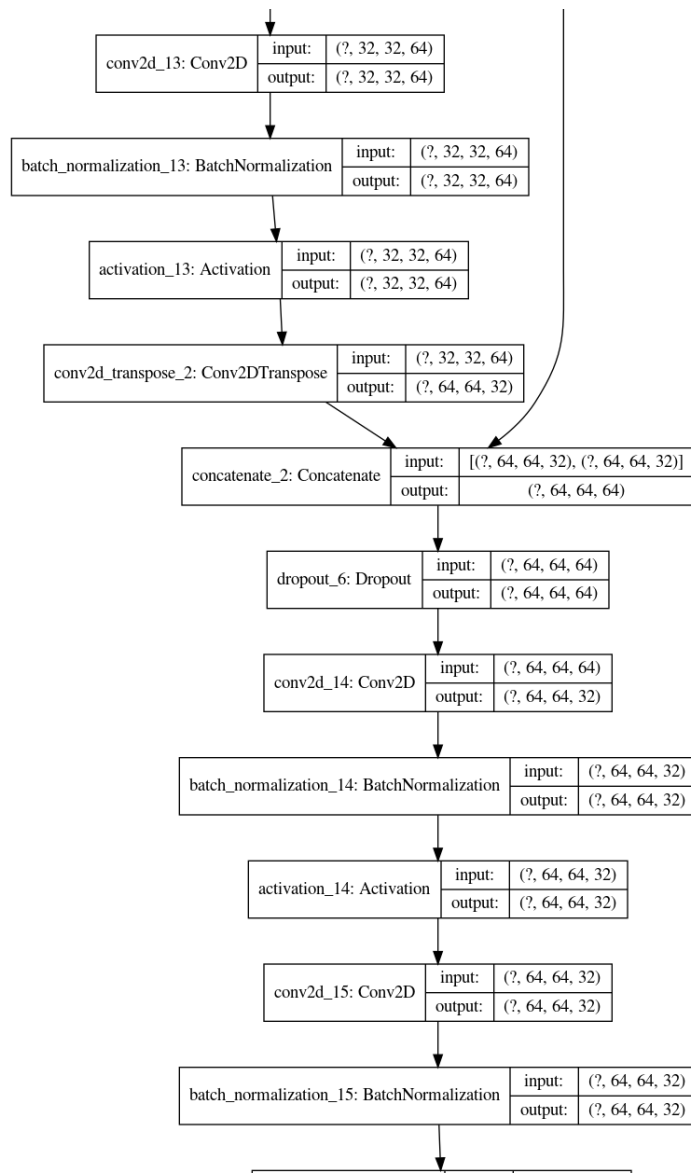
Annexe



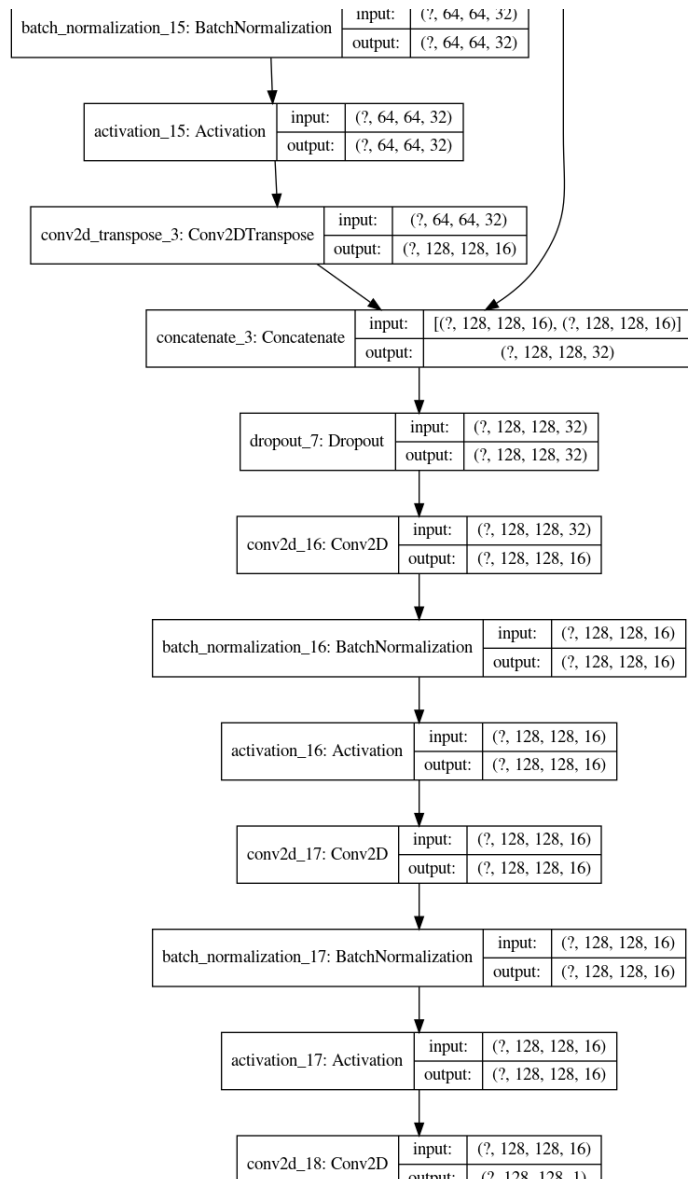
Annexe



Annexe



Annexe



Annexe

ANNEXE 2 :

Colab Cloud Service nous renvoi dans ses machines virtuelles si nous restons trop longtemps sans rien faire (30 à 90 minutes).

Pour éviter cette attente (CTRL + MAJ + i), nous devons en même temps ouvrir la vue de l'inspecteur sur notre navigateur.

Collons le code suivant dans la fenêtre de notre console et appuyez sur Entrée.

```
var clickConnect = function clickConnect(){
  console.log("Connect Clicked - Start");
  document.querySelector("#top-toolbar > colab-connect-button").shadowRoot.querySelector("#connect").click();
  console.log("Connect Clicked - End");
};

var intervalId = setInterval(clickConnect, 60000);

var stopClickConnectHandler = function stopClickConnect() {
  console.log("Connect Clicked Stopped - Start");
  clearInterval(intervalId);
  console.log("Connect Clicked Stopped - End");
};

return stopClickConnectHandler;
};

var stopClickConnect = startClickConnect();
```