

Filière

Génie Industriel : Data Science et Intelligence Artificielle (GI-IA&DS)

PROJET D'INITIATION

Sujet : Détection et Classification des conditions climatiques

Réaliseurs :

SAOUADOGO Salifou

OUEDRAOGO Ousséni

Encadrant :

Mr.FASSI FIHIRI Abdelkader

Mr. HOSNI Mohamed

Deuxième année du cycle d'Ingénieur

ENSAM MEKNES 2021-2022

REMERCIEMENTS

Avant tout développement de cette expérience, il apparaît opportun pour nous de commencer ce rapport de projet d'initiation par nos profonds remerciements à **Mr. MASROUR Tawfik**, le responsable de la filière **Génie Industriel option Data Science et Intelligence Artificielle**

Ensuite, nous tenons à remercier surtout Mr. **FASSI FIHRI Abdelkader** et Mr. **HOSNI Mohamed** pour les remarques, les informations partagés avec nous et le temps qu'ils ont consacré à répondre nos attentes pour notre meilleur encadrement.

Enfin, nous remercions également nos chers professeurs de la Filière **Génie Industriel option Data Science et Intelligence Artificielle** pour leurs guides à la réussite de ce projet.

Résumé et motivations

Le but de ce travail est d'utiliser des systèmes de détection (cameras, capteurs, radar...) des conditions climatiques puis utiliser les algorithmes de Deep Learning pour classifier ces classes climatiques afin de faciliter la conduite en cas des conditions dégradées.

La classification des images consiste à répartir systématiquement des images selon des classes établies au préalable, classer une image lui fait correspondre une classe, marquant ainsi sa parenté avec d'autres images. En général reconnaître une image est une tâche aisée pour un humain au fil de son existence, il a acquis des connaissances qui lui permettent de s'adapter aux variations qui résultent de conditions différents d'acquisition. Il est par exemple relativement simple de reconnaître un objet dans plusieurs orientations partiellement caché par un autre de près ou de loin et selon diverses illuminations. Toutefois les progrès technologiques en terme d'acquisition d'images (microscopes, caméras, capteurs) et de stockage engendrent des bases de données riche en information et multiplient les domaines d'applications, il devient alors difficile pour l'humain d'analyser le nombre important d'images, le temps requis le caractère répétitif de la tâche et la concentration nécessaire sont problématiques. Toutefois celle-ci n'est pas forcément aisée pour un programme informatique pour lequel une image est un ensemble de valeur numérique. L'objectif de la classification d'images est d'élaborer un système capable d'affecter une classe automatiquement à une image. Ainsi, ce système permet d'effectuer une tâche d'expertise qui peut s'avérer coûteuse à acquérir pour un être humain en raison notamment de contraintes physiques.

En effet pour élaborer un tel système, nous avons utilisé les algorithmes de Deep Learning avec les librairies et logiciels comme Tensorflow et Keras pour aboutir à la mise en place de ce système. Les réseaux de neurones convolutionnel figurent à la tête des meilleurs algorithmes utilisé par la communauté d'où la nécessité pour nous d'en utiliser.

Table des matières

CHAPITRE 1 : ÉTUDE BIBLIOGRAPHIQUE ET CAHIER DE CHARGE	7
I. PROBLEMATIQUE	7
II. ADVANCED DRIVER ASSISTANCE SYSTEM (ADAS)	7
III. LES RESEAUX DE NEURONES CONVOLUTIFS CNN	8
IV. CAHIER DE CHARGE	9
2) <i>L'Expression du Besoin /objectifs du projet.....</i>	9
3) <i>Description fonctionnelle des besoins.....</i>	10
4) <i>Principe de fonctionnement et arbre de Processus du Système étudié.....</i>	11
5) <i>Périmètre du projet</i>	12
6) <i>Délai de réalisation du projet.....</i>	12
7) <i>Cahier de charge fonctionnelle</i>	12
V. CONCLUSION CHAPITRE 1	14
CHAPITRE 2 : ÉTUDE DES SYSTEMES DE DETECTION	15
I. INTRODUCTION.....	15
II. CLASSIFICATIONS DES CONDITIONS CLIMATIQUES	15
III. LES DIFFERENTS TYPES DE CAPTEURS	15
IV. ÉTUDE COMPARATIVE DES SYSTEMES DE DETECTIONS SELON CHAQUE CLASSE DE CLIMAT	16
V. CONCLUSION CHAPITRE 2.....	22
CHAPITRE 3 : IMPLEMENTATION, PROGRAMMATION/CLASSIFICATION DES IMAGES.....	23
I. INTRODUCTION	23
II. CLASSIFICATION DES IMAGES	23
1) <i>Notion de base d'une image</i>	23
2) <i>Quelques Caractéristiques d'une image</i>	23
3) <i>Méthode de classification d'image.....</i>	24
4) <i>Indicateur de performance de la classification</i>	24
III. LE DEEP LEARNING	25
1) <i>Algorithmes de Deep Learning.....</i>	25
2) <i>Les réseaux de neurone convolutionnel CNN</i>	26
IV. IMPLEMENTATION/PROGRAMMATION.....	27
1) <i>Logiciels et Librairies utilisés.....</i>	27
2) <i>Configuration du Modèle (Séquentiel).....</i>	28
3) <i>Résultats obtenus après classification</i>	32
V. CONCLUSION GENERALE.....	37

Figures

Figure 1 : Système ADAS.....	7
Figure 2 : Niveau du Système ADAS.....	8
Figure 3 : Perception Multicouches.....	9
Figure 4 : Exemple de détection.....	10
Figure 5 : Processus du Système de détection.....	11
Figure 6 : Bête à corne.....	12
Figure 7 : Diagramme de Pieuvre.....	13
Figure 8 : Configuration du modèle.....	29
Figure 9 : Projection sur le dataset.....	30
Figure 10 : Graphique des résultats prédits.....	33
Figure 11 : Matrice de Confusion.....	34
Figure 12 : Graphique à secteur de la précision et erreurs.....	35
Figure 13 : précision par classe climatique.....	35

Introduction générale

Les conditions météorologiques dégradées : la prise en compte des conditions météorologiques est capitale pour répondre aux deux enjeux que sont la sécurité routière et l'optimisation de la mobilité. Lorsqu'elles se dégradent, elles augmentent le risque de congestion et d'accident. Les conditions météorologiques dégradées ont un effet sur la visibilité ou encore l'adhérence. Ces critères essentiels peuvent gravement nuire à la sécurité routière. La neige et le brouillard entraînent d'ailleurs 15% de ce type de congestion. Les systèmes de vision artificielle sont justement utilisés pour limiter les situations d'accident et de congestion. Ces systèmes doivent donc conserver leur efficacité quelles que soient les conditions météorologiques rencontrées. En effet, ils sont déployés à l'échelle industrielle que ce soit au bord des voies ou à bord des véhicules, et leur sûreté de fonctionnement doit être démontrée dans de telles conditions. Les systèmes de vision utilisés dans le contexte routier sont par exemple pour la vidéo surveillance et les ADAS (systèmes avancés d'assistance et d'aide à la conduite). Ils permettent ainsi d'assurer la sécurité et faciliter la mobilité. Ces systèmes de vision sont généralement affectés par la dégradation des conditions météorologiques en présence par exemple de brouillard ou encore de pluie forte, phénomènes limitant la visibilité et réduisant ainsi la qualité des images. Afin d'optimiser les performances des systèmes de vision, il est nécessaire de disposer d'un système de détection fiable de ces conditions météorologiques défavorables.

Chapitre 1 : Étude bibliographique et Cahier de charge

I. Problématique

Les facteurs humains sont en cause dans plus de 90 % des accidents, d'après un certain nombre de recherches, dont notamment celle de Sabey (1983). Ils sont la cause exclusive de 65 % des accidents. Le potentiel est donc très important en termes de prévention. Cependant, ces facteurs sont difficiles à identifier, c'est pourquoi l'accent a été mis jusqu'à présent sur l'amélioration des véhicules et des infrastructures. En plus des facteurs humains, on peut noter aussi un impact très significatif de la dégradation des conditions climatique et météorologique. Les tempêtes tous comme les autres conditions climatiques cause des accidents routiers chaque année par exemple en France, entre 2018 et 2019, 568 accidents ont eu lieu dans des conditions de neige, soit près d'un accident sur 200. En outre, « la gravité des accidents par temps de brouillard ou par temps de neige/grêle est deux fois plus élevée que celle par temps sec ou par temps de pluie (avec une part de 10 % des accidents mortels, contre 5 %) », Précise le rapport. En vue de cette problématique il est donc nécessaire de proposer des systèmes d'aide à la conduite pour les véhicules de type autonomes sans exclure les véhicules ordinaires ou à conduite manuelles ou assisté.

II. Advanced Driver Assistance System (ADAS)

Le système ADAS est un système d'aide à la conduite possédant un niveau de sécurité élevé.

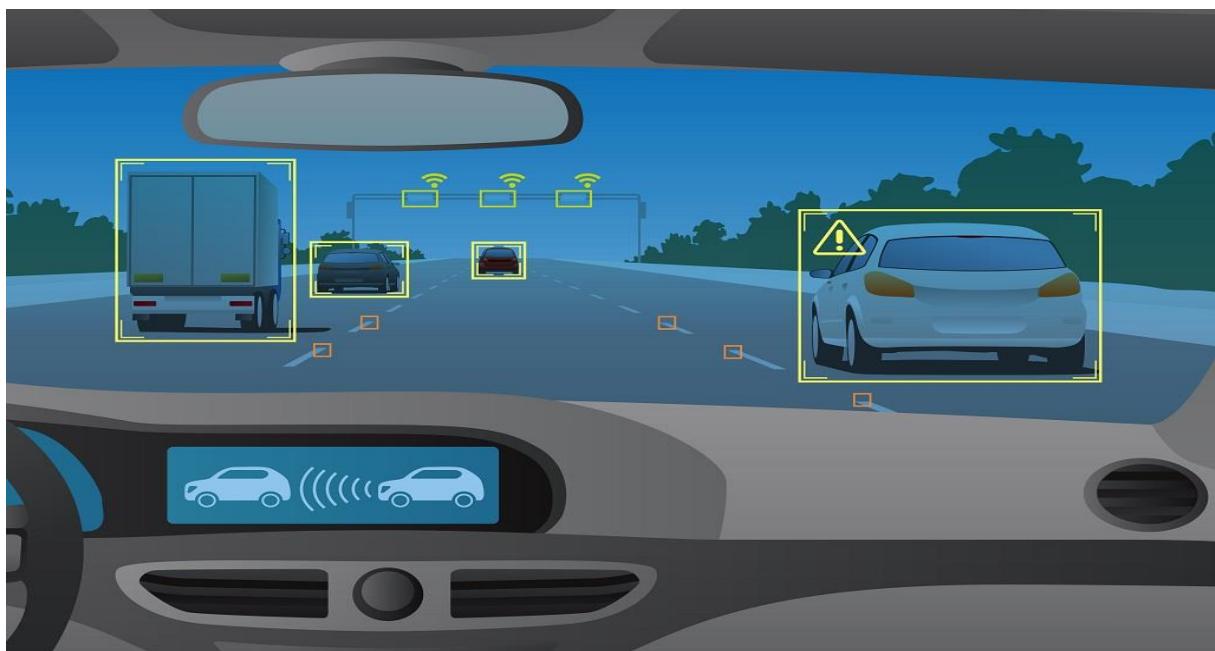


Figure 1 : système ADAS

En effet le système présente 6 selon Society of Automotive Engineers (SAE) niveaux qui sont les suivants :

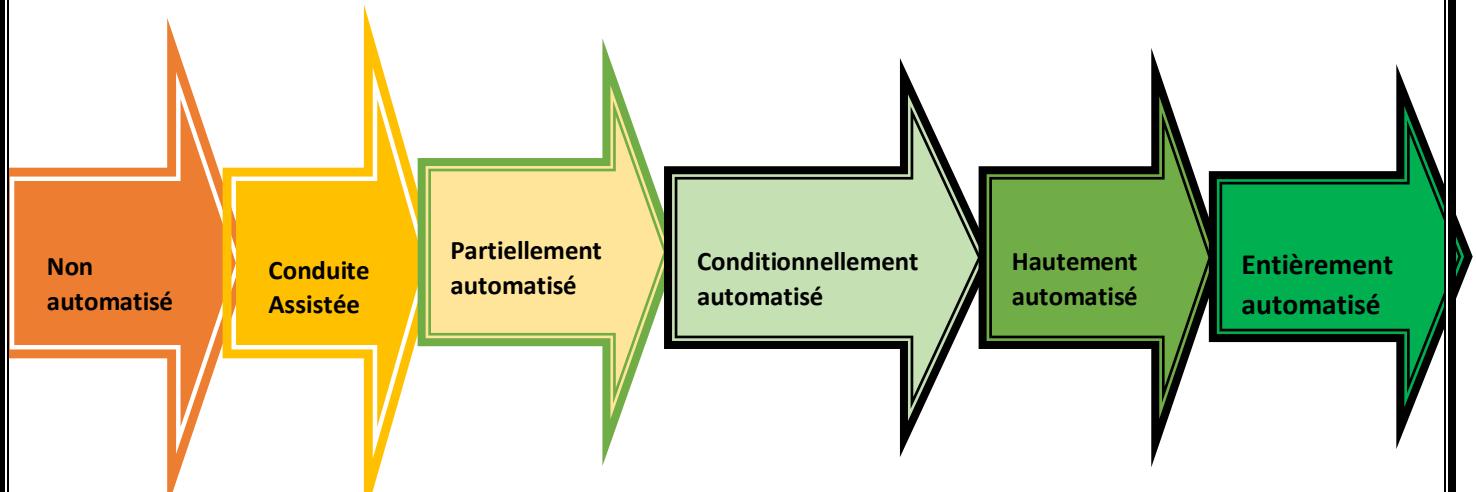


Figure 2 : niveau du système ADAS

- La surveillance totale de tous les aspects de la conduite
- Contrôle de direction ou d'accélération et décélération
- Automatisation partielle avec un ou plusieurs systèmes d'aide à la conduite qui contrôlent aussi bien les fonctions de direction que d'accélération et de décélération
- Contrôle de tous les aspects de la conduite dynamique, mais nécessite l'intervention du conducteur de manière appropriée lorsqu'il lui est demandé d'intervenir ;
- Contrôle de tous les aspects de la conduite dynamique, même quand le conducteur ne réagit pas de manière appropriée.
- Le contrôle en tout temps de tous les aspects de la conduite dynamique par un être humain, et ce, pour toutes les conditions routières.

III. Les réseaux de neurones convolutifs CNN

Les CNN désignent une sous-catégorie de réseaux de neurones et sont à ce jour un des modèles de classification d'images réputés être les plus performants.

Leur mode de fonctionnement est à première vue simple : l'utilisateur fournit en entrée une image sous la forme d'une matrice de pixels.

Celle-ci dispose de 3 dimensions :

- Deux dimensions pour une image en niveaux de gris.
- Une troisième dimension, de profondeur 3 pour représenter les couleurs fondamentales (Rouge, Vert, Bleu).

Contrairement à un modèle MLP (Multi Layers Perceptron) classique qui ne contient qu'une partie classification, l'architecture du Convolutional Neural Network dispose en amont d'une partie convulsive et comporte par conséquent deux parties bien distinctes :

- Une partie convulsive : Son objectif final est d'extraire des caractéristiques propres à chaque image en les compressant de façon à réduire leur taille initiale. En résumé, l'image fournie en entrée passe à travers une succession de **filtres**, créant par la même occasion de nouvelles images appelées cartes de convolutions. Enfin,

les cartes de convolutions obtenues sont concaténées dans un vecteur de caractéristiques appelé code CNN.

- Une partie classification : Le code CNN obtenu en sortie de la partie convulsive est fourni en entrée dans une deuxième partie, constituée de couches entièrement connectées appelées perceptron multicouche (MLP pour Multi Layers Perceptron). Le rôle de cette partie est de combiner les caractéristiques du code CNN afin de classer l'image.

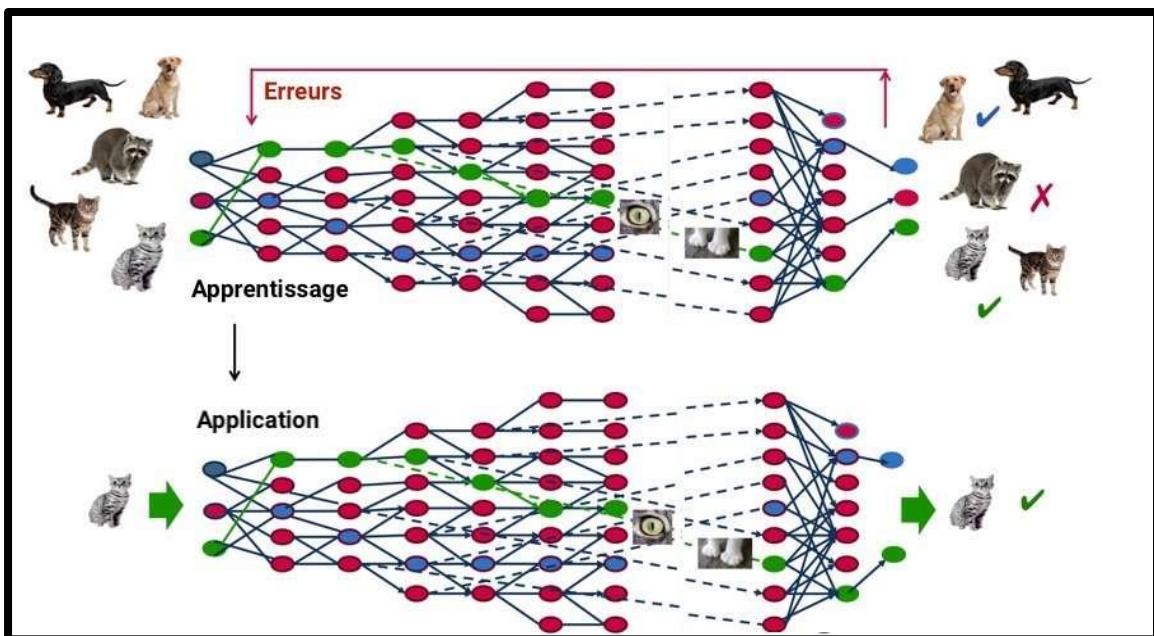


Figure 3 : perception Multicouche

IV. Cahier de charge

2) L'Expression du Besoin /objectifs du projet

Un véhicule autonome doit, par définition être capable de percevoir toutes les informations nécessaires à une navigation sans avoir d'incidents.

Le véhicule doit être capable de communiquer avec son environnement dans de nombreuses conditions, de plus le véhicule doit être capable de prendre des décisions et de planifier des actions de manières indépendantes.

Pour accomplir ces tâches, il est nécessaire que le véhicule reçoive des informations sur son environnement, ce qui inclus la nécessité d'avoir des données ou informations relatives aux différentes conditions climatiques puis les traiter afin de prendre des décisions importantes d'aide à la conduite.

Dans cette problématique notre objectif est d'utiliser des systèmes de détection (cameras, des capteurs...) qui permettent de fournir des inputs (des images ou des données numériques puis **classifier** ces inputs en utilisant les algorithmes de deep Learning.

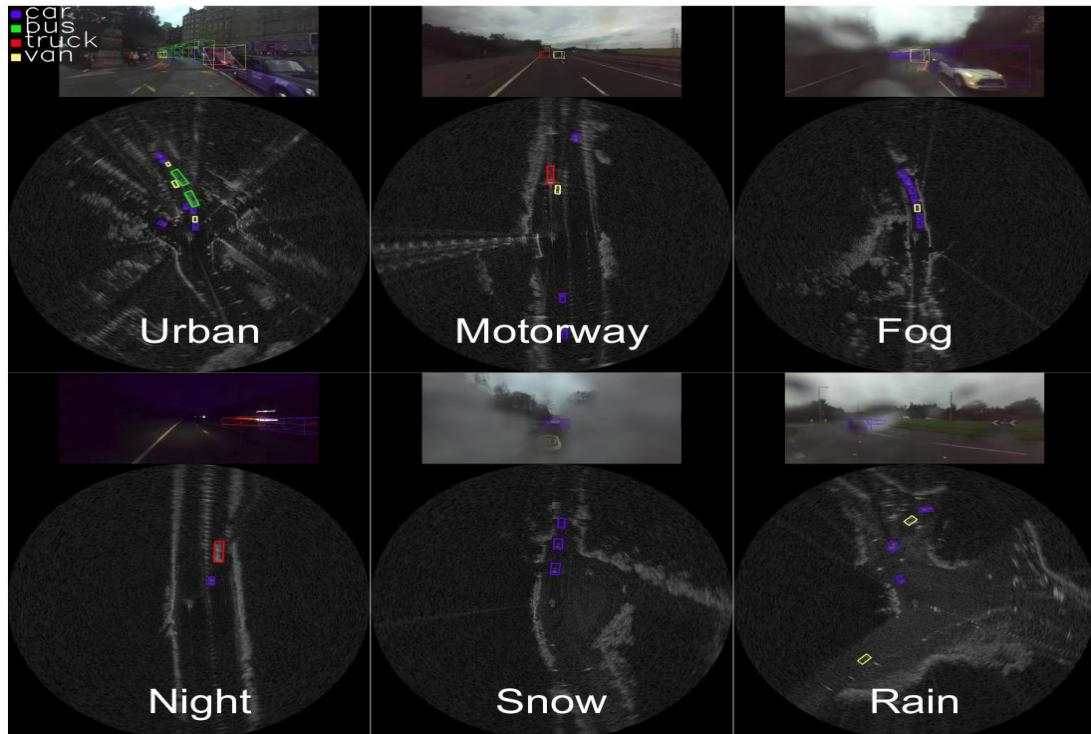


Figure 4 : Exemple de détection

3) Description fonctionnelle des besoins

Le système est muni d'une camera embarquée qui prend les images en temps réels, ces images seront ensuite les inputs qui sont traités par notre algorithme de deep Learning via les

réseaux de neurones convolutifs CNN, qui avec les résultats probabilistes fournis, nous identifiera la classe climatique détectée.

Cependant il doit respecter certaines exigences de fonctionnalités :

✓ **Une bonne précision**

Pas de fausse alerte ou données.

Identification rapide et efficace de la classe du climat en temps réel.

Capable de fonctionner que ça soit jour ou nuit.

✓ **La robustesse**

Capable de détecter toutes les conditions climatiques sur toutes les échelles et apparences et il doit être assez robuste.

✓ **Le coût**

Le système ne doit pas être trop cher pour étendre son accessibilité

4) Principe de fonctionnement et arbre de Processus du Système étudié

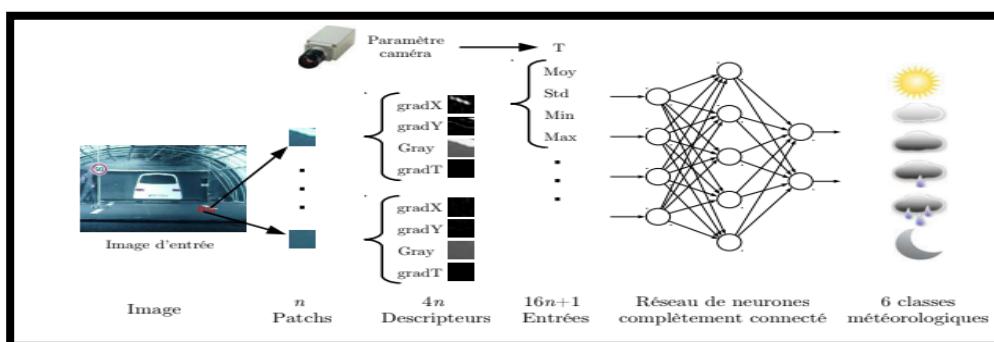
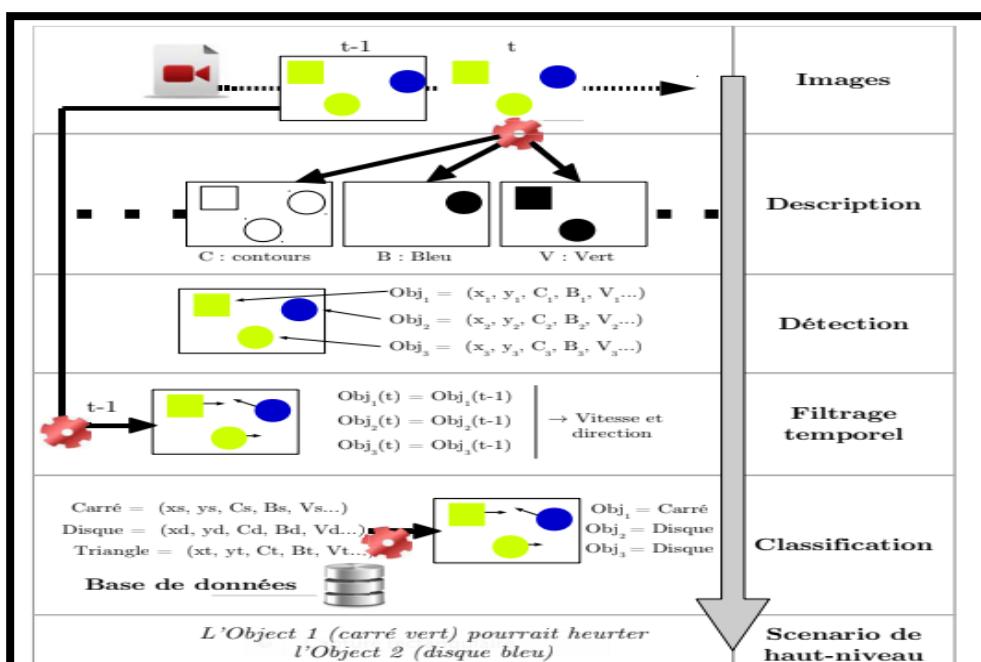


Figure 5 : Processus du système de détection

5) Périmètre du projet

De l'analyse de la problématique et les données qui circulent dans le cas des accidents, notre problématique est d'ordre international.

6) Délai de réalisation du projet

Le projet est livré par étape et il est étendu sur deux semestres.

7) Cahier de charge fonctionnelle

 La bête à corne

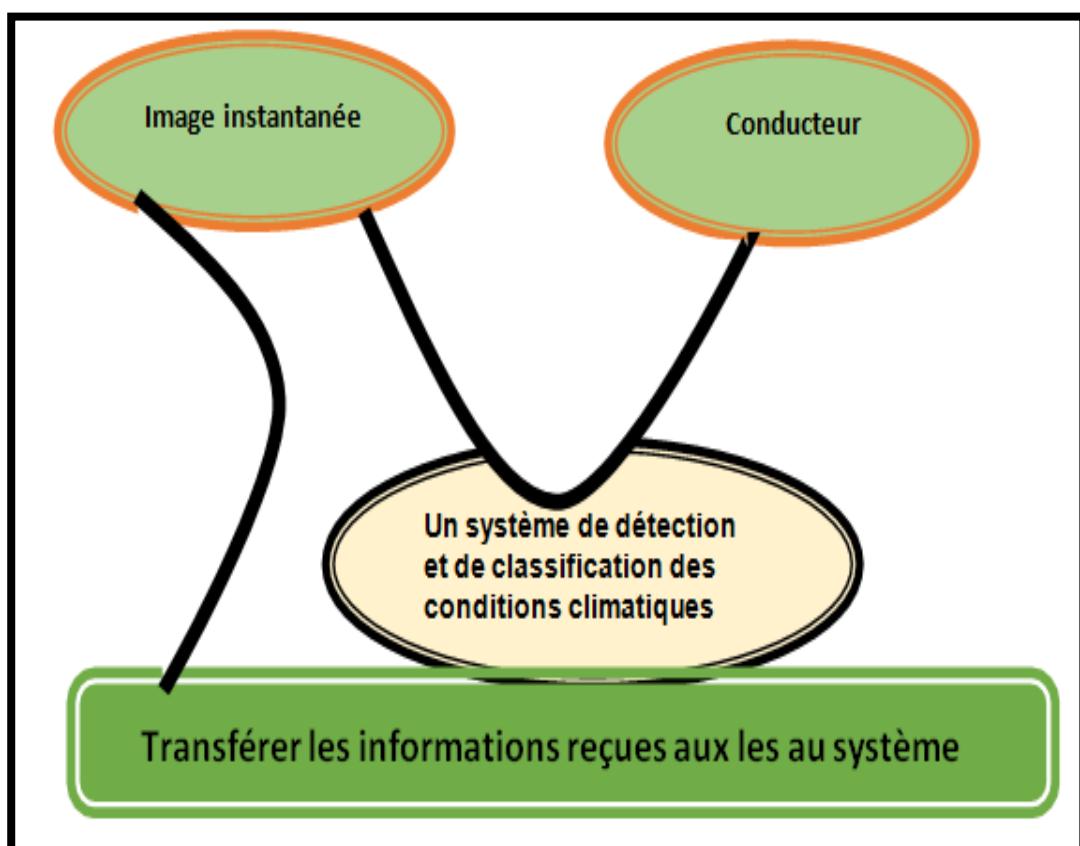


Figure 6 : Bête à corne

Diagramme de Pieuvre

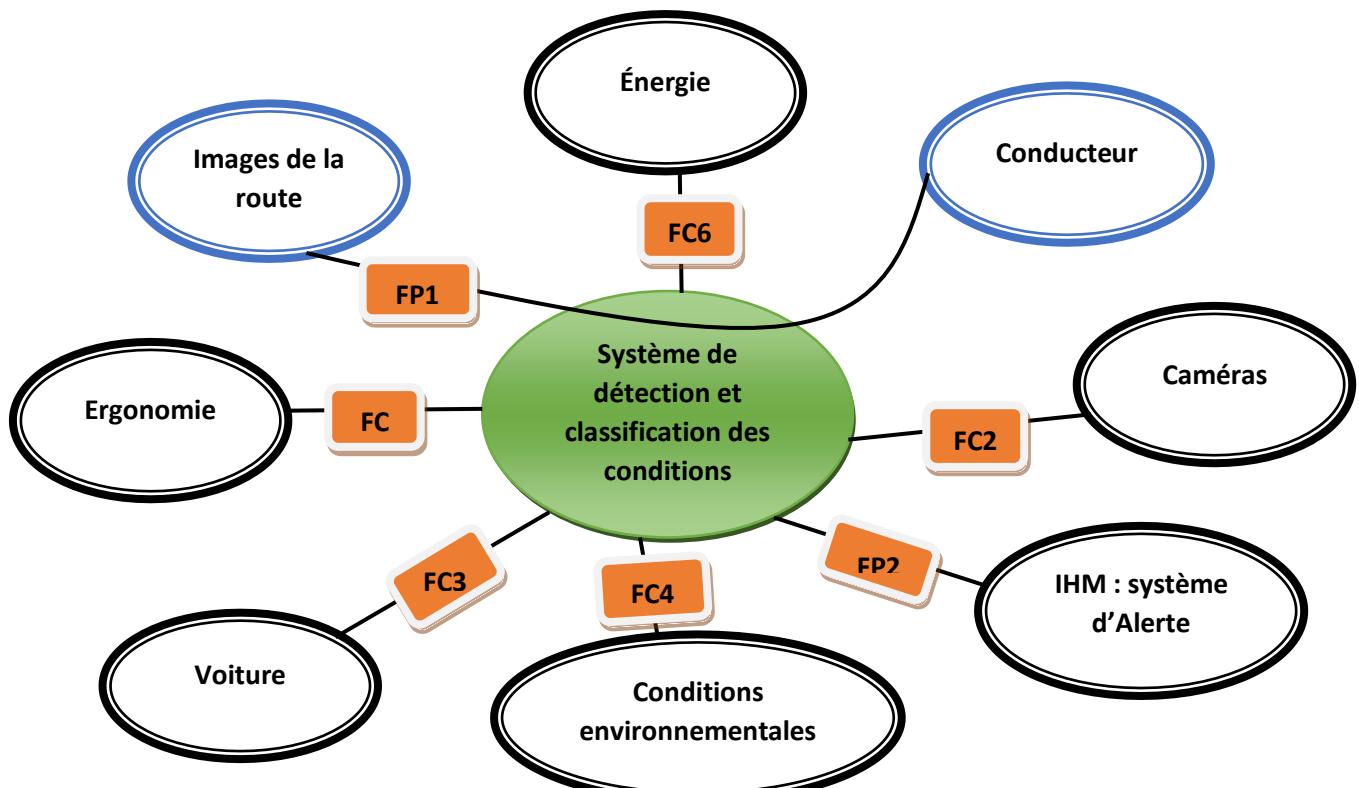


Figure 7 : diagramme de Pieuvre

Les fonctions contraintes

Fonctions	Expression des fonctions
FP1	Déetecter et classifier l'état climatique à partir des images prises par la caméra.
FP2	Informier le conducteur de l'état actuel du climat et de tout changement dans les conditions climatiques.
FC2	Se connecter à une caméra de résolution et puissance convenables.
FC3	Positionner sur une voiture.
FC4	S'adapter aux conditions d'environnement du milieu à traiter : niveau de luminosité, éclairage, couleur...
FC5	Être stable et pratique.

V. Conclusion chapitre 1

Nous avons fais une étude bibliographique horizontale de l'ensemble du projet.

Dans le chapitre suivant, nous ferons une étude comparatives des différents système de détections (cameras, capteurs, radar...).

Chapitre 2 : Étude des systèmes de détection

I.Introduction

Dans ce chapitre, nous allons faire une étude comparative des différents systèmes de détection (capteurs, caméra...) puis un choix de système de détection.

II.Classifications des conditions climatiques

Il existe plusieurs conditions climatiques qui varient en fonction de localité.

Le tableau suivant présente plusieurs situations de conditions climatiques :

Les climats	Clair ou clair+nuageux
	Tempête de sable
	Pluvieux
	Neige
	La grêle
	Le brouillard

Chacune de ces classes de climat présente des dangers lorsqu'elles sont dans l'état critique ou dégradé.

Afin de mieux s'y prendre ou éviter des problèmes de conduction lors de ces situations climatiques, nous allons faire une étude comparative et descriptive des capteurs et caméra pouvant fournir des images ou des informations nécessaires qui décrivent chaque classe de climat étudiée.

III.Les différents types de capteurs

Il existe plusieurs types de capteurs en fonctions du domaine d'application.

- Capteurs d'accélération
- Capteur de pression
- Capteur de mouvement
- Capteur de luminosité

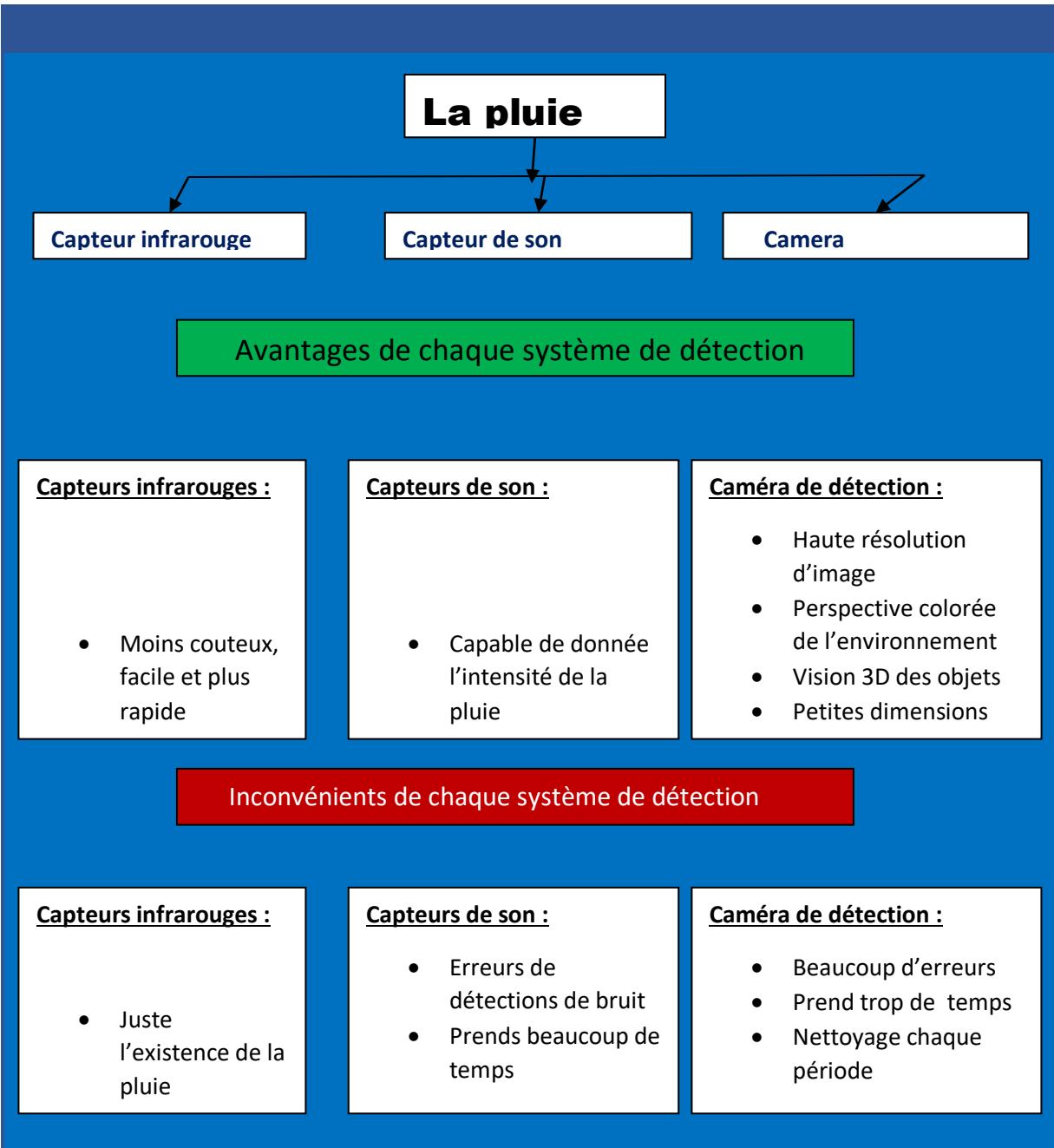
- Capteur de température
- Capteur de son
- Capteur d'humidité

Dans cette liste de capteur nous allons juste utiliser :

- Capteur d'image
- Capteur d'humidité
- Capteur de luminosité
- Capteur de température
- Capteur de son

IV. Étude comparative des systèmes de détections selon chaque classe de climat

Dans cette étude, nous feront ressortir les avantages et les inconvénients de système de détections afin de pouvoir avoir une idée sur leur efficacité.



La neige

Capteur infrarouge

Camera

Avantages de chaque système de détection

Capteurs infrarouges :

- Moins couteux, facile et plus rapide

Exemple de capteurs :

- Stone Scan
- Snow Scan
-

Caméra de détection :

- Haute résolution d'image
- Perspective colorée de l'environnement
- Vision 3D des objets
- Petites dimensions

Inconvénients de chaque système de détection

Capteurs infrarouges :

- Juste l'existence de la pluie

Caméra de détection :

- Beaucoup d'erreurs
- Prend trop de temps
- Nettoyage chaque période

Le brouillard

Radar+ camera

Camera de détection

Avantages de chaque système de détection

Radar+camera

- Longue portée de détection
- Pas d'erreur
- Petit et léger
- Moins d'énergie
- Moins chère que lidar

Caméra de détection :

- Haute résolution d'image
- Perspective colorée de l'environnement
- Vision 3D des objets
- Petites dimensions

Inconvénients de chaque système de détection

Radar:

- Non efficace
- Temps de prédiction énorme

Caméra de détection :

- Beaucoup d'erreurs
- Prend trop de temps
- Nettoyage chaque période
- Confusion avec temps nuageux

Les tempêtes de sable

Radar+ camera

Caméra de détection

Avantages de chaque système de détection

Radar+camera

- Longue portée de détection
- Pas d'erreur
- Petit et léger
- Moins d'énergie
- Moins chère que lidar

Caméra de détection :

- Haute résolution d'image
- Perspective colorée de l'environnement
- Vision 3D des objets
- Petites dimensions

Inconvénients de chaque système de détection

Radar:

- Non efficace
- Temps de prédiction énorme

Caméra de détection :

- Beaucoup d'erreurs
- Prend trop de temps
- Nettoyage chaque période
- Confusion avec temps nuageux

- ❖ Nous avons étudié chaque classe de climat (choisie) et ses systèmes de détections possibles ainsi que leurs avantages et inconvénients. Ce qui nous laisse une large gamme de choix de système de détection.
- ❖ L'utilisation d'un laser est envisageable mais semble tout de même limitée, compte tenu des réglages nécessaires pour détecter correctement la densité du brouillard. Le radar est peu perturbée par les conditions météorologiques et ne peut donc servir de détecteur de ces dites conditions.
- ❖ Les cameras semblent être adaptées pour notre objectifs

La pluie

La pluie est la forme de précipitation la plus répandue. Elle consiste en des gouttelettes d'eau devenues suffisamment lourdes dans les nuages pour tomber à la surface de la terre.

Type de précipitation	Forme	Diamètre de goutte (mm)	Vitesse de chute ($m.s^{-1}$)
Pluie	Liquide	0,5 - 3	4 - 8
Bruine	Liquide	0,1 - 0,4	< 0,5
Brouillard	Liquide	0,001 - 0,01	$3e^{-5} - 3e^{-3}$
Grêle	Solide	5 - 50	4 - 30
Neige	Solide	5 - 25	1 - 1,5

Classe de pluie	Intensité de pluie R_R ($mm.h^{-1}$)
Très faible	0,0 - 0,1
Faible	0,1 - 2,5
Modérée	2,5 - 7,5
Forte	> 7,5

Le brouillard

Le nuage de vapeur d'eau condensée en suspension dans l'air près du niveau de la surface de la Terre s'appelle brouillard. Cette vapeur est composée de gouttelettes d'eau, dont le diamètre ne dépasse pas quelques centaines de microns, mais lors de la diffusion et de la réflexion du signal, une dégradation se produit ou empêche totalement la transmission de la lumière à travers l'environnement.

Types du brouillard	Classes du brouillard routier	Distance de visibilité météorologique (m)
Brouillard météorologique	-	< 1000
Brouillard routier	1	200 - 400
	2	100 - 200
	3	50 - 100
	4	< 50

Niveau de problème engendré

La pluie :

- ✓ Problème de visibilité
- ✓ Dérapage de la voiture
- ✓ Éclaboussement

La neige :

- ✓ Problème de visibilité
- ✓ Glissement
- ✓ Collisions
- ✓ Chocs
- ✓ Consommation d'énergie/baisse de la température

Le brouillard :

- ✓ Problème de visibilité

Tempêtes de sables :

- ✓ Problème de visibilité
- ✓ Dérapage de la voiture

V. Conclusion chapitre 2

Ainsi, nous avons fais une étude sur les systèmes de détection.

Donc le chapitre suivant, nous feront la classification des images.

Chapitre 3 : Implémentation, programmation/Classification des images

I. Introduction

Après avoir parlé transversalement sur la détection des images et les systèmes de détection ainsi que leurs domaines d'application, il est nécessaire maintenant de passer à leur classification selon la classe climatique identifier afin de pouvoir traiter les problèmes de conduites liés aux dégradations de ces classes climatiques.

II. Classification des images

La classification automatique des images consiste à attribuer automatiquement une classe à une image à l'aide d'un système de classification. On retrouve ainsi la classification d'objets, de scènes, de textures, la reconnaissance de visages, d'empreintes digitale et de caractères. Il existe deux principaux types d'apprentissage : **l'apprentissage supervisé** et **l'apprentissage non-supervisé**. Dans l'approche supervisée, chaque image est associée à une étiquette qui décrit sa classe d'appartenance. Dans l'approche non supervisée les données disponibles ne possèdent pas d'étiquettes. **Dans notre travail on s'intéresse de l'approche supervisée.**

1) Notion de base d'une image

Une image est une représentation planaire d'une scène ou d'un objet situé en général dans un espace tridimensionnel, elle est issue du contact des rayons lumineux provenant des objets formants la scène avec un capteur (caméra, scanner, rayons X, ...). Il ne s'agit en réalité que d'une représentation spatiale de la lumière. L'image est considérée comme un ensemble de points auquel est affectée une grandeur physique (luminance, couleur).

2) Quelques Caractéristiques d'une image

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

❖ **Pixel**

Le pixel est l'abréviation du mot « Picture élément », est une unité de surface permettant de définir la base d'une image numérique.

❖ **Dimensions et la résolution**

La dimension est la taille de l'image. Elle se présente sous forme d'une matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image. Par

contre, la résolution est la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production d'images.

3) Méthode de classification d'image

Méthode supervisée

L'objectif de la classification supervisée est principalement de définir des règles permettant de classer des objets dans des classes à partir de variables qualitatives ou quantitatives caractérisant ces objets. On dispose au départ d'un échantillon dit d'apprentissage dont le classement est connu. Cet échantillon est utilisé pour l'apprentissage des règles de classement. Il est nécessaire d'étudier la fiabilité de ces règles pour les comparer et les appliquer, évaluer les cas de sous apprentissage ou de sur apprentissage (complexité du modèle). On utilise souvent un deuxième échantillon indépendant, dit de validation ou de test.

Méthode non supervisée

Procède de la façon contraire. C'est à dire ne nécessitant aucun apprentissage et aucune tâche préalable d'étiquetage manuel. Elle consiste à représenter un nuage des points d'un espace quelconque en un ensemble de groupes appelé Cluster. Il lié généralement au domaine de l'analyse des données comme l'ACP. Un «Cluster» est une collection d'objets qui sont «similaires» entre eux et qui sont «dissemblables » par rapport aux objets appartenant à d'autres groupes.

4) Indicateur de performance de la classification

La matrice de confusion

Prenons l'exemple d'un classificateur binaire qui prédit deux classes binaires classe 0 et classe 1. Pour mesurer les performances de ce classificateur, il est d'usage de distinguer 4 types d'éléments classés pour la classe voulue :

- ✓ **Vrai positif VP** : Élément de la classe 1 correctement prédit
- ✓ **Vrai négatif VN** : Élément de la classe 0 correctement prédit
- ✓ **Faux positif FP** : Élément de la classe 1 mal prédit
- ✓ **Faux négatif FN** : Élément de la classe 0 mal prédit

Ces informations peuvent être rassemblées et visualisées sous forme de tableau dans une matrice de confusion. Dans le cas d'un classificateur binaire, on obtient :

Classe réelle	Classe prédite	
	Classe 0	Classe 1
Classe 0	VN	FN
Classe 1	FP	VP

Si cette matrice de confusion est diagonale alors la classification est parfaite.

Il est possible de calculer plusieurs indicateurs résumant la matrice de confusion. Par exemple si nous souhaitons rendre compte de la qualité de la prédiction sur la classe 1, on définit :

- **Précision** : qui est la proportion d'élément bien classée pour une classe donnée.

$$\text{précision}_{\text{classe } 1} = \frac{VP}{VP + FP}$$

- **Rappel** : proportion d'éléments bien classés par rapport au nombre d'élément de la classe prédictive.

$$\text{rappel}_{\text{classe } 1} = \frac{VP}{VP + FN}$$

- **F-mesure** : mesure du compromis entre précision et rappel

$$F - \text{Mesure}_{\text{classe } 1} = \frac{2 * \text{precision} * \text{rappel}}{\text{precision} + \text{rappel}}$$

III.Le Deep Learning

Maintenant, nous allons parler des réseaux de neurones, comment fonctionnent les réseaux de neurones de manière générale, nous allons aborder le domaine du Deep Learning. Cette famille d'algorithme a permis de faire des progrès importants dans les domaines de la classification des images et du traitement du langage par exemple.

1) Algorithmes de Deep Learning

Quelques exemples d'algorithme de Deep Learning :

Le Deep Neural Networks ou DNN

Ce sont des **réseaux de neurones profonds** qui utilisent beaucoup de couches cachées. L'augmentation du nombre de couches, permet à un réseau de neurones de détecter de légères variations du modèle d'apprentissage, favorisant le sur-apprentissage ou sur-ajustement « overfitting ».

Convolutional Neural Networks ou CNN

Ici le problème est divisé en sous parties, et pour chaque partie, un «cluster» de neurones sera créer afin d'étudier cette portion spécifique. Par exemple, pour une image en couleur, il est possible de diviser l'image sur la largeur, la hauteur et la profondeur (les couleurs).

Le Deep Belief Networks

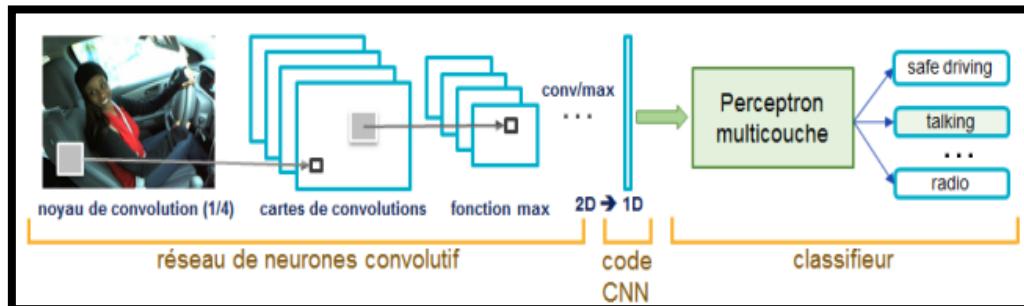
Ces algorithmes fonctionnent suivant une première phase non supervisée, suivi de l'entraînement classique supervisé. Cette étape

d'apprentissage non-supervisée, permet, en outre, de faciliter l'apprentissage supervisé.

Parmi les algorithmes décrits, nous allons travailler avec les réseaux de neurone convolutionnel CNN.

2) Les réseaux de neurone convolutionnel CNN

Les réseaux de neurone convolutionnel sont à ce jour les modèles les plus performants pour classer des images. Désignés par l'acronyme CNN, de l'anglais Convolutional Neural Network, ils comportent deux parties bien distinctes. En entrée, une image est fournie sous la forme d'une matrice de pixels. Elle a 2 dimensions pour une image en niveaux de gris. La couleur est représentée par une troisième dimension, de profondeur 3 pour représenter les couleurs fondamentales [Rouge, Vert, Bleu]. La première partie d'un CNN est la partie convulsive à proprement parler. Elle fonctionne comme un extracteur de caractéristiques des images. Une image est passée à travers une succession de filtres, ou noyaux de convolution, créant de nouvelles images appelées cartes de convolutions. Certains filtres intermédiaires réduisent la résolution de l'image par une opération de maximum local. Au final, les cartes de convolutions sont mises à plat et concaténées en un vecteur de caractéristiques, appelé code CNN.



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

a) Couche de Convolution CONV

La couche de convolution est le bloc de construction de base d'un CNN. Trois paramètres permettent de dimensionner le volume de la couche de convolution : la profondeur, le pas et la marge. 1. Profondeur de la couche : nombre de noyaux de

convolution (ou nombre de neurones associés à un même champ récepteur). 2. Le pas: contrôle le chevauchement des champs récepteurs. Plus le pas est petit, plus les champs récepteurs se chevauchent et plus le volume de sortie sera grand. 3. La marge (à 0) ou zéro padding : parfois, il est commode de mettre des zéros à la frontière du volume d'entrée. La taille de ce 'zero-padding' est le troisième hyper paramètre. Cette marge permet de contrôler la dimension spatiale du volume de sortie.

b) **Couche de Pooling POOL**

Le Pooling réduit la taille spatiale d'une image intermédiaire, réduisant ainsi la quantité de paramètres et de calcul dans le réseau. Il est donc fréquent d'insérer périodiquement une couche de Pooling entre deux couches convective successives d'une architecture CNN pour contrôler l'overfitting (sur-apprentissage). L'opération de Pooling créait aussi une forme d'invariance par translation.

c) **Couche de Correction ReLU**

Il est possible d'améliorer l'efficacité du traitement en intercalant entre les couches de traitement une couche qui va opérer une fonction mathématique (fonction d'activation) sur les signaux de sortie. La fonction ReLU (abréviation de Unités Rectifiée linéaires) : $F(x)=\max(0, x)$ Cette fonction force les neurones à retourner des valeurs positives.

d) **Couche de perte LOSS**

La couche de perte spécifie comment l'entraînement du réseau pénalise l'écart entre le signal prévu et réel. Elle est normalement la dernière couche dans le réseau. Diverses fonctions de perte adaptées à différentes tâches peuvent y être utilisées. La fonction « Softmax » permet de calculer la distribution de probabilités sur les classes de sortie.

IV.Implémentation/Programmation

Dans cette partie, on va définir l'architecture du modèle et par la suite on va appliquer ce modèle sur la base d'images DAWN et Foggy ou sur le dataset collecté. Pour cela, on va travailler avec les bibliothèques Tensorflow et Keras pour l'apprentissage et la classification et afin d'améliorer les performances du modèle.

1) **Logiciels et Librairies utilisés**

Tensorflow

Tensorflow est un Framework de programmation pour le calcul numérique qui a été rendu Open Source par Google en Novembre 2015. Depuis son release, Tensorflow n'a cessé de gagner en popularité, pour devenir très rapidement l'un des Framework les plus utilisés pour le Deep Learning et donc les réseaux de neurones. Son nom est notamment inspiré du fait que les opérations courantes sur des réseaux de neurones sont principalement faites via des tables de données multidimensionnelles, appelées Tenseurs (Tensor). Un Tensor à deux dimensions est l'équivalent d'une matrice. Aujourd'hui, les principaux produits de Google sont basés sur Tensorflow: Gmail, Google Photos, Reconnaissance de voix.

Keras

Keras est une API de réseaux de neurones de haut niveau, écrite en Python et capable de fonctionner sur Tensorflow ou Theano. Il a été développé en mettant l'accent sur l'expérimentation rapide. Être capable d'aller de l'idée à un résultat avec le moins de délai possible est la clé pour faire de bonnes recherches.

Python

Python est un langage de programmation de haut niveau interprété (il n'y a pas d'étape de compilation) et orienté objet avec une sémantique dynamique et adaptée dans le domaine de la data science et IA.

Aperçu de librairie utilisé dans Google colab et Jupyter notebook

```
import matplotlib.pyplot as plt
import numpy as np
import os
import PIL
import glob
import seaborn as sn
import tensorflow as tf
import pandas as pd
import seaborn as sunflower_path
import pickle
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.models import Sequential
from keras.preprocessing.image import image_dataset_from_directory
```

2) Configuration du Modèle (Séquentiel)

Ce schéma nous donne un aperçu de la configuration et l'architectez du model

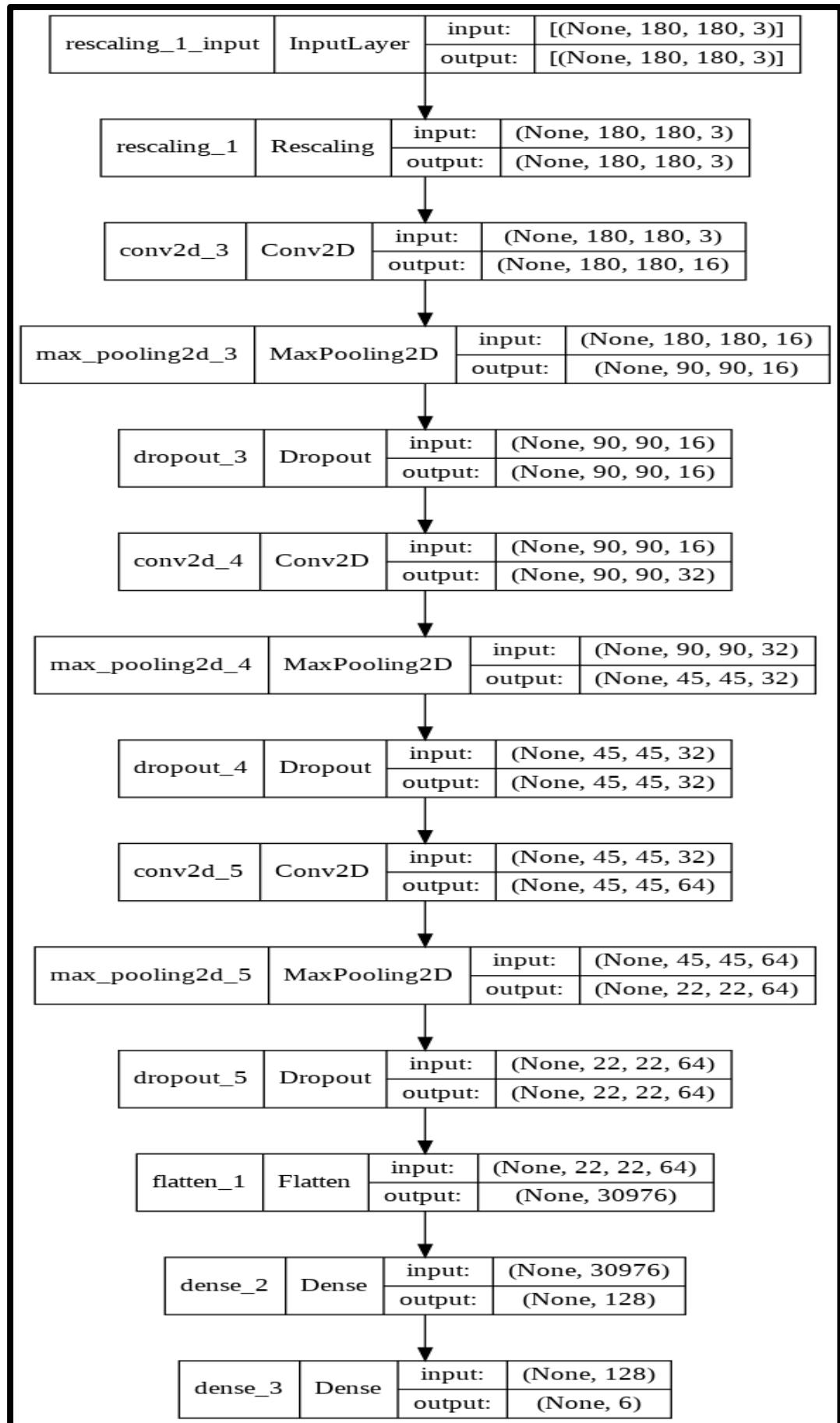


Figure 8 : Configuration du model

```

Model: "sequential_2"
-----  

Layer (type)          Output Shape       Param #
-----  

rescaling_2 (Rescaling)    (None, 180, 180, 3)   0  

conv2d_6 (Conv2D)         (None, 180, 180, 16)  448  

max_pooling2d_6 (MaxPooling 2D) (None, 90, 90, 16)  0  

dropout_6 (Dropout)        (None, 90, 90, 16)   0  

conv2d_7 (Conv2D)         (None, 90, 90, 32)   4640  

max_pooling2d_7 (MaxPooling 2D) (None, 45, 45, 32)  0  

dropout_7 (Dropout)        (None, 45, 45, 32)   0  

conv2d_8 (Conv2D)         (None, 45, 45, 64)   18496  

max_pooling2d_8 (MaxPooling 2D) (None, 22, 22, 64)  0  

dropout_8 (Dropout)        (None, 22, 22, 64)   0  

flatten_2 (Flatten)        (None, 30976)        0  

dense_4 (Dense)           (None, 128)         3965056  

dense_5 (Dense)           (None, 6)           774  

-----  

Total params: 3,989,414  

Trainable params: 3,989,414

```

Le modèle est composé de 3 couches de convolution et 3 couches de maxpooling et de 3 dropout . L'image en entrée est de taille 180*180, l'image passe d'abord à une phase de normalisation entre [0,1]. Ensuite elle passe par la première couche de convolution composée de 3*3 filtres de taille 180*180, la fonction d'activation ReLU est utilisé, cette fonction d'activation force les neurones à retourner des valeurs positives, après cette convolution 180 features maps de taille 180*180 et de filtre 16*16 seront créés. Ensuite on applique la fonction maxpooling enfin la fonction dropout. on répète les mêmes opérations 3 fois avant de mettre a plat par la fonction flatten.

A. Projection sur le dataset utilisé

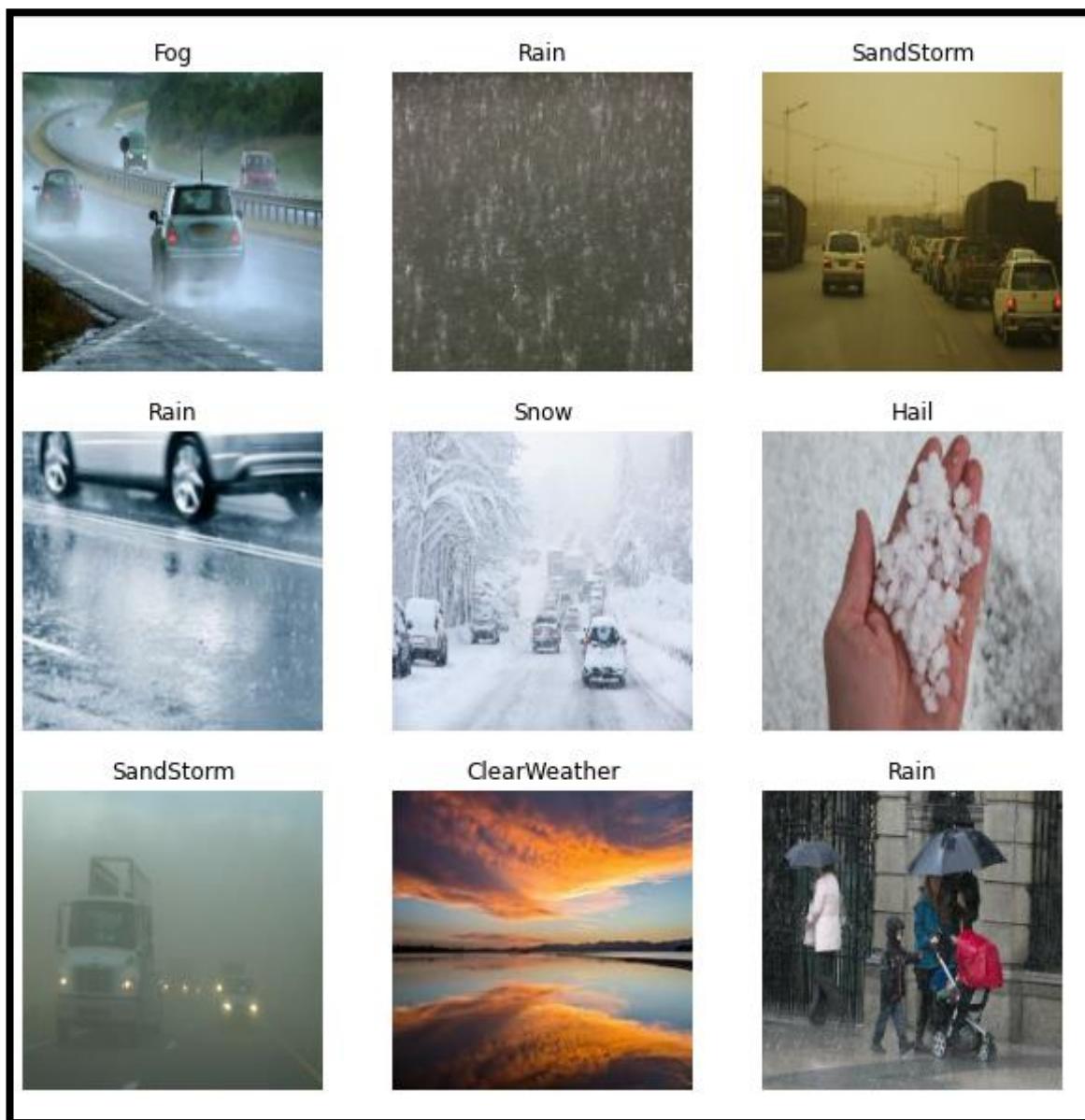


Figure 9 : projection sur le dataset

3) Résultats obtenus après classification

a) Résultats courants pendant la formation du modèle

Ces résultats sont obtenus sur une formation de 20 périodes largement suffisant.

Epoch 1/20

66/66 [=====] - 19s 90ms/step - loss: 1.5448 - accuracy: 0.4144 - val_loss: 1.2292 - val_accuracy: 0.5602

Epoch 2/20

66/66 [=====] - 6s 86ms/step - loss: 0.9906 - accuracy: 0.6458 - val_loss: 1.0081 - val_accuracy: 0.6616

Epoch 3/20

66/66 [=====] - 6s 85ms/step - loss: 0.7922 - accuracy: 0.7098 - val_loss: 0.8388 - val_accuracy: 0.7036

Epoch 4/20

66/66 [=====] - 6s 85ms/step - loss: 0.6690 - accuracy: 0.7596 - val_loss: 0.8769 - val_accuracy: 0.6654

Epoch 5/20

66/66 [=====] - 6s 85ms/step - loss: 0.6018 - accuracy: 0.7782 - val_loss: 0.7894 - val_accuracy: 0.7400

Epoch 6/20

66/66 [=====] - 6s 84ms/step - loss: 0.5597 - accuracy: 0.7940 - val_loss: 0.8149 - val_accuracy: 0.7208

Epoch 7/20

66/66 [=====] - 6s 85ms/step - loss: 0.5098 - accuracy: 0.8074 - val_loss: 0.8033 - val_accuracy: 0.7055

Epoch 8/20

66/66 [=====] - 6s 86ms/step - loss: 0.4833 - accuracy: 0.8255 - val_loss: 0.7886 - val_accuracy: 0.6922

Epoch 9/20

66/66 [=====] - 6s 88ms/step - loss: 0.4096 - accuracy: 0.8561 - val_loss: 0.7052 - val_accuracy: 0.7514

Epoch 10/20

66/66 [=====] - 6s 86ms/step - loss: 0.3811 - accuracy: 0.8709 - val_loss: 0.8166 - val_accuracy: 0.7189

Epoch 11/20

66/66 [=====] - 6s 85ms/step - loss: 0.3978 - accuracy: 0.8619 - val_loss: 0.6787 - val_accuracy: 0.7725

Epoch 12/20

66/66 [=====] - 6s 85ms/step - loss: 0.3223 - accuracy: 0.8891 - val_loss: 0.7299 - val_accuracy: 0.7839

Epoch 13/20

66/66 [=====] - 6s 85ms/step - loss: 0.2891 - accuracy: 0.8996 - val_loss: 0.8667 - val_accuracy: 0.7438

Epoch 14/20

66/66 [=====] - 6s 85ms/step - loss: 0.2469 - accuracy: 0.9116 - val_loss: 0.7605 - val_accuracy: 0.7859

Epoch 15/20

66/66 [=====] - 6s 85ms/step - loss: 0.2334 - accuracy: 0.9230 - val_loss: 0.7693 - val_accuracy: 0.7610

Epoch 16/20

66/66 [=====] - 6s 84ms/step - loss: 0.2310 - accuracy: 0.9159 - val_loss: 0.8342 - val_accuracy: 0.7476

Epoch 17/20

66/66 [=====] - 6s 85ms/step - loss: 0.2092 - accuracy: 0.9288 - val_loss: 0.8289 - val_accuracy: 0.7533

Epoch 18/20

66/66 [=====] - 6s 85ms/step - loss: 0.1816 - accuracy: 0.9393 - val_loss: 0.9552 - val_accuracy: 0.7572

```

Epoch 19/20
66/66 [=====] - 5s 84ms/step - loss: 0.1884 - accuracy: 0.9359 - val_loss:
0.8013 - val_accuracy: 0.7954
Epoch 20/20
66/66 [=====] - 6s 84ms/step - loss: 0.1424 - accuracy: 0.9469 - val_loss:
0.9971 - val_accuracy: 0.7725

```

1. Graphiques des résultats prédictifs gains et pertes d'entraînement et validation

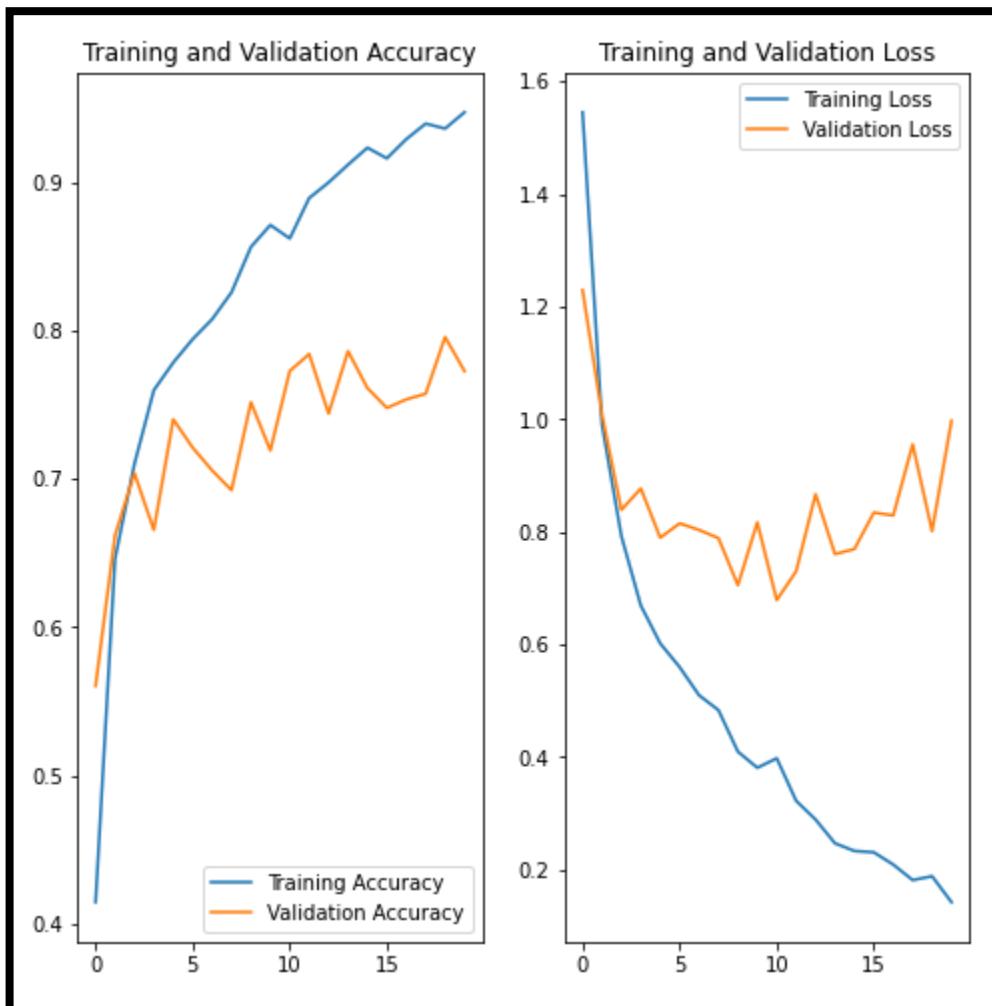


Figure 10 : graphique résultats prédictifs

D'après la Figure 10 La précision de l'apprentissage et de test augmente avec le nombre d'époque, ceci reflète qu'à chaque époque le modèle apprenne plus d'informations. Si la précision est diminuée alors on aura besoin de plus d'information pour faire apprendre notre modèle et par conséquent on doit augmenter le nombre d'époque et vice versa. De même, l'erreur d'apprentissage et de la validation diminue avec le nombre d'époque. Nous remarquons aussi que la totalité des images mal classées est de 74 images, un taux d'erreur de 38.94% et la totalité des images bien classées est de 116 un taux de précision de 61.05% pour un dataset de 190 images pour toutes les classes.

b) Matrice de confusion du modèle après prédictions sur de nouvelles données.

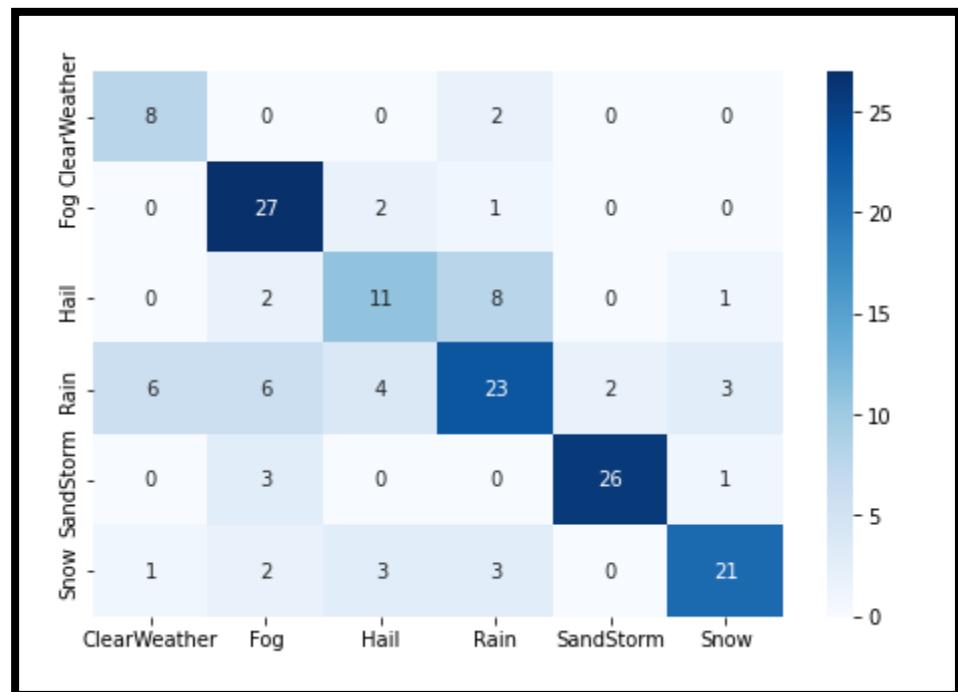


Figure 11 : Matrice de confusion

Classe d'image	Nombre image bien classé
ClearWeather	8/17
Fog	27/43
Hail	11/29
Rain	23/34
SandStorm	26/32
Snow	21/35

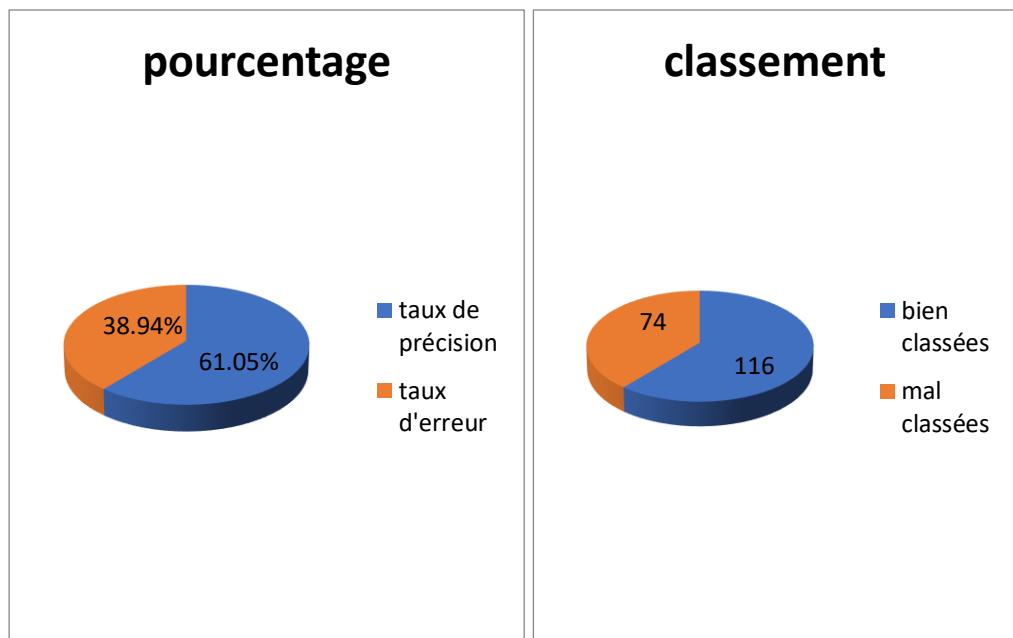


Figure 12 : graphique à secteur de la précision et erreurs

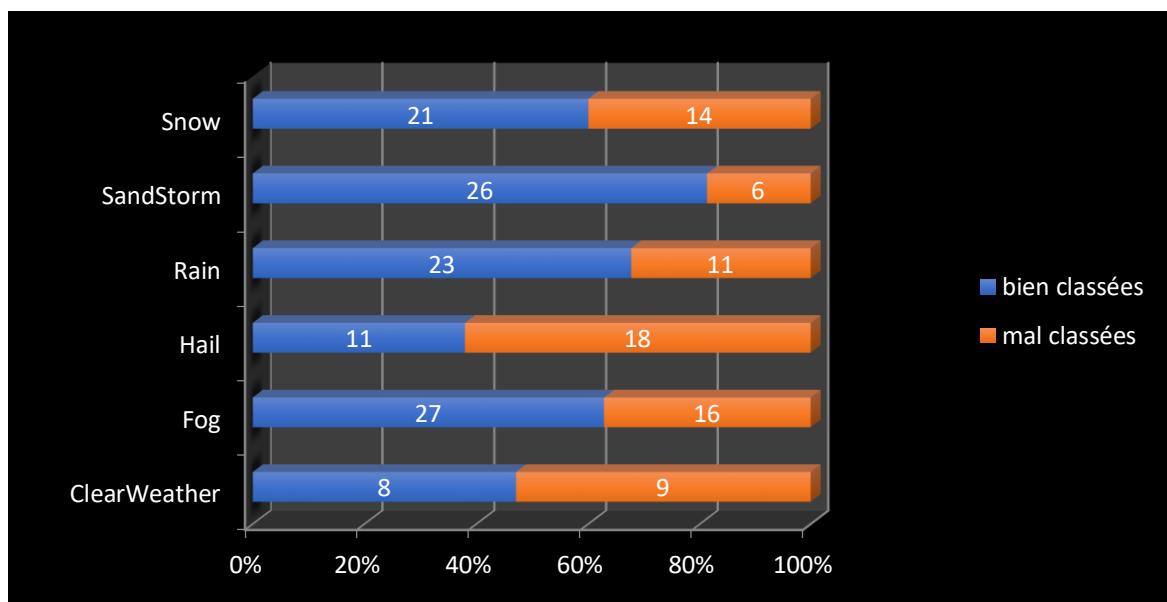


Figure 13 : précision par classe

c) Analyse de performance

Dans cette section, nous allons faire une analyse de performance du model par classe prédictive.

```
TP= cm.iloc[0,0]
FP=cm.iloc[0,:].sum()-TP
FN=cm.iloc[:,0].sum()-TP
TN=cm.sum().sum()-TP-FP-FN
accuracy=(TP+TN)/cm.sum().sum()
precision=TP/(TP+FP)
recall=TP/(TP+FN)
F1_Score=(2*precision*recall)/(precision+recall)

for i in range(cm.shape[0]):
    TP= cm.iloc[i,i]
    FP=cm.iloc[i,:].sum()-TP
    FN=cm.iloc[:,i].sum()-TP
    TN=cm.sum().sum()-TP-FP-FN
    accuracy=(TP+TN)/cm.sum().sum()
    precision=TP/(TP+FP)
    recall=TP/(TP+FN)
    F1_Score=(2*precision*recall)/(precision+recall)
    print("les resultats de la classe",cm.index[i],"\\n")
    print("accuracy=",+accuracy)
    print("precision=",+precision)

print("recall=",+recall)
print("F1_Score=",+F1_Score)
print("\\n")
```

Tableau récapitulatif de la performance du model

	ClearWeather	Fog	Hail	Rain	SandStorm	Snow
accuracy	0.94	0.90	0.87	0.78	0.96	0.91
precision	0.80	0.90	0.50	0.52	0.86	0.70
recall	0.53	0.67	0.55	0.62	0.92	0.80
F1_Score	0.64	0.77	0.52	0.56	0.89	0.75

V. Conclusion Générale

Pour mieux appréhender le problème, nous l'avons scindé en 3 chapitres dont le chapitre 1 pour entré dans l'univers de la problématique et comprendre le problème à résoudre (c'est-à-dire le cahier de charge et étude bibliographique).le second chapitre nous a permis de détaillé les différentes classes climatiques et leurs systèmes de détection ce qui nous a permis de trier et centré notre classification sur les plus critiques. Enfin un troisième grand chapitre nous a laissé la voix de bien parlé de la notion de classification d'image, du Deep Learning et ses algorithmes tels que les CNN.ces concepts de Deep Learning et ses algorithmes sont donc nos ressources pour élaborer notre système de classification automatique d'image dont son objectif est l'aide à la décision lors de la conduite des voitures d'ordre autonome et voire assisté aussi pendant les conditions climatiques dégradées. Ceci étant dit, afin de bien implémenter et programmer et former notre modèle il fallait des ressources supplémentaires du coté de langage de programmation et les librairies et API pour rendre possible le calcul assez conséquent de notre gros volume de données. Pour se faire, nous avons donc jugé nécessaire d'utiliser Google colab muni d'un GPU gratuit qui facilite les gros calculs.

Nous avons utilisé Tensorflow et Keras pour la classification et le dataset DAWN+Foggy+Snowy....

Bibliographies

- 1) Dataset, DAWN. <https://data.mendeley.com/datasets/766ygrbt8y/3>.
- 2) <https://www.tensorflow.org/tutorials/images/classification>
- 3) <https://keras.io/examples/>
- 4) The Impact of Adverse Weather Conditions on Autonomous Vehicles: How Rain, Snow, Fog, and Hail Affect the Performance of a Self-Driving Car.
<https://ieeexplore.ieee.org/document/8666747>