



**HAL**  
open science

# Analyse d'images par méthode de Deep Learning appliquée au contexte routier en conditions météorologiques dégradées

KhouLOUD Dahmane

► **To cite this version:**

KhouLOUD Dahmane. Analyse d'images par méthode de Deep Learning appliquée au contexte routier en conditions météorologiques dégradées. Vision par ordinateur et reconnaissance de formes [cs.CV]. Université Clermont Auvergne [2017-2020], 2020. Français. NNT : 2020CLFAC020 . tel-03022934

**HAL Id: tel-03022934**

**<https://theses.hal.science/tel-03022934>**

Submitted on 25 Nov 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



École doctorale : Sciences Pour l'Ingénieur

# THÈSE

pour obtenir le grade de :

## Docteur de l'Université Clermont Auvergne

### Spécialité : "Électronique et Systèmes"

*présentée et soutenue publiquement par*

**Khouloud DAHMANE**

le 17 juin 2020

## Analyse d'images par méthode de Deep Learning appliquée au contexte routier en conditions météorologiques dégradées

Directeur de thèse : **Frédéric CHAUSSE**

Co-encadrants de thèse : **Frédéric BERNARDIN, MICHÈLE COLOMB, CHRISTOPHE BLANC, NAJOUA ESSOUKRI BEN AMARA**

### Jury

|                                   |  |                    |
|-----------------------------------|--|--------------------|
| <b>Frédéric BERNARDIN,</b>        | Chercheur, Cerema                          | Co-encadrant       |
| <b>Christophe BLANC,</b>          | Maître de Conférences, UCA                 | Co-encadrant       |
| <b>Xavier CLADY,</b>              | Maître de Conférences, Sorbonne Université | Examinateur        |
| <b>Frédéric CHAUSSE,</b>          | Professeur, UCA                            | Directeur de thèse |
| <b>Michèle COLOMB,</b>            | Directeur de Recherche, Cerema             | Co-encadrante      |
| <b>Pierre DUTHON,</b>             | Ingénieur, Cerema                          | Invité             |
| <b>Najoua ESSOUKRI BEN AMARA,</b> | Professeur, Université de Sousse (Tunisie) | Président de jury  |
| <b>Nicolas HAUTIERE,</b>          | Chercheur, IFSTTAR                         | Rapporteur         |
| <b>Louhadi KHOUDOUR,</b>          | Directeur de Recherche, Cerema             | Rapporteur         |

---

---

## Résumé

### **Analyse d'images par méthode de Deep Learning appliquée au contexte routier en conditions météorologiques dégradées**

De nos jours, les systèmes de vision sont de plus en plus utilisés dans le contexte routier. Ils le 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.

Il existe des capteurs météorologiques dédiés à la mesure physique, mais ils sont coûteux. Ce problème peut être résolu en utilisant les caméras qui sont déjà installées sur les routes. En effet, ces dernières peuvent remplir simultanément deux fonctions : l'acquisition d'images pour les applications de surveillance et la mesure physique des conditions météorologiques au lieu des capteurs dédiés. C'est l'objectif des travaux de cette thèse : "Classification et mesure des conditions météorologiques avec caméras routières".

D'après l'état de l'art, il existe des méthodes de détection et de mesure de la visibilité du brouillard en utilisant une caméra fixe en conditions de jour. Il existe aussi d'autres travaux qui mesurent l'intensité des précipitations de pluie à l'aide de systèmes de vision artificielle en utilisant des méthodes de segmentation d'images.

Suite au grand succès des réseaux de neurones convolutifs (CNN) dans la classification et la reconnaissance d'images, nous avons utilisé une méthode d'apprentissage profond pour étudier le problème de la classification météorologique. Dans la littérature, le CNN a été utilisé pour classifier le temps en deux classes, par exemple, «nuageux» et «ensoleillé». Par ailleurs, d'autres travaux permettent la prédiction de la température ambiante, de la saison, du mois, de la semaine et du jour à partir d'une image donnée. L'objectif de notre étude est de chercher dans un premier temps à mettre au point un classifieur du temps, qui permet de discriminer entre temps « normal », brouillard et pluie. Dans un deuxième temps, une fois la classe connue, nous cherchons à développer un modèle de mesure de la distance de visibilité météorologique du brouillard.

Rappelons que l'utilisation des CNN exige l'utilisation de bases de données d'apprentissage et de test. Pour cela, deux bases de données ont été utilisées, "Cerema-AWP database", acquise en 2016 sur la plateforme PAVIN BP du Cerema (<https://ceremadlcfmds.wixsite.com/cerema-databases>), et la base "Cerema-AWH database", en cours d'acquisition depuis 2017 sur le site de la Fageole (Cantal) sur l'autoroute A75. Chaque image des deux bases est labellisée automatiquement grâce aux données météorologiques relevées sur le site permettant de caractériser diverses gammes de pluie et de brouillard.

La base Cerema-AWH, qui a été mise en place dans le cadre de nos travaux, contient cinq sous-bases : conditions normales de jour (JCN), brouillard fort (JB1), brouillard faible (JB2), pluie forte (JP1) et pluie faible (JP2). Les intensités de pluie varient de  $0 \text{ mm.h}^{-1}$  à  $70 \text{ mm.h}^{-1}$  et les visibilités météorologiques de brouillard varient entre  $50 \text{ m}$  et  $1800 \text{ m}$ . La sélection de ces images a été faite de telle sorte que les deux capteurs météorologiques utilisés pour labelliser les images (deux capteurs pour la pluie et deux pour le brouillard)

---

aient un écart relatif inférieur à 60 %.

Parmi les réseaux de neurones connus et qui ont montré leur performance dans le domaine de la reconnaissance et la classification, nous pouvons citer LeNet, ResNet-152, Inception-v4 et DenseNet-121. Nous avons appliqué ces réseaux dans notre système de détection et de classification des conditions météorologiques dégradées. Une analyse de ce système a été faite afin de comprendre l'influence de ses composantes sur le résultat final de la classification.

En premier lieu, une étude justificative de l'usage des réseaux de neurones convolutifs est effectuée. Elle étudie la nature de la donnée d'entrée et les hyper-paramètres optimaux qu'il faut utiliser pour aboutir aux meilleurs résultats. Ensuite, une analyse des différentes composantes d'un réseau de neurones est menée en construisant une architecture instrumentale de réseau de neurones. Les conclusions tirées de cette analyse nous ont conduit à utiliser certains réseaux de neurones profonds de l'état de l'art. La classification des conditions météorologiques avec ce type de réseaux a atteint un score de 83% pour une classification de cinq classes et 99% pour une classification de trois classes.

Ensuite, une analyse sur les données d'entrée et de sortie a été faite permettant d'étudier l'impact du changement de scènes et celui du nombre de données d'entrée et du nombre de classes météorologiques sur le résultat de classification.

Enfin, une méthode de transfert de bases de données a été appliquée. Cette méthode permet d'étudier la portabilité du système de classification des conditions météorologiques d'un site à un autre. Un score de classification de 63% a été obtenu en faisant un transfert entre une base publique et la base Cerema-AWH.

Après la phase de la classification, la deuxième étape de notre étude consiste à mesurer la visibilité météorologique du brouillard. Pour cela, nous utilisons un réseau de neurones qui génère des valeurs continues basées sur des images d'entrée. Ce réseau est similaire à un CNN utilisé pour la classification. Deux variantes de brouillard ont été testées : brouillard faible et brouillard fort combinés et brouillard fort (brouillard routier) tout seul. L'évaluation du résultat se fait à l'aide d'un coefficient de corrélation  $R^2$  entre les valeurs réelles et les valeurs prédites. Nous comparons ce coefficient avec le coefficient de corrélation  $R_{\text{capteurs}}^2$  entre les deux capteurs utilisés pour mesurer la visibilité météorologique sur site. Parmi les résultats obtenus et plus précisément pour le brouillard routier, le coefficient de corrélation atteint une valeur de 0,74 qui est proche de celle des capteurs physiques (0,76).

**Mots clés :** Systèmes de transport intelligents, Systèmes de vision, Vision par ordinateur, Intelligence artificielle, Traitement d'images, Conditions météorologiques dégradées, Brouillard, Pluie, Classification, Mesure, Bases de données météorologiques, Caméras routières, Réseaux de neurones convolutifs, Apprentissage profond.

---

## Abstract

Nowadays, vision systems are becoming more and more used in the road context. They are, for example, for video surveillance and ADAS (Advanced driver Assistance Systems). They ensure safety and facilitate mobility. These vision systems are generally affected by the degradation of weather conditions, for example, heavy fog or strong rain, phenomena limiting the visibility and thus reducing the quality of the images. In order to optimize the performance of the vision systems, it is necessary to have a reliable detection system for these adverse weather conditions.

There are meteorological sensors dedicated to physical measurement, but they are expensive. Since cameras are already installed on the road, they can simultaneously perform two functions : image acquisition for surveillance applications and physical measurement of weather conditions instead of dedicated sensors. This is the objective of the work : "Classification and measurement of meteorological conditions with road cameras".

According to the state of the art, there are methods for detecting and measuring the visibility of fog using a fixed camera in day conditions. There is also other work that measures the intensity of rainfall using artificial vision systems through segmentation methods.

Following the great success of convolutional neural networks (CNN) in classification and image recognition, we used a deep learning method to study the problem of meteorological classification. In the literature, CNN has been used to classify time into two classes, for example, "cloudy" and "sunny". In addition, other work allows the prediction of the ambient temperature, the season, the month, the week and the day from a given image based on the VGG16 architecture.

The objective of our study is to first seek to develop a classifier of time, which discriminates between "normal" conditions, fog and rain. In a second step, once the class is known, we seek to develop a model for measuring precipitation (meteorological visibility).

The use of CNN requires the use of train and test databases. For this, two databases were used, "Cerema-AWP database", acquired in 2016 on Cerema's Fog& Rain R& D platform (<https://ceremadlcfmnds.wixsite.com/cerema-databases>), and the "Cerema-AWH database", which has been acquired since 2017 on the Fageole site on the highway A75. Each image of the two bases is labeled automatically thanks to meteorological data collected on the site to characterize various levels of precipitation for rain and fog.

The Cerema-AWH base, which was set up as part of our work, contains 5 sub-bases : normal day conditions (JCN), heavy fog (JB1), light fog (JB2), heavy rain (JP1) and light rain (JP2). Rainfall intensities range from  $0 \text{ mm.h}^{-1}$  to  $70 \text{ mm.h}^{-1}$  and fog weather visibilities range from 50m to 1800m. The selection of these images was made in such a way that the two meteorological sensors used to label the images (two rain sensors and two fog sensors) had a relative difference of less than 60%.

Among the known neural networks that have demonstrated their performance in the field of recognition and classification, we can cite LeNet, ResNet-152, Inception-v4 and DenseNet-121. We have applied these networks in our adverse weather classification system. An analysis of this system was made in order to understand the influence of its different components on the final classification result.

We start by the study of the use of convolutional neural networks. The nature of the input data and the optimal hyper-parameters that must be used to achieve the best results. An analysis of the different components of a neural network is done by constructing

---

an instrumental neural network architecture. The conclusions drawn from this analysis show that we must use deep neural networks. This type of network is able to classify five meteorological classes of Cerema-AWH base with a classification score of 83% and three meteorological classes with a score of 99%

Then, an analysis of the input and output data was made to study the impact of scenes change, the input's data and the meteorological classes number on the classification result.

Finally, a database transfer method is developed. We study the portability from one site to another of our adverse weather conditions classification system. A classification score of 63% by making a transfer between a public database and Cerema-AWH database is obtained.

After the classification, the second step of our study is to measure the meteorological visibility of the fog. For this, we use a neural network that generates continuous values based on input images. This network is similar to the CNN used for classification. Two fog variants were tested : light and heavy fog combined and heavy fog (road fog) only. The evaluation of the result is done using a correlation coefficient  $R^2$  between the real values and the predicted values. We compare this coefficient with the correlation coefficient  $R^2_{\text{capteurs}}$  between the two sensors used to measure the weather visibility on site. Among the results obtained and more specifically for road fog, the correlation coefficient reaches a value of 0.74 which is close to the physical sensors value (0.76).

**Keywords :** Intelligent transport systems, Vision systems, Computer vision, Artificial intelligence, Image processing, Adverse weather conditions, Fog, Rain, Classification, Measurement, Meteorological databases, Road cameras, Convolutional neural networks, Deep learning.

---

## Remerciements

Arrivant à l'aboutissement de ma tâche, je me trouve dans l'obligation respectueuse de devoir présenter mes chaleureux remerciements et témoignage de ma gratitude à tous ceux qui ont contribué aimablement et avec patience à l'élaboration de ce mémoire.

Tout d'abord, je tiens à remercier profondément les membres du jury qui m'ont fait l'honneur de juger mon travail. Merci à Mme Najoua Essoukri Ben Amara, Mr Xavier Clady, Mr Nicolas Hautiere, et Mr Louhadi Khoudour. Ce jury est réellement composé d'experts des différents sujets abordés au cours de ma thèse, j'en suis très reconnaissante et profondément touchée.

Je souhaite aussi remercier très sincèrement les personnes m'ayant encadré tout au long de mon parcours de thèse pour leur générosité en matière d'encadrement, pour leurs judicieux conseils, leur soutien moral et leur aide durant l'élaboration de ce projet.

Merci à Michèle Colomb, responsable de l'unité Mobilité Durable et Sécurité, pour avoir partagé sa vision de la recherche qui nécessite une rigueur scientifique. Merci à Frédéric Bernardin, pour sa grande disponibilité et pour sa bienveillance au quotidien. Merci aussi pour les nombreuses relectures et corrections.

Merci à Frédéric Chausse, mon directeur de thèse, pour ses conseils précieux, son soutien et la confiance qu'il m'a accordé tout au long de ma thèse. Merci aussi pour m'avoir donné la chance de dispenser quelques travaux pratiques d'informatique, cet exercice fut très enrichissant.

Merci à Christophe Blanc, pour m'avoir encadré et dirigé sur la voie du succès avec ses recommandations.

Je tiens aussi à remercier l'ensemble de la direction du Département Laboratoire de Clermont-Ferrand (DLCF), pour m'avoir donné l'opportunité d'effectuer ma thèse au sein du laboratoire. Je remercie également Pierre Duthon pour ses judicieux conseils et son aide technique.

Je remercie aussi toute l'équipe du Cerema pour leur sympathie, leur accueil chaleureux et les moments de convivialités qu'on a partagé.

Je remercie aussi l'Institut Pascal, en tant que fidèle partenaire du Cerema sur de nombreux projets de recherche, mais aussi pour le financement d'une partie du matériel ayant permis de réaliser cette thèse. Je n'oublie pas non plus le Labex IMobS3 qui est un puissant moteur pour les collaborations clermontoises.

Les meilleurs pour la fin, je remercie chaleureusement mes très chers parents, Hichem et Najoua pour leurs encouragements, leur amour inconditionnel et leurs précieux conseils.

J'aurais bien aimé que vous soyez à côté de moi le jour de ma soutenance. Je remercie également ma sœur Sahar d'avoir été d'un grand soutien inébranlable.

---

Mes derniers remerciements vont à ma famille Clermontoise Eya, Rania, Khemakhem, Aida, Rania, Hedi et Malek pour leur soutien durant mes 3 ans, 8 mois et 17 jours de doctorat.

« Last but not least, I want to thank me. I want to thank me for believing in me. I want to thank me for doing all this work. I want to thank me for having no days off. I want to thank me for never quitting [...] I want to thank me for just being me at all time » *Calvin Cordozar Broadus Jr.*

**Dr. Khouloud Dahmane**

# Table des matières

|   |             |
|---|-------------|
| <b>Table des matières</b>   | <b>ix</b>   |
| <b>Liste des figures</b>  | <b>xi</b>   |
| <b>Liste des tableaux</b>   | <b>xiii</b> |
| <b>1 Introduction</b>   | <b>1</b>    |
| 1.1 Cadre de la thèse   | 2           |
| 1.2 Contexte de la thèse  | 4           |
| 1.3 Objectifs de la thèse   | 5           |
| 1.4 Problématique scientifique  | 6           |
| 1.5 Plan du mémoire   | 7           |
| <b>2 Revue bibliographique des algorithmes de classification et de mesure</b> | <b>9</b>    |
| 2.1 Introduction  | 10          |
| 2.2 Machine Learning et ses applications                                      | 10          |
| 2.3 Deep Learning vs Machine Learning   | 18          |
| 2.4 Application du Deep Learning au contexte routier                          | 20          |
| 2.5 Applications des réseaux de neurones de classification                    | 22          |
| 2.6 Algorithmes de classification des conditions météorologiques              | 25          |
| 2.7 Applications des réseaux de neurones de mesure                            | 27          |
| 2.8 Algorithmes de mesure des conditions météorologiques                      | 30          |
| 2.9 Synthèse  | 30          |
| <b>3 Les réseaux de neurones artificiels</b>                                  | <b>33</b>   |
| 3.1 Introduction  | 34          |
| 3.2 Historique du Deep Learning   | 34          |
| 3.3 Réseaux de neurones artificiels   | 36          |
| 3.4 Réseaux de neurones convolutifs   | 43          |
| 3.5 Réseaux de neurones utilisés  | 49          |
| <b>4 Description des bases de données météorologiques</b>                     | <b>57</b>   |
| 4.1 Introduction  | 58          |
| 4.2 Les conditions météorologiques dégradées                                  | 58          |
| 4.3 Les bases de données existantes et leurs limitations                      | 61          |
| 4.4 Création de la base de données  | 68          |
| 4.5 Synthèse  | 80          |

|          |  |            |
|----------|--|------------|
| <b>5</b> | <b>Classification des conditions météorologiques par caméra routière</b> | <b>81</b>  |
| 5.1      | Introduction . . . . .   | 82         |
| 5.2      | Justification de l'usage de réseaux de neurones convolutifs . . . . .    | 82         |
| 5.3      | Influence des données d'entrée et de sortie . . . . .                    | 92         |
| 5.4      | Transfert de bases de données . . . . .                                  | 97         |
| 5.5      | Mesure de la distance de visibilité météorologique . . . . .             | 103        |
| 5.6      | Synthèse . . . . .   | 106        |
| <b>6</b> | <b>Conclusion</b>  | <b>107</b> |
| 6.1      | Bilan général . . . . .  | 108        |
| 6.2      | Perspectives . . . . .   | 109        |

# Liste des figures

|      |   |    |
|------|---|----|
| 2.1  | Exemple d'une régression linéaire. . . . .  | 13 |
| 2.2  | Exemple de la fonction Sigmoidé. . . . .  | 14 |
| 2.3  | Classification avec la machine à vecteurs de support. . . . .   | 17 |
| 2.4  | Classification avec l'algorithme des k plus proches voisins. . . . .  | 18 |
| 3.1  | Modélisation d'un neurone artificiel. . . . .   | 36 |
| 3.2  | Exemple d'un réseau de neurones artificiel. . . . .   | 37 |
| 3.3  | Deux types de perceptrons. . . . .  | 38 |
| 3.4  | Schéma d'un réseau de neurones artificiel à $l - 1$ couches cachées. . . . .                                  | 39 |
| 3.5  | Exemples de fonctions d'activation. . . . .   | 40 |
| 3.6  | Différents types de convolution [72]. . . . .   | 45 |
| 3.7  | Cartes des caractéristiques. . . . .  | 47 |
| 3.8  | Exemple d'une opération de pooling de taille $2 \times 2$ . . . . .   | 48 |
| 3.9  | Architecture du réseau de neurones LeNet. . . . .   | 49 |
| 3.10 | Bloc convolutif résiduel (adapté de [11]). . . . .  | 50 |
| 3.11 | Architecture du réseau ResNet-34. . . . .   | 51 |
| 3.12 | Modules d'inception du réseau de neurones Inception-v4 [218]. . . . .   | 53 |
| 3.13 | Architecture du réseau Inception-v4. . . . .  | 53 |
| 3.14 | Bloc convolutif dense (adapté de [11]). . . . .   | 54 |
| 3.15 | Architecture du réseau DenseNet-121. . . . .  | 55 |
| 4.1  | La Tour Eiffel sous différentes conditions météorologiques [45]. . . . .                                      | 62 |
| 4.2  | Exemple des conditions météorologiques présentes dans la base Cerema-AWP. . . . .                             | 64 |
| 4.3  | Exemples d'images de la base MWD [153]. . . . .   | 65 |
| 4.4  | Exemples d'images de la base Image2Weather. . . . .   | 66 |
| 4.5  | Exemples d'images présentant des ambiguïtés. . . . .  | 67 |
| 4.6  | Le site autoroutier du col de la Fageole. . . . .   | 68 |
| 4.7  | Capteurs de mesure de l'intensité de pluie. . . . .   | 69 |
| 4.8  | Capteurs de mesure de la visibilité météorologique. . . . .   | 69 |
| 4.9  | Interface graphique du logiciel d'acquisition de la base météorologique d'images. . . . .                     | 70 |
| 4.10 | Principe du fonctionnement du logiciel de déclenchement automatique de l'acquisition des images [74]. . . . . | 70 |
| 4.11 | Interface graphique du système d'édition des scénarios de déclenchement de l'acquisition. . . . .             | 71 |
| 4.12 | Principe de fonctionnement du logiciel de gestion des données de la base [74]. . . . .                        | 73 |

|      |   |     |
|------|---|-----|
| 4.13 | Corrélation des capteurs mesurant la distance de visibilité météorologique du brouillard. . . . .               | 75  |
| 4.14 | Corrélation des capteurs mesurant l'intensité de la pluie. . . . .  | 75  |
| 4.15 | Exemples d'images issues de la base de données Cerema-AWH. . . . .  | 76  |
| 4.16 | La série chronologique des distances de visibilité météorologique présentes dans la base Cerema-AWH. . . . .    | 77  |
| 4.17 | Distribution de la distance de visibilité météorologique sur la base Cerema-AWH. . . . .                        | 78  |
| 4.18 | La série chronologique des intensités de pluie présentes dans la base Cerema-AWH. . . . .                       | 79  |
| 4.19 | Distribution de l'intensité de pluie sur la base Cerema-AWH. . . . .  | 79  |
| 5.1  | Différents types de patchs de la base Cerema-AWP. . . . .   | 84  |
| 5.2  | Résultats d'apprentissage avec différents couples d'hyper-paramètres sur la base de données Cerema-AWH. . . . . | 86  |
| 5.3  | Architecture du réseau ClariNet1. . . . .   | 87  |
| 5.4  | Architecture du réseau ClariNet2. . . . .   | 88  |
| 5.5  | Impact du nombre de neurones et du nombre des filtres sur le score de justesse avec ClariNet1. . . . .          | 88  |
| 5.6  | Impact du nombre de neurones et du nombre des filtres sur le score de justesse avec ClariNet2. . . . .          | 89  |
| 5.7  | Comparaison des architectures des DNNs. . . . .   | 91  |
| 5.8  | Exemple d'une zone fixe extraite du centre d'une image de la base Cerema-AWH. . . . .                           | 93  |
| 5.9  | Exemple de deux patchs similaires des bases Cerema-AWP et Cerema-AWH. . . . .                                   | 98  |
| 5.10 | Exemple de deux images de pluie des bases Cerema-AWH et Cerema-AWP. . . . .                                     | 99  |
| 5.11 | Architecture du réseau de neurones de mesure ClariNet-M. . . . .  | 104 |
| 5.12 | Courbe de régression linéaire entre la visibilité prédite et la visibilité terrain pour JB1+JB2. . . . .        | 105 |
| 5.13 | Courbe de régression linéaire entre la visibilité prédite et la visibilité terrain pour JB1. . . . .            | 105 |
| 5.14 | Courbe de régression linéaire des deux capteurs de visibilité pour JB1. . . . .                                 | 106 |

# Liste des tableaux

|      |   |     |
|------|---|-----|
| 3.1  | Différentes architectures de ResNet (adapté de [105]). . . . .  | 51  |
| 3.2  | Différentes architectures de DenseNet (adapté de [115]). . . . .  | 55  |
| 4.1  | Types des précipitations en fonction de leur diamètre et leur vitesse de chute.   | 59  |
| 4.2  | Différentes classes de pluie selon la norme NF P 99-320 [5]. . . . .  | 59  |
| 4.3  | Différents types et classes de brouillard selon la norme NF P 99-320 [5]. . . .   | 61  |
| 4.4  | Caractéristiques des bases de données existantes. . . . .   | 63  |
| 4.5  | Caractéristiques des capteurs de mesure de l'intensité de pluie. . . . .  | 68  |
| 4.6  | Caractéristiques des capteurs de mesure de la visibilité météorologique. . .  | 69  |
| 4.7  | Caractéristiques de la base de données Cerema-AWH. . . . .  | 77  |
| 4.8  | Caractéristiques de la base de données Cerema-AWH-M. . . . .  | 78  |
| 5.1  | Résultats détaillés sur la prédiction avec des patches uniformes. . . . .   | 85  |
| 5.2  | Résultats détaillés sur la prédiction avec des patches contrastés. . . . .  | 85  |
| 5.3  | Hyper-paramètres utilisés pour l'apprentissage des réseaux de neurones. . .   | 86  |
| 5.4  | Étude comparative des architectures de l'état de l'art. . . . .   | 92  |
| 5.5  | Résultats de classification des conditions météorologiques en 5 classes. . . .  | 92  |
| 5.6  | Résultats du changement de scènes sur la base Cerema-AWH avec 5 classes<br>météorologiques. . . . .                                     | 94  |
| 5.7  | Résultats de classification des conditions météorologiques en 5 classes en<br>changeant le nombre des données d'apprentissage . . . . . | 94  |
| 5.8  | Résultats de classification des conditions météorologiques en 5 classes en<br>changeant le nombre des données de test. . . . .          | 95  |
| 5.9  | Résultats de classification des conditions météorologiques en 3 et 5 classes.   | 96  |
| 5.10 | Comparaison de la classification sur 3 bases différentes. . . . .   | 97  |
| 5.11 | Résultats détaillés du transfert de bases entre Cerema-AWP et Cerema-AWH.   | 98  |
| 5.12 | Résultats de classification du transfert de bases entre Cerema-AWH et les<br>bases publiques avec 3 classes météorologiques. . . . .    | 99  |
| 5.13 | Résultats de classification du transfert de bases entre les bases publiques et<br>Cerema-AWH avec 3 classes météorologiques. . . . .    | 100 |
| 5.14 | Nombre d'images pour chaque variante de base mélangée. . . . .  | 100 |
| 5.15 | Comparaison de la classification sur des bases mélangées. . . . .   | 101 |
| 5.16 | Nombre d'images pour chaque variante des bases AWH+I2W et AWH+MWD.  | 101 |
| 5.17 | Résultats de classification obtenus avec les différentes variantes des bases<br>AWH+I2W et AWH+MWD. . . . .                             | 102 |
| 5.18 | Résultats détaillés sur la prédiction avec la base mélangée MWD+I2W. . . .  | 102 |

|   |     |
|---|-----|
| 5.19 Résultats de classification du transfert entre les bases mélangées avec 3 classes météorologiques. . . . . | 103 |
| 5.20 Résultats de la mesure de la distance de visibilité météorologique sur la base Cerema-AWH-M. . . . .       | 104 |

# Chapitre 1

## Introduction

### Sommaire

---

|   |          |
|---|----------|
| <b>1.1 Cadre de la thèse</b> . . . . .  | <b>2</b> |
| 1.1.1 Laboratoire d'Excellence <i>Innovative Mobility : Smart and Sustainable Solutions</i> . . . . . | 2        |
| 1.1.2 Cerema . . . . .  | 3        |
| 1.1.3 Institut Pascal . . . . .   | 3        |
| 1.1.4 LATIS . . . . .   | 4        |
| <b>1.2 Contexte de la thèse</b> . . . . .   | <b>4</b> |
| <b>1.3 Objectifs de la thèse</b> . . . . .  | <b>5</b> |
| <b>1.4 Problématique scientifique</b> . . . . .   | <b>6</b> |
| <b>1.5 Plan du mémoire</b> . . . . .  | <b>7</b> |

---

## 1.1 Cadre de la thèse

Le travail de thèse présenté ci-après et intitulé "Analyse d'images par méthode de Deep Learning appliquée au contexte routier en conditions météorologiques dégradées" s'inscrit dans un contexte routier et œuvre pour la mobilité innovante. Celui-ci fait appel à l'intelligence artificielle ainsi qu'aux systèmes de vision afin d'améliorer la sécurité routière et la mobilité.

La mobilité innovante des personnes, des biens et des machines est l'une des ambitions du Laboratoire d'Excellence IMobS3 (*Innovative Mobility : Smart and Sustainable Solutions*). Il s'agit d'un centre international de référence pour la mobilité soutenu par les Investissements d'Avenir, des Fonds Européens de Développement Régional (FEDER) et de la Région Auvergne Rhône-Alpes. Le LabEx est porté par cinq établissements dont l'Université Clermont Auvergne et le Cerema et il est constitué de plusieurs laboratoires dont l'Institut Pascal.

C'est dans cet environnement scientifique que s'inscrivent les travaux de thèse en ayant le Cerema comme établissement d'accueil, avec un co-encadrement dans l'équipe de recherche Systèmes de Transport Intelligents (STI) [2]. La direction de thèse ainsi qu'un co-encadrement sont assurés par l'Institut Pascal dans l'équipe ComSee. Ce doctorat fait l'objet d'une collaboration internationale avec le laboratoire LaTiS de l'université de Sousse en Tunisie. Ces différents établissements sont présentés dans la suite.

### 1.1.1 Laboratoire d'Excellence *Innovative Mobility : Smart and Sustainable Solutions*

Le Laboratoire d'Excellence IMobS3 (*Innovative Mobility : Smart and Sustainable Solutions*) a pour ambition de développer des briques technologiques efficaces et respectueuses de l'environnement pour une mobilité innovante des personnes, des biens et des machines en jouant sur le triptyque "Recherche – Formation – Valorisation".

Le LabEx se base sur trois défis :

- **Véhicules et Machines Intelligents** : marchés de niche à très forte valeur ajoutée avec les systèmes d'aides à la conduite, les véhicules autonomes, la robotique manufacturière, la robotique agricole, les machines outils agiles et reconfigurables et les ateliers flexibles.

- **Services et Systèmes de Mobilité Intelligente** : systèmes d'aide à la décision novateurs permettant la gestion optimisée d'une flotte de véhicules ou d'un ensemble de machines (ateliers flexibles, bioréacteur) au plan de la qualité du service rendu, de la sécurité de l'utilisateur et de l'énergie consommée.

- **Procédés de production d'énergie pour la mobilité** : mise au point de processus innovants visant à l'intégration et l'intensification des bioréacteurs pour la genèse d'un concept de bioraffinerie d'avenir, à partir de la fixation du carbone par les microalgues (nouveaux capteurs, interaction lumière-matière optimisée, contrôle-commande des processus...).

C'est dans le défi des véhicules et machines intelligents que s'inscrit la thèse puisqu'elle porte sur l'analyse des systèmes de transport intelligents et leur fonctionnement en conditions météorologiques dégradées.

### 1.1.2 Cerema

Le Cerema est un établissement public à caractère administratif (EPA), sous la tutelle conjointe du Ministère de la Transition écologique et solidaire et du Ministère de la Cohésion des territoires et des Relations avec les collectivités territoriales. Le Cerema développe des relations étroites avec les collectivités territoriales qui sont présentes dans ses instances de gouvernance. Il répond ainsi au besoin de disposer d'un appui scientifique et technique renforcé, pour élaborer, mettre en œuvre et évaluer les politiques publiques de l'aménagement et du développement durables. Une des spécificités du Cerema est que son champ d'action va de l'expertise terrain jusqu'à la recherche appliquée.

Cette thèse s'inscrit dans le champ d'action mobilité et transport. Cependant, le Cerema présente d'autres champs tels que :

- aménagement et développement des territoires
- villes et stratégies urbaines
- transition énergétiques et changement climatique
- gestion des ressources naturelles et respect de l'environnement
- prévention des risques
- bien-être et réduction des nuisances
- gestion, optimisation, modélisation et conception des infrastructures
- habitat et bâtiment.

Acteur neutre et impartial, il développe des méthodes et outils opérationnels pour répondre, à toutes les échelles territoriales, aux besoins des collectivités locales ainsi qu'à ceux des acteurs économiques ou associatifs. Quelle que soit la taille de leurs projets, il leur propose un large panel de prestations : expertise et ingénierie, évaluation, méthodologie, essais et contrôle, recherche et développement, certification, normalisation.

La thèse se déroule au sein de l'unité Mobilité Durable et Sécurité du Département Laboratoire de Clermont-Ferrand, Cerema Centre-Est. Cette unité de recherche se focalise sur les domaines de la mobilité et de la sécurité routière en abordant les thématiques suivantes : modélisation et simulation des conditions météorologiques dégradées, technologies du véhicule et de l'infrastructure, perception et comportement humain, accidentologie, signalisations. Ainsi, dans le cadre de ces travaux, le Cerema a pu apporter toute son expertise sur les conditions météorologiques dégradées mais aussi la définition des éléments concernant le contexte routier. L'équipe du Cerema a aussi mis en œuvre tout le système d'acquisition des données et assure la gestion des bases de données utilisées dans ces travaux.

### 1.1.3 Institut Pascal

Placé sous la tutelle de l'Université Clermont Auvergne, du CNRS (institut principal INSIS, instituts secondaires INS2I, INP) et de SIGMA Clermont, l'Institut Pascal est un laboratoire de recherche interdisciplinaire s'inscrivant dans des domaines stratégiques des Sciences de l'Ingénierie et des Systèmes : Génie des Procédés, Mécanique, Robotique, Physique des Sciences de l'Information, Santé.

L'Institut Pascal développe des systèmes innovants et intelligents par une approche systémique et multi-échelle, s'appuyant sur des champs scientifiques identifiés : électromagnétisme, fiabilité, génie des procédés et des bioprocédés, génie civil, imagerie, matériaux, mécanique, microsystèmes, nanophysique, perception, photonique, robotique, séparation et fonctionnalisation de biomolécules, technologies interventionnelles et diagnostiques médicales.

L'Institut Pascal comporte cinq axes de recherche :

- Génie des Procédés, Energétique et Biosystèmes.
- Image, Systèmes de Perception, Robotique.
- Mécanique, Génie Mécanique, Génie Civil, Génie Industriel.
- Photonique, Ondes, Nanomatériaux.
- Thérapies Guidées par l'Image.

La thèse se situe plus particulièrement dans l'axe Image, Systèmes de Perception, Robotique (ISPR) au sein de l'équipe ComSee (*Computers that See*). Cette équipe s'intéresse à deux grands thèmes qui sont la vision géométrique (modélisation et l'étalonnage de capteurs de vision, localisation et la reconstruction 3D par vision) et l'interprétation de séquences d'images (détecter, reconnaître, suivre les différents éléments d'une scène dans des séquences dynamiques, complexes et parfois dans des conditions de visibilité dégradées). Les domaines d'application des travaux de recherche vont des transports intelligents, à la réalité augmentée, en passant par la vidéo-surveillance et la vision pour la robotique. Dans le cadre de la thèse, l'Institut Pascal apporte donc tous les aspects académiques concernant les algorithmes de vision artificielle, mais aussi un sens pratique des problèmes rencontrés par ces algorithmes en conditions dégradées.

#### 1.1.4 LATiS

Le laboratoire de recherche LATIS - *Laboratory of Advanced Technology and Intelligent Systems* - est une structure de recherche de l'Ecole Nationale d'Ingénieurs de Sousse, université de Sousse. Le LATIS développe une recherche pluridisciplinaire dans le domaine large du génie électrique et de la technologie de l'information.

Ses activités de recherche s'articulent autour de quatre thèmes principaux qui sont :

- Vision par ordinateur & Intelligence artificielle
- Analyse et traitement de documents
- Compatibilité électromagnétique et Sécurité de Fonctionnement
- Energies renouvelables, Smart Grid et Commande des machines

## 1.2 Contexte de la thèse

De nombreux accidents routiers dus à différents facteurs, comme la vitesse excessive, ont lieu tous les ans. Les accidents de la route sont classés comme la 9<sup>ème</sup> cause majeure de décès dans le monde. Selon l'Observatoire national interministériel de la sécurité routière (ONISR), 3 488 personnes ont perdu la vie sur les routes de France en 2018 [239]. Cette perte en vies humaines ainsi que l'augmentation de la circulation ont conduit au développement

des Systèmes de Transport Intelligents (STI). Avec l'amélioration de la recherche dans le domaine de la vision par ordinateur, les réseaux routiers sont devenus de plus en plus intelligents et communicants. Les solutions offertes par les STI reposent sur une perception et une analyse de l'environnement [30]. L'étape de la perception est basée sur l'utilisation de divers systèmes de vision par ordinateur qui sont constitués généralement d'un capteur de vision et d'un algorithme d'analyse. Parmi ces systèmes, nous trouvons, dans certains cas, la vidéo surveillance. Ce sont des caméras qui sont installées au bord de la route et qui permettent de contrôler les flux de circulation comme la présence des congestions ou même la détection d'incidents comme les accidents. Il existe aussi les systèmes d'aide à la conduite avancée – ou *Advanced Driver Assistance System* – (ADAS) qui permettent au conducteur de bénéficier d'une assistance dans certaines conditions de circulation. Comme la vidéo surveillance, les ADAS actuels en développement utilisent les caméras qui leur permettent la détection de piétons, la détection des obstacles sur la route, la détection des marquages au sol, etc.

### 1.3 Objectifs de la thèse

Assurer la sécurité par les systèmes de vision n'est plus garanti en présence de conditions météorologiques dégradées. Ce problème de sécurité et de protection dans des conditions défavorables était le centre d'intérêt de nombreuses études qui ont été réalisées pour mettre en évidence la vulnérabilité du fonctionnement des services de transport en raison des conditions météorologiques [7]. Dans des études précédentes, des vitesses et des comportements de conduite contrôlés par les conditions météorologiques avaient été proposés [193]. En effet, un abaissement de la vitesse limite a été imposé par le code de la route dans le cas d'une visibilité limitée suite à un brouillard dense ou une pluie forte. Par exemple en France, pour une distance de visibilité inférieure à 50 m, une vitesse maximale de 50 kilomètres par heure est autorisée [103]. Outre le problème de la sécurité routière, la visibilité réduite entraîne des retards et des perturbations dans le réseau routier. Sur les autoroutes, les entassements massifs créent des embouteillages non récurrents qui obligent parfois la fermeture momentanée de la route. A noter aussi, même si ce n'est pas notre sujet, qu'une mesure de visibilité pourrait également être intéressante pour les aspects sanitaires. En effet, une visibilité réduite est considérée comme un symptôme de problèmes environnementaux car elle est un signe de pollution atmosphérique [118]. En outre, il a été démontré que la visibilité réduite en milieu urbain et la mortalité sont corrélées [224]. Selon Thach et al. [224], la visibilité fournit un indicateur utile pour l'évaluation des risques de santé liés à l'environnement et une approche valable pour évaluer les impacts de la pollution atmosphérique sur la santé publique. La pluie est également une condition météorologique critique, qui représente 13% des accidents en France [184]. Ces conditions critiques exigent donc des solutions particulières afin d'améliorer la sécurité routière.

Les systèmes de vision qui sont utilisés dans le contexte routier ne sont pas évalués dans des conditions météorologiques dégradées. En effet, dans ces conditions, le fonctionnement des systèmes de transport est dégradé. Les systèmes actuels reposent soit sur une série de capteurs coûteux (par exemple des capteurs de détection d'objets), soit sur une assistance humaine pour identifier les conditions météorologiques [42, 182, 203]. Ces

solutions entraînent des coûts supplémentaires et ont une portée limitée. Pour assurer le bon fonctionnement de tous les services de transport dans toutes les conditions météorologiques, un système de détection fiable est nécessaire pour classer les conditions météorologiques. Afin de résoudre ce problème, il faut prendre en compte d'une part, la diversité de la nature des conditions météorologiques, et d'autre part, l'inexistence de caractéristiques discriminantes entre les diverses conditions météorologiques. Ces dernières années, d'importantes contributions ont été apportées pour tenter de résoudre le problème de la classification météorologique. Un ensemble de ces tentatives a ciblé le problème du point de vue de la classification météorologique pour des fins de circulation en se limitant à une seule condition défavorable telle que la pluie [53?, 137, 196] ou le brouillard [17, 188, 187].

Aujourd'hui, il existe des stations météorologiques sur quelques sites du territoire (environ 554 stations). Ces stations contiennent des capteurs météorologiques dédiés à la mesure physique. Cependant, ces derniers restent coûteux et demandent une maintenance aux bords de la route. Puisque les caméras routières sont déjà installées et réparties sur tout le territoire pour assurer la surveillance du trafic et la détection des incidents, il serait plus simple de modifier leur logiciel d'analyse d'images en introduisant de nouvelles fonctionnalités pour mesurer les conditions météorologiques dégradées. Cela permettrait de faire des économies liées à la maintenance et à l'installation. Pour cela, nous nous sommes intéressés au développement d'un système de détection et de mesure des phénomènes climatiques en se basant sur des caméras routières. En effet, ce système se base sur une analyse automatique des images acquises par des caméras installées en bords des voies et en utilisant une approche d'apprentissage profond par des réseaux de neurones convolutifs.

## 1.4 Problématique scientifique

Parmi les travaux de l'état de l'art, il en existe quelque uns qui s'intéressent à l'influence des conditions météorologiques sur les descripteurs d'images [74]. Ils se basent sur des méthodes traditionnelles d'apprentissage automatique afin d'extraire les descripteurs d'images manuellement. Cette méthode d'extraction ne garantit pas l'obtention de caractéristiques pertinentes permettant d'avoir des systèmes de classification météorologiques fiables et efficaces. Pour résoudre ce problème, il est nécessaire d'extraire d'une manière automatique des descripteurs discriminants qui participent à l'optimisation de ce type de systèmes de vision et de les rendre robustes face aux changements climatiques.

La classification des conditions météorologiques dégradées nécessite un grand nombre de données d'images. Cependant, les bases de données météorologiques utilisées à cet effet ne se basent pas sur des capteurs dédiés à la mesure physique des conditions atmosphériques, mais plutôt sur des images collectées à partir de différents sites et sur une labellisation manuelle. Ainsi, la fiabilité de ces données peut-elle influencer sur la performance des systèmes de classification météorologiques? Pour répondre à cette question, nous comparons une sélection de bases de données publiques à notre base de données qui se base dans sa labellisation sur des capteurs météorologiques.

L'utilisation de la méthode de Deep Learning se base sur l'apprentissage des réseaux de neurones profonds, ces réseaux possèdent des architectures différentes les unes des autres.

Elles sont constituées par des couches de neurones différentes selon le nombre et le type (couches de convolution, couches de pooling, etc). Mais, est-ce que ces couches ont un impact sur l'apprentissage? Et en quoi consiste l'importance de chacune d'elles? Afin de répondre à ces deux questions, nous analysons l'impact de chaque type de ces couches sur les résultats de classification des conditions météorologiques dégradées.

L'application de la classification des conditions météorologiques dégradées en contexte routier dépend de l'environnement dans lequel sont installées les caméras routières. Dans ce cas, il est nécessaire d'avoir un système de vision unique qui sera valable pour tous les sites tout en gardant une meilleure performance. Dans ce cas, un transfert de bases serait-il une bonne méthode pour faire l'apprentissage de ces systèmes? Pour répondre à cette question, nous n'utilisons pas qu'une base de données en faisant l'apprentissage du réseau de neurones sur une base et le test sur une autre, tout en analysons l'ensemble des résultats obtenus.

## 1.5 Plan du mémoire

Les approches proposées dans cette thèse sont réparties comme suit : ces travaux se basent sur l'apprentissage profond par des réseaux de neurones convolutifs vu leurs succès à résoudre de nombreuses tâches y compris la classification. Mais tout d'abord, il est nécessaire de présenter les différentes méthodes de Machine Learning tout en comparant leurs performances avec celles du Deep Learning. Cela est présenté dans le chapitre 2. Ce chapitre porte aussi sur une revue bibliographique des applications en contexte routier développées en utilisant le Deep Learning ainsi que les applications basées sur les réseaux de neurones de classification et de mesure. Il décrit aussi les différents algorithmes de classification et de mesure des conditions météorologiques de l'état de l'art.

Le chapitre 3 présente, au début, un rapide historique du Deep Learning. Ensuite, une description détaillée du fonctionnement des réseaux de neurones artificiels et convolutifs est présentée. Et enfin, ont décrit les réseaux de neurones convolutifs qui seront utilisés dans la suite de nos travaux.

La création d'un système de classification des conditions météorologiques dégradées demande un nombre important d'images acquises sous différents phénomènes atmosphériques comme le brouillard et la pluie. Aussi, elle exige la disponibilité des informations météorologiques bien précises indiquant la présence des conditions défavorables associées à leurs intensités. Ainsi, le chapitre 4 présente la base de données météorologiques d'images créée à partir d'un site réel extérieur instrumenté. Cette base a été développée après avoir identifié les limitations des bases de données déjà existantes. Ce chapitre comporte aussi une description détaillée des différentes bases de données publiques qui vont être utilisées tout au long de cette thèse.

Le chapitre 5 porte sur la présentation de l'approche utilisée pour la classification et la mesure des conditions météorologiques dégradées. Pour la phase de la classification, une étude académique a été effectuée initialement pour justifier l'utilisation des réseaux de neurones convolutifs en étudiant les différents éléments qui les construisent et qui sont nécessaires pour leur apprentissage. Ensuite, une étude sur les données d'entrée et de sortie du système est présentée. En effet, ces différents éléments sont susceptibles d'influencer le résultat de la détection des conditions dégradées. Enfin, une étude de

la méthode de transfert de bases a été effectuée. Elle permet d'étudier la portabilité du système de classification d'un site météorologique à un autre. La dernière section de ce chapitre concerne la deuxième partie de notre approche qui est la mesure de la distance de visibilité météorologique dans le cas où le système de classification météorologique a détecté le brouillard.

Finalement, le chapitre 6 présente le bilan général des travaux de cette thèse, ainsi qu'un ensemble de perspectives qui pourront être explorées à la suite de ce travail de recherche.

# Chapitre 2

## Revue bibliographique des algorithmes de classification et de mesure

### Sommaire

---

|   |           |
|---|-----------|
| <b>2.1 Introduction</b>   | <b>10</b> |
| <b>2.2 Machine Learning et ses applications</b>                         | <b>10</b> |
| 2.2.1 Régression linéaire   | 11        |
| 2.2.2 Régression logistique   | 13        |
| 2.2.3 Arbre de décision   | 14        |
| 2.2.4 Machines à vecteurs de support                                    | 15        |
| 2.2.5 K-plus proches voisins  | 17        |
| <b>2.3 Deep Learning vs Machine Learning</b>                            | <b>18</b> |
| <b>2.4 Application du Deep Learning au contexte routier</b>             | <b>20</b> |
| <b>2.5 Applications des réseaux de neurones de classification</b>       | <b>22</b> |
| <b>2.6 Algorithmes de classification des conditions météorologiques</b> | <b>25</b> |
| <b>2.7 Applications des réseaux de neurones de mesure</b>               | <b>27</b> |
| <b>2.8 Algorithmes de mesure des conditions météorologiques</b>         | <b>30</b> |
| <b>2.9 Synthèse</b>   | <b>30</b> |

---

## 2.1 Introduction

L'intelligence artificielle (IA) est en plein développement ces dernières années. Comprendre les dernières avancées dans ce domaine revient à étudier deux concepts très populaires tels que l'apprentissage automatique ou *Machine Learning* (ML) et l'apprentissage profond ou *Deep Learning* (DL).

Dans ce chapitre, nous allons présenter l'apprentissage automatique et ses algorithmes ainsi que ses applications dans différents domaines (section 2.2). Une comparaison entre le *Machine Learning* et le *Deep Learning* sera présentée tout en montrant que l'apprentissage profond gagne en popularité en raison de ses performances dans différentes tâches (section 2.3). La section 2.4 présente les différentes applications du *Deep Learning* dans le contexte routier afin de montrer ses différents usages dans ce domaine. Dans les sections 2.5 et 2.6, nous nous concentrons sur les applications des réseaux de neurones dans la classification et la mesure respectivement.

## 2.2 Machine Learning et ses applications

Le Machine Learning est une méthode employée dans l'intelligence artificielle. C'est une méthode d'apprentissage statistique où chaque instance dans une base de données est décrite par un ensemble de caractéristiques ou d'attributs.

L'idée générale est d'être capable de reproduire une relation grâce à une fonction  $f$ . Cette fonction est identifiée par un algorithme d'apprentissage. Pour aboutir à la décision ou à la prédiction  $Y$ , l'algorithme utilise en entrée un jeu de données  $X$  afin de déterminer un modèle de classification (2.1).

$$Y = f(X) \tag{2.1}$$

Il existe différents types d'algorithmes à utiliser chacun selon la tâche à accomplir et selon le type de données disponibles. Ces algorithmes se divisent en trois catégories : supervisés, non supervisés et par renforcement.

Les algorithmes d'apprentissage non supervisé utilisent un ensemble de données contenant de nombreuses caractéristiques, puis apprennent des propriétés utiles de la structure de cet ensemble de données. La majorité des algorithmes d'apprentissage non supervisés se basent sur la méthode de clustering qui consiste à diviser le jeu de données en groupes d'exemples similaires.

Les algorithmes d'apprentissage supervisé utilisent un ensemble de données contenant des caractéristiques, mais chaque exemple est également associé à une étiquette ou à une cible. Par exemple, le jeu de données Iris de Fisher [25] est étiqueté avec les espèces de chaque plante d'iris. Un algorithme d'apprentissage supervisé peut étudier le jeu de données sur l'iris et apprendre à classer les plantes à l'iris en trois espèces différentes (Iris setosa, Iris virginica, Iris versicolor) en fonction de leurs mesures.

Les données d'apprentissage pour l'apprentissage supervisé comprennent un couple d'informations, l'échantillon d'entrée et la sortie désirée. Cette sortie peut être une valeur discrète/catégorique ou réelle. Par exemple, les modèles de régression estiment les sorties à valeur réelle, tandis que les modèles de classification estiment les sorties à valeur discrète. Les modèles de classification binaire simples n'ont que deux labels de sortie, 1 (positive) et

0 (négative). Pour le traitement d'images, par exemple, un système d'intelligence artificielle peut être doté d'images étiquetées de véhicules appartenant à des catégories différentes telles que les voitures et les camions. Après un nombre d'observations suffisant, le système devrait être capable de distinguer et de classer par catégories les images non étiquetées; c'est le moment auquel l'apprentissage peut être considéré comme complet.

En général, l'apprentissage non supervisé implique l'observation de plusieurs exemples d'un vecteur aléatoire  $x$  et tente implicitement ou explicitement d'apprendre la distribution de probabilité  $p(x)$ , ou certaines propriétés intéressantes de cette distribution. Tandis que l'apprentissage supervisé consiste à observer plusieurs exemples d'un vecteur aléatoire ainsi que la valeur associée  $y$ , et apprendre à prédire  $y$  à partir de  $x$ , généralement en estimant  $p(y|x)$ . Le terme apprentissage supervisé provient de la vue de la cible  $y$  fournie par un instructeur ou un superviseur qui guide le système d'apprentissage automatique (d'où le nom apprentissage supervisé). Dans l'apprentissage non supervisé, il n'y a pas d'instructeur ni de superviseur et l'algorithme doit apprendre à donner un sens aux données sans ce guide.

L'apprentissage non supervisé et l'apprentissage supervisé ne sont pas des termes formellement définis. De nombreuses technologies d'apprentissage automatique peuvent être utilisées pour effectuer les deux tâches. Par exemple, la règle de probabilité indique que, pour un vecteur  $x \in \mathbb{R}^n$ , la distribution conjointe peut être décomposée comme suit :

$$p(x) = \prod_{i=1}^n p(x_i|x_1, \dots, x_{i-1}) \quad (2.2)$$

Cette décomposition signifie que nous pouvons résoudre le problème non supervisé de la modélisation de  $p(x)$  en le divisant en  $n$  problèmes d'apprentissage supervisé. Alternativement, en utilisant (2.3) nous pouvons résoudre le problème de l'apprentissage supervisé de  $p(y|x)$  en nous basant sur les techniques d'apprentissage non supervisées traditionnelles qui apprennent la distribution conjointe  $p(x, y)$ .

$$p(y|x) = \frac{p(x, y)}{\sum_{y'} p(x, y')} \quad (2.3)$$

Parmi les algorithmes populaires d'apprentissage supervisé il y a la régression linéaire ou *linear regression*, la régression logistique ou *logistic regression*, l'arbre de décision ou *decision tree*, les machines à vecteurs de support ou *support vector machine*, les réseaux de neurones ou *neural network*, ainsi que les modèles non paramétriques tels que les K-plus proches voisins ou *K-neighbors*.

### 2.2.1 Régression linéaire

La régression linéaire est utile pour trouver une relation entre deux variables continues. L'un est une variable indépendante et l'autre est une variable dépendante. La régression linéaire cherche une relation statistique, mais pas une relation déterministe. La relation entre deux variables est dite déterministe si une variable peut être exprimée avec précision par l'autre. Par exemple, en utilisant la température en degrés Celsius, il est possible de prédire avec précision la température en Fahrenheit. La relation statistique n'est pas précise pour déterminer la relation entre deux variables. Par exemple, relation entre taille et poids dans une population.

L'idée principale est d'obtenir une droite qui correspond le mieux aux données. La droite la mieux adaptée est celle pour laquelle l'erreur de prédiction totale (tous les points de données) est la plus petite possible. L'erreur est la distance entre le point et la ligne de régression.

Comme son nom l'indique, la régression linéaire résout un problème de régression. Autrement dit, l'objectif est de construire un système pouvant prendre en entrée un vecteur  $x \in \mathbb{R}^n$  et prévoir la valeur d'un scalaire  $y \in \mathbb{R}$  en sortie. Dans le cas d'une régression linéaire, la sortie est une fonction linéaire de l'entrée. Soit  $\hat{y}$  la valeur que notre modèle prédit que  $y$  devrait prendre. On définit la sortie comme :

$$\hat{y} = w^T x \quad (2.4)$$

où  $w \in \mathbb{R}^n$  est un vecteur de paramètres  $w_i$ .

Les paramètres sont des valeurs qui contrôlent le comportement du système. Dans ce cas,  $w_i$  est le coefficient que nous multiplions par la caractéristique  $x_i > 0$  avant de sommer les contributions de toutes les caractéristiques. Nous pouvons considérer  $w$  comme un ensemble de pondérations qui déterminent comment chaque caractéristique influe sur la prédiction. Si une caractéristique  $x_i$  reçoit un poids positif  $w_i$ , alors augmenter la valeur de cette caractéristique augmente la valeur de notre prédiction  $\hat{y}$ . Si une caractéristique reçoit un poids négatif, alors augmenter la valeur de cette caractéristique diminue la valeur de notre prédiction. Si le poids d'une caractéristique est de grande valeur, alors il a un grand effet sur la prédiction. Si le poids d'une caractéristique est égal à zéro, alors il n'a aucun effet sur la prédiction.

Il est à noter que le terme régression linéaire est souvent utilisé pour désigner un modèle légèrement plus sophistiqué avec un paramètre supplémentaire : l'ordonnée à l'origine  $b$ . L'équation de la sortie peut être définie comme suit :

$$\hat{y} = w^T x + b \quad (2.5)$$

Dans ce modèle, la mise en relation des paramètres aux prédictions reste donc une fonction linéaire, mais celui allant des caractéristiques aux prédictions est désormais une fonction affine. Cette extension à une fonction affine signifie que la courbe des modèles de prédiction ressemble toujours à une droite, mais qui ne passe pas par l'origine (Figure 2.1). Au lieu d'ajouter le paramètre de biais  $b$ , nous pouvons continuer à utiliser le modèle avec uniquement des poids, mais augmentons  $x$  avec une entrée supplémentaire toujours définie à 1. Le poids correspondant à l'entrée supplémentaire joue le rôle du paramètre de biais.

L'ordonnée à l'origine  $b$  est souvent appelée le paramètre de biais de la transformation affine. Cette terminologie provient du fait que la sortie de la transformation est biaisée en faveur de  $b$  en absence de toute entrée. Ce terme diffère de l'idée d'un biais statistique, dans lequel l'estimation attendue d'une quantité par un algorithme d'estimation statistique n'est pas égale à la vraie quantité.

La régression linéaire est un algorithme d'apprentissage extrêmement simple et limité, mais elle fournit un exemple du fonctionnement d'un algorithme d'apprentissage.

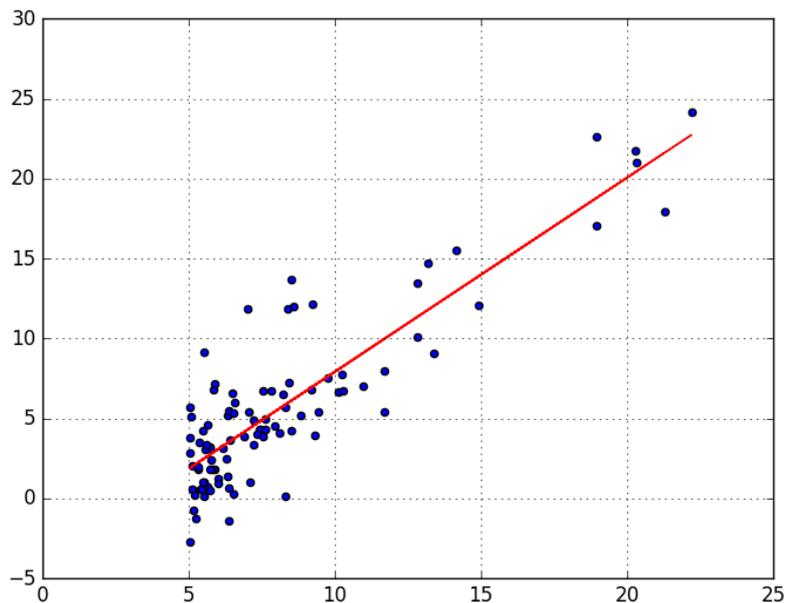


FIGURE 2.1 – Exemple d’une régression linéaire.

### 2.2.2 Régression logistique

La régression logistique prédit la probabilité d’un résultat qui ne peut avoir que deux valeurs (c’est-à-dire une dichotomie). La prédiction est basée sur l’utilisation d’un ou de plusieurs prédicteurs (numériques et catégoriques).

Une régression logistique produit une courbe logistique, qui est limitée aux valeurs comprises entre 0 et 1 (2.6). Elle est similaire à une régression linéaire, mais la courbe est construite en utilisant le logarithme des cotes ou *odds* de la variable cible, plutôt que la probabilité. De plus, les prédicteurs ne doivent pas être distribués normalement ou avoir une variance égale dans chaque groupe.

$$p = \frac{1}{1 + e^{-(b_0 + b_1 x)}} \quad (2.6)$$

Dans la régression logistique, la constante  $b_0$  déplace la courbe à droite et à gauche, et la pente  $b_1$  définit l’inclinaison de la courbe. Un cas particulier de la régression logistique est la fonction Sigmoidale (Figure 2.2) où  $b_0 = 0$  et  $b_1 = 1$ .

Par simple transformation, l’équation de régression logistique peut être écrite en termes de rapport de cotes (2.7).

$$\frac{p}{1 - p} = e^{b_0 + b_1 x} \quad (2.7)$$

Enfin, en prenant le logarithme des deux côtés, nous pouvons écrire l’équation en termes de log-odds (logit) qui est une fonction linéaire. Le coefficient  $b_1$  est la quantité de log-odds (logit) qui varie avec un changement d’une unité dans  $x$ .

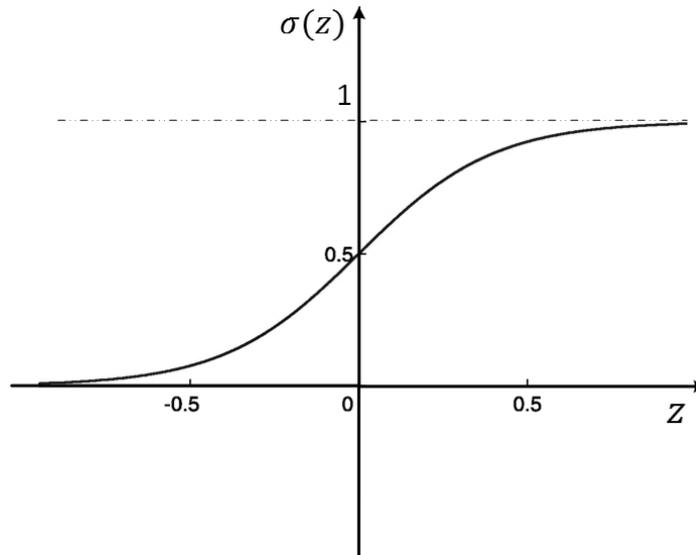


FIGURE 2.2 – Exemple de la fonction Sigmoide.

$$\ln\left(\frac{p}{1-p}\right) = b_0 + b_1 x \quad (2.8)$$

Comme mentionné précédemment, la régression logistique peut gérer un nombre quelconque de variables numériques et/ou catégoriques.

$$p = \frac{1}{1 + e^{-(b_0 + b_1 x_1 + b_2 x_2 + \dots + b_p x_p)}} \quad (2.9)$$

La régression logistique est utilisée dans la plupart des domaines médicaux et les sciences sociales. Par exemple, le score TRISS (*Trauma and Injury Severity Score*), largement utilisé pour prédire la mortalité chez les patients blessés, a été développé à l'origine par Boyd et al. en utilisant la régression logistique [28]. De nombreuses autres échelles médicales utilisées pour évaluer la gravité d'un patient ont été développées à l'aide d'une régression logistique [24, 132, 142, 167]. La régression logistique peut être utilisée pour prédire le risque de développer une maladie donnée (par exemple la diabète ou une maladie coronarienne), sur la base des caractéristiques observées du patient (âge, sexe, indice de masse corporelle, résultats de divers tests sanguins, etc.) [80, 226]. Un autre exemple consiste à prédire si un électeur indien votera au BJP, au Congrès de Trinamool ou au Front de gauche ou au Congrès, en fonction de l'âge, du revenu, du sexe, de la race, de la résidence, des élections précédentes, etc [100]. Cette technique peut également être utilisée en ingénierie, notamment pour prédire la probabilité de défaillance d'un processus, système ou produit donné [185, 215]. Elle est également utilisée dans des applications de marketing telles que la prédiction de la propension d'un client à acheter un produit ou à mettre fin à un abonnement, etc. [21].

### 2.2.3 Arbre de décision

Un arbre de décision [250] est un graphe qui utilise une méthode de branchement pour illustrer tous les résultats possibles d'une décision.

Les arbres de décision peuvent être dessinés à la main ou créés avec un programme graphique ou un logiciel spécialisé. De manière informelle, les arbres de décision sont utiles pour focaliser la discussion lorsqu'un groupe doit prendre une décision. Ils peuvent être utilisés pour attribuer des valeurs aux résultats possibles, afin que les décisions puissent être automatisées. L'algorithme de l'arbre de décision est utilisé dans l'exploration de données pour simplifier les défis stratégiques complexes et évaluer la rentabilité des décisions de recherche et de gestion. Les variables dans un arbre de décision sont généralement représentées par des cercles.

Voici un exemple simple : un arbre de décision de gestion de courrier électronique peut commencer par une zone intitulée « Recevoir le nouveau message ». À partir de là, une branche en début de session peut conduire à « Nécessite une réponse immédiate ». À partir de là, une case « Oui » mène à une seule décision : « Répondez ». Une case « Non » mène à « Cela prendra moins de trois minutes pour répondre » ou « Prendra plus de trois minutes pour répondre ». Dans la première case, une branche mène à « Répondre » et de la deuxième case, une branche mène à « Marquer comme tâche et attribue une priorité ». Les branches pourraient ensuite converger vers « Email répondu à? Classer ou supprimer un message ».

D'après l'exemple précédent, il est clair que l'arbre de décision est caractérisé par une interprétabilité élevée. En effet, il existe une logique lors du suivi des nœuds de l'arbre afin d'atteindre la décision.

Les arbres de décision ont été largement utilisés dans le domaine routier pour la classification du trafic [69]. Elles ont été aussi utilisés dans le domaine médical pour la classification des cellules sanguines [178] et la classification des globules blancs [140]. Les arbres de décision ont été utilisés pour la reconnaissance des formes [110, 177] et la reconnaissance des caractères [95, 228].

## 2.2.4 Machines à vecteurs de support

La machine à vecteurs de support (SVM) est l'une des approches les plus déterminantes de l'apprentissage supervisé utilisées pour la classification ou la régression. Ce modèle est similaire à la régression logistique dans le sens où il est piloté par une fonction linéaire  $w^T x + b$ . Contrairement à la régression logistique, la machine à vecteurs de support ne fournit pas de probabilités, elle ne génère qu'une identité de classe. La classification se fait en séparant le jeu de données en deux classes à travers une droite appelée hyperplan séparateur selon la formule suivante :

$$h(x) = \sum_{i=1}^n w_i a_i + b = w^T \cdot x + b, \quad (2.10)$$

où  $w = (w_1, w_2, \dots, w_n)$  le vecteur de poids,  $x = (a_1, a_2, \dots, a_n)$  l'entrée à classer et  $b$  le biais.

Soit  $x_k$  un point à classer, selon le signe de  $h(x_k)$ , le SVM décide si  $x_k$  appartient à telle ou telle classe (2.11).

$$\begin{cases} h(x_k) \geq 0 \Rightarrow x_k \in \text{classe1} \\ h(x_k) < 0 \Rightarrow x_k \in \text{classe2} \end{cases} \quad (2.11)$$

Comme d'autres algorithmes d'apprentissage supervisé, une SVM nécessite l'apprentissage de données étiquetées pour la classification. Soit  $l_k$  le label du point  $x_k$  ayant la valeur 1 si  $x_k$  appartient à la classe 1 et  $-1$  s'il appartient à la classe 2. Dans le cas où le SVM prédit correctement la bonne classe on aura  $l_k(w^T \cdot x_k + b) \geq 0$ . Donc, le but d'un SVM est de trouver le vecteur de poids  $w$  et le biais  $b$  vérifiant cette relation pour tout  $x_k$  et  $l_k$ . Cela conduit à trouver plus qu'un hyperplan séparant les deux classes. Pour résoudre ce problème, Vapnik et al. [231] ont montré que l'hyperplan optimal consiste à maximiser la marge entre les données et l'hyperplan séparateur. La marge est la distance minimale entre l'hyperplan et les échantillons les plus proches appelés vecteurs de supports (voir Figure 2.3). Cela revient à chercher les uniques  $w$  et  $b$  qui vérifient le problème (2.12).

$$\left\{ \begin{array}{l} \text{minimiser } \frac{\|w\|^2}{2} \\ \text{sous les contraintes } \forall k, \quad l_k(w^T \cdot x_k + b) \geq 1 \end{array} \right. \quad (2.12)$$

La résolution de ce genre de problème (2.12) se fait en utilisant la méthode des multiplicateurs de Lagrange [212] ce qui conduit à avoir les multiplicateurs de Lagrange optimaux  $\alpha_k^*$  (le développement du calcul complet de cette méthode sort du cadre de cette thèse). Après avoir remplacé le vecteur de poids  $w$  par sa valeur optimale  $\sum_{k=1}^p \alpha_k x_k$ , nous pouvons faire des prédictions à l'aide de la fonction :

$$h(x) = \sum_{k=1}^p \alpha_k^* l_k(x \cdot x_k) + b \quad (2.13)$$

Dans le cas non-linéaire où il n'existe pas d'hyperplan capable de séparer les données en deux classes, l'astuce du noyau -*kernel trick*- a été mise en place. L'idée est de considérer que si les données dans l'espace de départ  $E$  sont linéairement inséparables, alors il existe un autre espace  $E'$  de dimension supérieure dans lequel les données seraient linéairement séparables. La redistribution de ces données se fait à l'aide d'une fonction  $\varphi : E \rightarrow E'$  non-linéaire. Cette astuce met en place la fonction noyau  $K : E \times E \rightarrow \mathbb{R}$  définie par  $K(x, x') = \varphi(x)^T \cdot \varphi(x')$ . Dans ce cas, le calcul de l'hyperplan dans  $E'$  ne nécessite que la connaissance de  $K$ . La fonction noyau doit respecter les conditions du théorème de Mercer [171] qui impose que  $K$  soit continue, symétrique et semi-définie positive. Parmi les fonctions noyaux les plus utilisées, il y a le noyau polynomial, le noyau gaussien, le noyau laplacien, etc. En utilisant la fonction  $K$  dans le cas non-linéaire, l'équation de l'hyperplan devient :

$$h(x) = \sum_{k=1}^p \alpha_k^* l_k K(x_k, x) + b \quad (2.14)$$

L'astuce du noyau est puissante pour trois raisons. Premièrement, il nous permet d'apprendre des modèles non-linéaires dans un espace donné en les considérant comme des modèles linéaires dans un espace de dimension plus grande. Deuxièmement, le calcul se fait dans l'espace de départ qui est plus petit ce qui permet d'éviter des calculs coûteux. Troisièmement, la connaissance de la fonction  $\varphi$  n'est pas nécessaire puisqu'elle n'intervient pas dans les calculs.

Les SVM ont été inventés par Vladimir N. Vapnik et Alexey Ya. Chervonenkis en 1963. Depuis cette date, les systèmes ont été utilisés pour la classification des textes et des images. Ils peuvent travailler aussi avec des caractères manuscrits ainsi que la reconnaissance vocale [107, 157]. Les algorithmes ont été utilisés pour résoudre le problème de suivi [149]

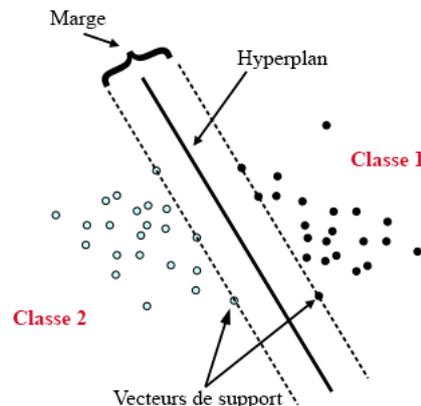


FIGURE 2.3 – Classification avec la machine à vecteurs de support.

et dans les laboratoires de biologie pour effectuer des tâches telles que le tri des protéines. Le SVM a été utilisé aussi par Jun et al. avec le RBM pour résoudre le problème de la détection du réseau trafic [245].

### 2.2.5 K-plus proches voisins

K-plus proches voisins - ou *K-Nearest Neighbors*- (KNN) est une famille de techniques pouvant être utilisées pour la classification ou la régression. En tant qu'un algorithme d'apprentissage non-paramétrique, les K-plus proches voisins ne sont pas limités à un nombre déterminé de paramètres. Par contre, leur complexité dépendra de la taille de la base d'apprentissage.

Contrairement à la régression linéaire, qui a un vecteur de poids de longueur fixe, le modèle de KNN stocke simplement le  $X$  et le  $y$  à partir de l'ensemble d'apprentissage. Lorsqu'il est censé de classer un point de test  $x$ , le modèle recherche l'entrée la plus proche dans l'ensemble d'apprentissage et renvoie la cible de régression associée. En d'autres termes,  $\hat{y} = y_i$  où  $i = \operatorname{argmin} \|X_i - x\|_2^2$ . L'algorithme peut également être généralisé à des métriques de distance autres que la norme  $L^2$ , telles que des métriques de distance apprises [93]. Si l'algorithme est autorisé à rompre les liens en moyennant les valeurs  $y_i$  pour tous les  $X_i$  qui sont attachés au plus proche, alors cet algorithme est en mesure d'obtenir l'erreur d'apprentissage la plus petite possible sur toute la base de régression.

Si nous autorisons l'algorithme à utiliser tous ces voisins pour voter, plutôt que de choisir au hasard l'un d'entre eux, la procédure converge vers le taux d'erreur de Bayes. La grande capacité du KNN lui permet d'obtenir une grande précision compte tenu d'une grande base d'apprentissage. Cependant, cela se fait à un coût de calcul élevé, et peut éventuellement donner de mauvais résultats dans le cas d'un nombre limité de données d'apprentissage.

Une des faiblesses des K-plus proches voisins est qu'il ne peut pas apprendre qu'une caractéristique est plus discriminante qu'une autre. Par exemple, imaginons que nous ayons une tâche de régression avec  $x \in \mathbb{R}^{100}$  tirée d'une distribution gaussienne isotrope, mais qu'une seule variable  $x_1$  est pertinente pour la sortie. Supposons en outre que cette

caractéristique encode la sortie directement, c'est-à-dire que  $y = x_1$  dans tous les cas. La régression du plus proche voisin ne pourra pas détecter ce modèle simple. Le plus proche voisin de la plupart des points  $x$  sera déterminé par le grand nombre de caractéristiques de  $x_2$  à  $x_{100}$  et non par la seule caractéristique  $x_1$ . Ainsi, le résultat sur de petits ensembles d'apprentissage sera essentiellement aléatoire.

L'algorithme KNN peut concourir avec les modèles les plus précis, car il fait des prédictions extrêmement précises. Par conséquent, il peut être utilisé pour les applications nécessitant une grande précision, mais qui ne nécessitent pas la génération de modèles lisibles par l'être humain, car KNN ne produit pas de modèles compréhensibles. Les K-plus proches voisins ont été utilisés dans plusieurs domaines et beaucoup d'applications telles que l'estimation statistique et la reconnaissance de formes [55], la prédiction des événements économiques [119], l'estimation de la capacité d'une batterie lithium-ion [114], la mesure de distance [236], la catégorisation de texte [99] et la classification multi-label [253].

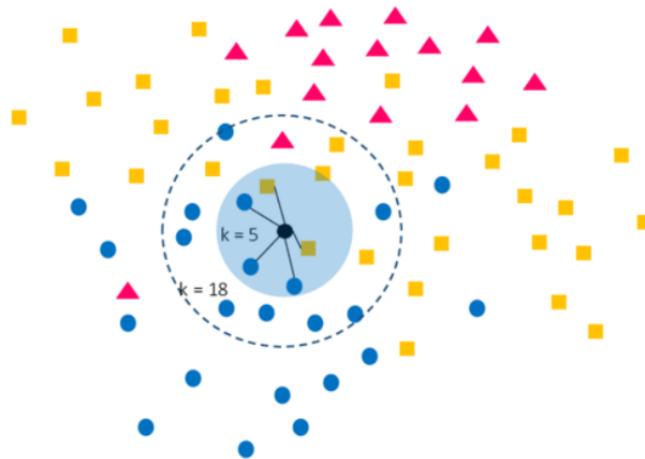


FIGURE 2.4 – Classification avec l'algorithme des  $k$  plus proches voisins, le nouvel individu sera affecté à la classe "cercle bleu" dans les deux cas :  $k=5$  ou  $k=18$ , car la majorité de ses  $k$  voisins appartiennent à cette classe [127].

## 2.3 Deep Learning vs Machine Learning

Durant les dernières années, l'apprentissage profond [161] a fait l'objet de nombreuses études et a obtenu des résultats remarquables dans de nombreux domaines de la vision par ordinateur [41, 240], notamment la classification des images, la reconnaissance vocale [138, 176], la détection d'objets, la segmentation sémantique, le traitement du langage naturel [106, 223], la capture des images, l'estimation de la pose, la détection de la saillance, la détection des contours, etc [149]. Il a montré ses performances face aux différents problèmes de l'intelligence artificielle dépassant ainsi les algorithmes classiques de ML [69, 108, 149, 216, 234]. Dans [69], une comparaison entre les méthodes traditionnelles de l'apprentissage automatique et le Deep Learning a été mise en place. En effet, les auteurs ont utilisé des méthodes de classification classiques, telles que l'arbre de décision *Decision*

*Tree* [190], le classifieur naïve bayésien *Naïve Bayes* [125] et la machine à vecteurs de support (SVM) *Support Vector Machine* [34] pour la classification du trafic. Ils ont comparé les résultats de ces méthodes au résultat obtenu avec un algorithme qui rassemble le réseau de neurones RBM et le SVM [245]. Le meilleur résultat apparaît lorsqu'ils ont utilisé la méthode d'apprentissage SVM-RBM.

Une autre étude comparative d'algorithmes d'apprentissage automatique comprenant un réseau de neurones artificiels, une machine à vecteurs de support et une forêt d'arbres décisionnels *Random Forest* [211] est réalisée, dans [234], pour la prédiction de l'usure des outils d'usinage. Par conséquent, les fonctionnalités conçues manuellement dans la technique traditionnelle d'apprentissage automatique ont conduit à des performances d'inspection insuffisantes et insatisfaisantes dans un processus de changement dynamique [204].

D'après les littératures ci-dessus, les modèles d'apprentissage profond surpassent les techniques d'apprentissage automatique traditionnelles avec des fonctionnalités sophistiquées telles que la machine à vecteurs de support en termes de précision de la classification.

La naissance de l'apprentissage profond profite non seulement de la riche accumulation de techniques traditionnelles d'apprentissage automatique, mais également de l'inspiration de l'apprentissage statistique [234].

Regardons plus en détail les avantages que possède le Deep Learning par rapport au Machine Learning.

L'apprentissage profond et l'apprentissage automatique sont tous les deux capables de gérer les grosses bases de données. Toutefois, les réseaux de neurones profonds évoluent beaucoup mieux avec plus de données que les outils traditionnels de Machine Learning qui ont des problèmes lorsqu'ils sont confrontés au Big Data [89, 135]. Souvent, le meilleur conseil pour améliorer la précision avec un réseau de neurones profond est simplement d'utiliser plus de données! Avec les algorithmes de ML classiques, cette solution rapide et facile ne fonctionne pas aussi bien et des méthodes plus complexes sont souvent nécessaires pour améliorer le résultat de classification ou de régression. En effet, les méthodes de Machine Learning sont beaucoup plus utiles avec des bases de données de petite taille où ils donnent de meilleurs résultats. Par exemple, si nous ne disposons que de 100 images, les arbres de décision, les K-plus proches voisins et d'autres modèles de ML nous seront beaucoup plus efficaces que l'utilisation d'un réseau de neurones profond.

Les algorithmes de ML classiques techniques traditionnelles d'apprentissage automatique requièrent une expertise humaine en matière d'extraction de caractéristiques [234]. Habituellement, une analyse exploratoire approfondie de données est d'abord effectuée sur la base de données. Une réduction de la dimension de la base pourrait alors être effectuée pour faciliter le traitement et rendre les motifs plus visibles pour que les algorithmes d'apprentissage fonctionnent vu que leurs performances dépendent fortement des fonctionnalités techniques. Enfin, les meilleures caractéristiques doivent être soigneusement sélectionnées pour entraîner l'algorithme de ML. Cela n'est pas nécessaire lors de l'utilisation d'un réseau profond vu son succès dans l'extraction automatique de caractéristiques via des transformations non-linéaires multi-couches. En effet, le principal avantage des algorithmes d'apprentissage profond est qu'ils essaient d'apprendre les caractéristiques de haut niveau à partir de données de manière incrémentielle, et donc les données peuvent être transmises directement au réseau pour obtenir de bons scores dès le

départ. Cela élimine totalement la phase d'extraction manuelle des caractéristiques du processus d'apprentissage.

Une autre différence majeure entre la technique d'apprentissage profond et l'apprentissage automatique est l'adaptabilité. En effet, les techniques d'apprentissage profond peuvent être adaptées à différents domaines et applications bien plus facilement que les algorithmes de ML classiques. Premièrement, le transfert d'apprentissage a rendu efficace l'utilisation de réseaux profonds pré-entraînés pour différentes applications dans le même domaine. Par exemple, en vision par ordinateur, les réseaux de classification d'images pré-entraînés sont souvent utilisés pour l'extraction de caractéristiques pour faire la détection et la segmentation des objets. L'utilisation de ces réseaux pré-entraînés facilite l'apprentissage du modèle et permet souvent d'obtenir des performances supérieures en un temps réduit. En outre, les mêmes idées et techniques d'apprentissage profond utilisées dans différents domaines sont souvent transférables. Par exemple, une fois que la théorie d'apprentissage du domaine de la reconnaissance de la parole est comprise, apprendre à appliquer les réseaux profonds au traitement du langage naturel n'est plus une tâche difficile, car les bases sont assez similaires. Contrairement aux algorithmes de ML, les techniques d'apprentissage automatique se diffèrent d'un domaine à autre et d'une application à une autre car cela nécessite souvent une étude spécialisée approfondie dans chaque domaine.

En règle générale, un algorithme d'apprentissage profond prend beaucoup de temps pour faire l'apprentissage à cause du grand nombre de paramètres qu'il utilise. Par contre, les algorithmes traditionnels de ML prennent quelques secondes à quelques heures pour s'entraîner. Le scénario est complètement inversé en phase de test. Au moment du test, l'algorithme de DL prend beaucoup moins de temps à s'exécuter. Par contre, en le comparant avec le K-plus proches voisin, la durée du test augmente avec l'augmentation de la taille des données. Bien que cela ne s'applique pas à tous les algorithmes de ML, certains d'entre eux ont également des temps de test courts.

Bien que le DL présente des performances aussi élevées, le ML présente encore quelques avantages en un certain nombre de situations spécifiques dans lesquelles il serait préférable d'utiliser des algorithmes classiques comme la régression linéaire ou encore l'arbre de décision plutôt qu'un réseau de neurones profond.

Nous pensons que l'utilisation de l'apprentissage profond pourrait répondre à nos besoins dans cette étude, car il a été démontré que les techniques d'apprentissage profond possèdent de puissantes capacités d'extraction de caractéristiques et de classification des objets. Pour cela, nous allons l'adopter dans la suite de notre thèse. Comme nous travaillons dans un contexte routier, la section suivante présentera les différentes applications du Deep Learning utilisées dans le domaine de la route.

## 2.4 Application du Deep Learning au contexte routier

Comme nous avons montré à la section précédente, le Deep Learning a montré ses performances sur différents domaines tels que la vision par ordinateur et la médecine [48, 50, 131, 134, 197, 254]. Vu ses résultats encourageants, beaucoup de chercheurs ont appliqué le Deep Learning dans le contexte routier afin d'assurer la mobilité et la sécurité routière.

Le trafic urbain rencontre des problèmes avec l'augmentation de la mobilité et des demandes de trafic. Les approches visant à réduire les impacts négatifs du trafic incluent d'une part un contrôle amélioré des feux de circulation et d'autre part la mise en place de systèmes de guidage du trafic dynamiques prenant en compte les conditions actuelles. Le système de contrôle du trafic organique (Organic Traffic Control ou OTC) constitue une solution à cet égard. Il fournit un système auto-organisé et auto-adaptatif reposant sur les principes du calcul organique [213]. Dans ce cadre, les auteurs de [164] proposent une méthode basée sur un réseau de neurones convolutif (CNN) permettant l'apprentissage du trafic et de prédire sa vitesse avec une grande précision.

De nos jours, beaucoup de travaux de recherche dans le contexte routier s'orientent vers la conduite autonome. Cette dernière est confrontée aux problèmes de fortes interactions avec l'environnement, y compris d'autres véhicules, des piétons et des travaux routiers. Pour cela, beaucoup de travaux ont essayé de l'améliorer en utilisant l'intelligence artificielle et plus précisément les réseaux de neurones profonds. Nous notons qu'il existe deux défis majeurs qui rendent la conduite autonome différente des autres tâches robotiques. Tout d'abord, il est nécessaire d'assurer la sécurité fonctionnelle - un problème avec lequel l'apprentissage automatique n'a pas réussi à optimiser les performances au niveau attendu par de nombreux algorithmes. Deuxièmement, le processus de décision markovien qui est souvent utilisé en robotique est problématique dans le cas de la conduite autonome en raison du comportement imprévisible d'autres agents. Dans [202, 208], les auteurs ont eu recours à l'apprentissage profond par renforcement qui est considéré comme un puissant paradigme de l'IA pouvant être utilisé pour apprendre aux machines une interaction avec l'environnement en se basant sur des réseaux de neurones récurrents.

La conduite autonome nécessite une détection et une reconnaissance fiables et précises des objets dans des environnements de conduite réels. Dans [227], les auteurs proposent un nouveau système hybride local multiple (LM-CNN-SVM) basé sur les réseaux de neurones convolutifs (CNN) et les machines à vecteurs de support (SVM) en raison de leur puissante capacité d'extraction de caractéristiques et de leur propriété de classification robuste, respectivement. [222] présente une approche qui réunit à la fois la classification, la détection et la segmentation sémantique en utilisant une architecture unifiée dans laquelle l'encodeur est partagé entre les trois tâches. D'autres travaux se sont concentrés sur la détection des objets 3D tels que [38] qui permet d'effectuer une détection d'objet 3D dans des images monoculaires uniques dans le domaine de la conduite autonome, et [39] qui vise la détection d'objets 3D de haute précision dans des scénarios de conduite autonome en utilisant des réseaux 3D Multi-View (MV3D), un framework de fusion sensorielle prenant à la fois les images LIDAR et les images RGB.

Extraire de manière fiable des informations à partir d'images aériennes et par satellite à très haute résolution (VHR) est un problème difficile avec de nombreuses applications pratiques. Un cas particulier de ce problème est la tâche de détecter automatiquement les routes. Bien que cette tâche soit cruciale dans les systèmes de navigation autonomes, elle présente un problème de vision difficile à cause des occlusions, des ombres et d'une grande variété d'objets non-routiers. [235] s'est basé sur des réseaux de neurones profonds (DNN) et une machine à états finis (finite state machine - FSM) pour aboutir à l'extraction des réseaux routiers. Des résultats de certains travaux ont réussi à confronter l'hypothèse disant que l'apprentissage profond est prometteur pour la conduite autonome, tels que [116] où les auteurs ont réussi à détecter les voitures et les voies en utilisant un grand

ensemble de données sur les autoroutes et en appliquant des algorithmes d'apprentissage profond et de vision par ordinateur en temps réel. Aussi, [150, 169, 183] ont proposé des architectures de réseau de neurones convolutif afin d'aborder le problème de la détection automatique des voies de la route. Par contre, [255] a utilisé le réseau résiduel profond U-Net pour réaliser cette tâche. La plupart de ces travaux [31, 116, 169] sont caractérisés par un temps d'inférence rapide ce qui les rend particulièrement adaptés aux applications en temps réel.

[252] s'est particulièrement concentré sur la détection automatique des fissures dans les chaussées pour garantir la sécurité de la conduite. Pour cela, les auteurs ont utilisé un réseau de neurones basé que sur des couches de convolutions appelé ConvNet.

Finalement, dans le domaine routier, la détection des piétons est une tâche importante afin d'assurer la sécurité routière, et même pour faciliter la mobilité. Elle présente plusieurs défis à surmonter, tels que les problèmes d'encombrement de l'arrière-plan, de changement de luminosité, des poses, des points de vue, de résolutions d'image, de paramètres de caméra, et les effets de flou et d'occlusions. Malgré les améliorations significatives, la détection des piétons reste un défi ouvert qui nécessite des algorithmes de plus en plus précis. Pour cette raison, des travaux de recherche ont été mis en place pour améliorer les performances des systèmes de détection de piétons en utilisant les réseaux de neurones profonds [113, 152, 225]. [70] permet une détection fine, rapide et robuste des piétons tout en utilisant une nouvelle approche qui se base sur la fusion des réseaux de neurones profonds. [163] propose une nouvelle méthode basée sur des réseaux de neurones commutables ou *Switchable Deep Network* (SDN) qui apprennent automatiquement les caractéristiques hiérarchiques, les cartes de saillance et les représentations mixtes de différentes parties du corps. [8] en utilisant les réseaux de neurones en cascade pour une détection en temps réel. [156] se base sur le réseau convolutif ConvNet pour la détection multispectrale des piétons tout en analysant le réseau Faster R-CNN. Pour ce faire, les auteurs utilisent des images couleur et thermiques pour l'apprentissage des DNN.

## 2.5 Applications des réseaux de neurones de classification

La classification est l'une des tâches de prise de décision les plus fréquentes de l'activité humaine. Un problème de classification se produit lorsqu'un objet doit être affecté à un groupe ou une classe prédéfini en fonction d'un nombre d'attributs observés liés à cet objet [249]. Afin de trouver une solution à ce problème, des chercheurs dans différents domaines ont eu recours aux réseaux de neurones.

De nombreuses études telles que [56, 173, 194] ont permis d'effectuer plusieurs comparaisons de performances entre classificateurs neuronaux et classificateurs classiques. D'après ces références, les classificateurs basés sur les réseaux de neurones ont donné des résultats prometteurs dans la classification et la reconnaissance. C'est pour cela que les chercheurs les ont appliqués avec succès à diverses tâches de classification du monde réel dans les domaines de l'industrie, des affaires et de la science [238].

Commençons par le domaine des affaires, un cas spécifique a été évalué concerne la prévision de la faillite. L'hypothèse commune à tous les modèles de prévision de la faillite est que les états financiers tiennent compte des facteurs économiques fondamentaux et des caractéristiques d'une entreprise. Par conséquent, en utilisant des outils d'analyse et

des données des rapports financiers de l'entreprise, il est possible d'évaluer et de prédire sa situation financière future [146]. Dans le but de prévoir le risque de la défaillance des entreprises, certains travaux ont mis en place une comparaison entre différents techniques de classification. Dans [79], les auteurs ont utilisé la régression logistique semi-paramétrique et les réseaux de neurones. Tandis que Kar Yan Tam et al. [221] ont comparé l'approche neuronale à un classifieur linéaire, à la régression logistique, et à KNN en utilisant les données par défaut des banques. Dans [146] une comparaison entre différents modèles de réseaux de neurones a été faite afin d'aboutir au modèle donnant la meilleure prédiction. Les scores de prédiction des réseaux de neurones sont également comparés aux résultats obtenus par les modèles classiques d'analyse discriminante. Suite à cette comparaison, les résultats montrent que les réseaux de neurones sont une méthode prometteuse pour évaluer les conditions des banques en termes de précision prédictive, d'adaptabilité et de robustesse [79, 179, 221, 248].

L'une des applications célèbres des réseaux de neurones est reconnaissance des chiffres manuscrits qui est utilisée pour les chèques, les codes postaux, l'immatriculation des voitures, etc. [49, 51, 83, 201, 230]. Ces travaux ont utilisé des réseaux de neurones convolutifs et ont réussi à obtenir des taux de reconnaissance très intéressants; 98% pour [230], 98,6% pour [83] et même 99,32% pour [201].

Un autre système basé sur les réseaux de neurones et qui est largement utilisé est le système de reconnaissance vocale. Il existe des travaux [58, 107, 96] qui ont utilisé les réseaux de neurones profonds ou *Deep Neural Network* (DNN) comportant de nombreuses couches cachées pour réaliser cette tâche. Ce type de réseaux s'est révélé plus performant que les Modèles de mélange gaussien ou *Gaussian Mixture Model* (GMM) [107]. L'amélioration des performances est en partie attribuée à la capacité du DNN à modéliser des corrélations complexes dans les caractéristiques de la parole. Dans [3], les auteurs montrent qu'une réduction supplémentaire du taux d'erreur peut être obtenue en utilisant des réseaux de neurones à convolution ou *Convolutional Neural Network* (CNN) vu leur capacité à normaliser les variations spectrales des signaux vocaux [4]. Les résultats expérimentaux montrent que les CNN réduisent le taux d'erreur de 6% à 10% par rapport aux DNN [3]. La reconnaissance vocale peut être obtenue en se basant sur les réseaux de neurones récurrents (RNN) qui sont des modèles puissants pour les données séquentielles [133].

Continuons avec les systèmes de reconnaissance, généralement utilisés dans les domaines de sécurité et de l'identification des personnes comme la reconnaissance de visage, d'empreintes digitales ou d'iris.

Un système de reconnaissance biométrique comprend deux phases [29] : la création d'un ensemble de motifs biométriques d'utilisateurs autorisés et l'authentification à l'aide d'un réseau de neurones de classification. Pour créer l'ensemble de motifs principaux, une image des empreintes biométriques d'un utilisateur autorisé est divisée en un ensemble de caractéristiques ou en régions d'intérêt. Le système détermine les caractéristiques les plus utiles à des fins d'identification. Des modèles principaux sont ensuite créés à partir de ces caractéristiques principales. Au cours de l'authentification, un ensemble d'exemples d'utilisateurs à authentifier est créé de manière similaire. Ensuite, un réseau de neurones est utilisé pour comparer les motifs prédits aux vrais motifs afin de déterminer si l'utilisateur doit être authentifié. Afin d'éviter les sur-apprentissages et de réduire le temps de calcul, les auteurs de [170] ont utilisé la méthode de l'analyse en composantes principales (ACP) pour extraire, en premier lieu, les caractéristiques faciales. Ensuite, les

caractéristiques résultantes sont traitées par l'analyse discriminante linéaire de Fisher (FLD) pour acquérir des caractéristiques discriminantes de dimension inférieure. Un algorithme d'apprentissage hybride est utilisé pour entraîner les réseaux de neurones RBF. Les résultats des simulations effectuées sur la base de données ORL montrent que le système réalise d'excellentes performances, à la fois en termes de taux d'erreur de classification et d'efficacité d'apprentissage. L'apparition de l'apprentissage profond a considérablement fait progresser la reconnaissance des visages. Les réseaux de neurones très profonds ont récemment connu un grand succès en matière de reconnaissance des objets grâce à leur forte capacité d'apprentissage. [217] propose deux architectures de réseaux de neurones très profondes, appelées DeepID3, pour la reconnaissance des visages. Les deux architectures proposées arrivent à atteindre un taux de reconnaissance de visages de 99,53%. Parmi les systèmes biométriques de vérification d'utilisateur, on peut citer les systèmes de reconnaissance de l'iris [37]. Le système présenté dans cet article est constitué de deux parties principales : une localisation de l'iris et une reconnaissance du motif de l'iris. L'image brute est capturée à l'aide d'un appareil photo numérique. L'iris est ensuite extrait de l'arrière-plan après amélioration et élimination du bruit. En raison du bruit dans le motif de l'iris, seules certaines parties de la structure de l'iris sont sélectionnées pour la reconnaissance. La structure de l'iris sélectionnée est ensuite reconstruite. À l'aide d'un réseau de neurones entraîné, les modèles d'iris sont reconnus avec un taux de reconnaissance d'environ 83%.

On retrouve aussi des applications dans le domaine médical comme la reconnaissance de cellules et de tumeurs. [73] a décrit la méthode qui utilise les réseaux de neurones pour l'identification automatique de la coloration panachée, qui est considérée comme l'une des caractéristiques les plus prédictives du mélanome malin. Dans [129] un réseau de neurones a été utilisé dans un système pour détecter les anomalies dans les cellules, y compris le cancer dans les cellules du tissu de la vessie. Le système comporte un système d'analyse d'image permettant de générer des données représentatives de variables d'imagerie à partir d'une image de cellules colorées. L'ensemble de données est fourni, ensuite, à un réseau de neurones qui a été entraîné pour détecter des anomalies des cellules tissulaires connues. La classification des cellules tissulaires inconnues comme normales ou anormales a été basée sur un réseau neuronal classique activé par un sigmoïde ou sur un réseau de neurones hybride ayant une combinaison de fonctions d'activation sigmoïde, gaussienne et sinusoïdale. Une autre application médicale qui consiste à classer des cancers en catégories de diagnostic spécifiques basées sur leurs signatures d'expression génique a été développée dans [128] à l'aide de réseaux de neurones artificiels (RNA). Les RNA ont été entraînés en prenant pour exemples des petites tumeurs rondes à cellules bleues. Ces cancers appartiennent à quatre catégories de diagnostic distinctes et présentent souvent des dilemmes de diagnostic. Les RNA ont réussi à classer correctement tous les échantillons et identifier les gènes les plus pertinents pour la classification. Cette étude démontre les applications potentielles de ces méthodes pour le diagnostic des tumeurs et l'identification de cibles candidates pour la thérapie. Le cancer du côlon vient en deuxième position après le cancer du poumon en tant que cause de mortalité liée au cancer dans les pays occidentaux. [129] propose une méthode basée sur les réseaux de neurones pour l'analyse du cancer du côlon. Il s'agit de classer les profils d'expression génique dans des cellules normales ou des cellules de cancer du côlon.

Passons au domaine urbain, la reconnaissance des panneaux de signalisation est une

tâche importante et difficile pour les systèmes de transport intelligents [124]. En effet, la reconnaissance d'objets dans les environnements extérieurs pose des problèmes à cause des conditions d'éclairage qui ne peuvent être ni contrôlées ni prédites, les objets peuvent être partiellement occultés et leur position et leur orientation sont inconnues a priori. La reconnaissance des panneaux de signalisation routière est utile pour la maintenance de la signalisation, l'inventaire sur les autoroutes et les villes, les systèmes d'aide à la conduite et les véhicules autonomes [63]. Dans ce papier, un algorithme génétique a été utilisé pour l'étape de détection, permettant une invariance de localisation face aux changements de position, d'échelle, de rotation, de conditions météorologiques, d'occlusion partielle et de présence d'autres objets de la même couleur. Après avoir utilisé un réseau de neurones pour la classification, le système global reconnaît non seulement le panneau de signalisation, mais fournit également des informations sur son état. De nombreux travaux se sont concentrés sur cette reconnaissance en utilisant les réseaux de neurones convolutifs tel que [124] qui a proposé une architecture de réseaux offrant une convergence plus rapide et plus stable que les architectures existantes et un taux de reconnaissance de 99,65%. D'autres travaux ont atteint des scores proches de ce dernier tels que [46] avec un taux de 99,15% en rassemblant le CNN et le MLP, 98,97% pour [207] et 99,46% dans [47] en utilisant une combinaison de plusieurs DNN entraînés sur des données pré-traitées dans un seul DNN à plusieurs colonnes ou *Multi-Column DNN* (MCDNN). Ce dernier permet d'améliorer les performances de reconnaissance en rendant le système insensible aux variations de contraste et d'éclairage.

Dans le cadre de la thèse, nous nous intéressons plutôt à la classification des conditions météorologiques. Pour cela, nous présentons, dans la section suivante, les différents algorithmes de classification météorologique qui existent.

## 2.6 Algorithmes de classification des conditions météorologiques

Durant les dernières années, de nombreuses contributions se sont intéressées à l'impact des conditions météorologiques défavorables sur la circulation telles que la pluie [26, 53, 136, 137, 196] ou le brouillard [17, 188, 187]. Plusieurs travaux se sont alors concentrés sur la classification des conditions météorologiques.

Dans [85, 86, 87], les auteurs mesurent l'intensité de la pluie à l'aide des systèmes de vision artificielle en utilisant la segmentation dans les images. En effectuant une extraction de l'arrière-plan et en comparant des images successives, ils parviennent à détecter des objets en mouvement. Si ces mouvements sont réguliers et dans le même sens, alors il est supposé qu'il pleut. Dans ce cas, ils mettent en place une méthode de mesure de l'intensité de la pluie qui se base sur la surface segmentée en utilisant la vitesse des gouttes.

Comme Narasimhan et Nayar [180] se sont spécialisés dans l'étude des aspects visuels de différentes conditions météorologiques, ils ont abordé la relation entre l'apparence visuelle et les conditions météorologiques [44]. À partir de là, plusieurs travaux ont été proposés pour cibler la tâche de classification de temps. La complexité de cette tâche réside dans la difficulté de trouver un ensemble de caractéristiques les plus pertinentes pour classer les diverses conditions météorologiques. Dans [160], cinq caractéristiques météorologiques comprenant le ciel, les ombres, la réflexion, le contraste et la brume ont

été extraites, et un apprentissage collaboratif a été proposé pour classer les images en deux classes, ensoleillées et nuageuses. Dans [40], les auteurs ont utilisé une méthode similaire pour identifier trois classes de temps : couvert, nuageux et ensoleillé. De même, Zhang et Ma [256] se sont basés sur l'identification et l'extraction manuelles des caractéristiques pour développer leur classifieur météorologique. Ils utilisent des descripteurs globaux et locaux pour représenter les images. En effet, les descripteurs locaux (par exemple le ciel, les ombres, la pluie, les flocons de neige) sont responsables de la caractérisation d'une certaine condition météorologique donnée et donc, ils ne seront pas présents dans les images ayant une condition différente. Pour une visualisation plus généralisée, les auteurs utilisent des descripteurs globaux (par exemple le contraste et la saturation) qui sont présentées en tant que caractéristiques générales pouvant être présentes dans des images avec n'importe quel type de condition. D'autres travaux se sont basés aussi sur les descripteurs locaux et globaux tels que Yan et al. [244] pour identifier trois classes de temps : ensoleillé, nuageux et pluvieux; et [257] pour classifier le temps selon 4 classes : ensoleillé, pluie, neige et brume. Roser et al. [196] ont identifié les conditions météorologiques en trois classes afin de reconnaître les différentes intensités de pluie : temps clair, pluie légère et pluie forte. Ils utilisent une base de données contenant des images capturées par une caméra embarquée et ils proposent une méthode qui regroupe plusieurs caractéristiques d'histogrammes dans un vecteur, telles que la luminosité, le contraste, la netteté, la saturation et la teinte. L'ensemble de cette classification est générée par un SVM linéaire.

Comme [196], Zhang et al. [247] se sont concentrés sur l'identification des différentes intensités d'une seule condition météorologique qu'est le brouillard. La classification est donc faite selon 3 classes : temps clair, brouillard faible et brouillard fort. Pour ce faire, leur méthode est essentiellement basée sur des descripteurs d'images qui sont largement utilisés dans les systèmes de récupération d'informations basées sur le contenu (CBIR) comme CEDD [35], EHD [186], Gabor [62], FCTH [36], FOH [229], etc. La classification a été ensuite effectuée par un SVM. Dans [45], un classifieur par forêt aléatoire a été proposé pour intégrer divers types de caractéristiques visuelles et classer les images dans l'une des cinq conditions météorologiques suivantes : ensoleillé, nuageux, neigeux, pluvieux ou brumeux.

Pendant des années, Jacobs et ses collègues ont mené une série d'études sur les attributs de scène en utilisant une base de données d'image appelée Archive of Many Outdoor Scene (AMOS) [123], dans laquelle les images étaient capturées par des webcams statiques sur une longue période. Dans [123], les auteurs ont découvert que les conditions météorologiques, l'activité humaine et les changements de saison entraînent des variations de scènes à des échelles de temps plus longues. Sur la base du jeu de données AMOS, ils ont proposé que les webcams, installées à travers le monde, puissent être considérées comme des capteurs d'image et nous permettent de comprendre les conditions météorologiques et les variations dans le temps [122]. Plus spécifiquement, ils ont adopté une analyse en composantes principales et une analyse de corrélation pour prédire la vitesse du vent et la pression de vapeur à partir d'une séquence d'images.

Bien que les travaux précédents aient proposé des solutions de classification météorologique intéressantes, ils ne prennent pas en compte toutes les conditions météorologiques possibles en même temps.

Plusieurs travaux récents ont mis en place des réseaux de neurones convolutifs (CNN) dans le domaine de la vision et plus précisément dans la classification et la reconnaissance

des images [68, 90, 134, 206, 210]. Krizhevsky et al. [134] ont mis en œuvre l'architecture spéciale des CNN pour la classification des images de la base de données ImagNet appartenant à 1000 classes d'objets divers, tirant parti des avancées en termes de puissance de calcul nécessaires à son apprentissage. Leur mise en œuvre a montré les capacités d'un classifieur d'image avec une extraction de descripteurs automatique et intégrée basée sur un apprentissage supervisé, au lieu de l'extraction manuelle utilisées dans d'autres applications d'apprentissage automatique [134].

Dans le domaine de la classification météorologique, les CNN ont été utilisés dans [76] et [260]. Dans leur article, Elhoseiny et al. [76] ont créé un CNN basé sur les travaux de Krizhevsky et al. [134] pour la catégorisation des images entre deux classes météorologiques en utilisant la même base de données que [160]. Leur contribution représente une approche judicieuse dans laquelle la technologie utilisée à des fins générales est transformée en une application hautement spécifique, prouvant qu'elle peut atteindre des résultats exceptionnels. Lu et al. [159] ont étendu leur travail de classification météorologique à deux classes en ajoutant les CNN utilisés par [76]. De même, Zhu et al. [260] ont exploré une implémentation qui s'est révélée efficace pour la tâche de classification des images contenant des conditions météorologiques extrêmes, qui est l'architecture connue sous le nom de GoogLeNet, proposée dans [219]. Cette mise en œuvre de CNN a été caractérisée par un nombre important de couches et a donné de meilleurs résultats dans l'expérimentation réalisée dans [260] utilisée sur une collection d'images météorologiques générées par la même équipe.

Di Lin et al. [153] ont proposé un cadre d'apprentissage profond appelé modèle de sélection de régions et de concurrence (RSCM), qui utilisait des indices régionaux dans l'image pour prédire les conditions météorologiques. Les auteurs de [232] ont estimé la température ambiante, la saison, le mois, la semaine et le jour à partir d'une image donnée. L'apprentissage a été effectué sur la base de données AMOS [123], qui contient plus d'un million d'images acquises à partir de différentes webcams situées partout dans le monde. Dans [97], les auteurs ont adapté une méthode qui se base sur les CNN pour faire la classification en cinq classes météorologiques : ensoleillé, nuageux, pluvieux, neigeux, et brouillard. Cette méthode consiste à utiliser en premier lieu la segmentation en superpixel comme un outil d'augmentation de données afin d'améliorer les résultats de classification. Ensuite, des descripteurs sont extraits à partir d'un CNN pré-entraîné pour devenir l'entrée d'un SVM.

Ces travaux identifiés dans l'état de l'art, même s'ils arrivent à classifier le temps, présentent un inconvénient majeur lié aux bases d'apprentissage. En effet, les bases de données utilisées sont généralement très hétérogènes et ou les conditions météorologiques sont inconnues ou ne sont pas précises vu qu'elles ne sont pas caractérisées par des capteurs dédiés à la mesure physique, ce qui pose un problème de fiabilité. Pour cette raison, nous allons utiliser notre propre base de données météorologique (présentée dans la section 4.4) avec les derniers CNN présentés dans la section 3.5.

## 2.7 Applications des réseaux de neurones de mesure

Les réseaux de neurones artificiels (RNA) sont couramment utilisés dans les tâches de décision et de régression en tant qu'estimateurs non-paramétriques. Avec les réseaux

de neurones, des problèmes assez complexes pourraient être résolus beaucoup plus facilement qu'en utilisant des solutions d'ingénierie classiques. Les réseaux de neurones peuvent jouer un rôle important dans les systèmes de mesure; par exemple, ils peuvent être utilisés dans des systèmes de capteurs complexes et dans l'utilisation des données d'observation pour construire des modèles de système. Du point de vue des systèmes de mesures, la propriété clé des réseaux de neurones est leur capacité d'approximation universelle [9, 112].

Il existe de nombreuses applications des RNA dans le domaine de mesures, certaines nécessitant une implémentation logicielle, d'autres nécessitant une implémentation matérielle. Celles-ci concernent l'étalonnage des instruments et des capteurs, le développement de nouvelles méthodes de mesure, le traitement du signal analogique et numérique.

Une brève description, donnée ci-dessous, d'applications typiques devrait permettre de mieux apprécier leurs avantages.

Les RNA ont attiré davantage l'attention sur l'étalonnage des instruments et des capteurs. En intégrant les RNA dans le processus d'étalonnage, la précision a été améliorée et les interférences causées par d'autres variations ont facilement été compensées [10, 126]. Les RNA ont été utilisés pour l'amélioration du capteur de distance à ultrasons [33] et des capteurs à fibre optique pour les mesures de déplacement et de réflectivité de surface [251] et l'étalonnage de réseaux de capteurs, de bras de robots [9], de nez artificiels [84] et d'instruments industriels à six ports hyperfréquences [158].

Les RNA ont été utilisés avec succès pour le développement de nouvelles méthodes de mesure. En astronomie, ils ont été utilisés pour reconstruire une description mathématique du front d'onde entrant. Les mesures sont effectuées au moyen de capteurs à front d'onde pour échantillonner le faisceau optique entrant. En utilisant les étapes des RNA, les difficultés d'interprétation des signaux bruyants obtenus à partir de sources lumineuses faibles ont été surmontées [9]. Dans le traitement des minéraux, les RNA se sont avérés plus efficaces que les techniques classiques de prévision des densités et des distributions d'hydrocyclone [78].

La surveillance, l'estimation et le diagnostic de dynamiques complexes au moyen de RNA ont été utilement appliqués à des dispositifs de mesure linéaires, non-linéaires et hybrides. Les RNA ont été utilisés pour le diagnostic et la surveillance du chauffage du transformateur en tenant également compte des influences environnementales [61], la surveillance en temps réel des systèmes de distribution [19] et la détection des pannes d'instrument [20]. En effet, les machines se dégradent en raison du vieillissement et de l'usure, ce qui diminue la fiabilité de leurs performances et augmente le risque de pannes et de défaillances. L'impact des pannes et des défaillances des machines sur la productivité des usines est une préoccupation importante pour les industries manufacturières. Les impacts économiques liés à la disponibilité et à la fiabilité des machines, ainsi que les coûts de maintenance corrective, ont poussé les usines à améliorer leurs techniques et opérations de maintenance afin de surveiller la dégradation de la machine et de détecter les défaillances. Dans [145] une méthodologie proposée vise la prévention des pannes, réduisant ainsi les coûts de maintenance et améliorant la productivité. Cette méthode pourrait aider les opérateurs à configurer des machines pour un critère donné, à déterminer si la machine fonctionne correctement et à prévoir les problèmes avant qu'ils ne surviennent. En conséquence, les heures de maintenance pourraient être utilisées plus efficacement.

La capacité des RNA à modéliser des systèmes non-linéaires complexes a été utilisée dans la modélisation des dispositifs de mesure. Deux catégories de composants ont été considérées : les dispositifs de mesure analogiques, tels que les capteurs, les transducteurs, les filtres et les amplificateurs ; et les dispositifs de mesure mixtes analogiques/numériques, tels que les convertisseurs ADC et CNA. En utilisant une structure RNA appropriée et une organisation appropriée de l'ensemble de formation, il est possible, après une phase de configuration appropriée, d'obtenir un modèle neuronal fournissant un résultat correspondant à celui obtenu à partir du dispositif réel modélisé. La précision du modèle dépend du choix de la structure RNA et des données d'apprentissage. Dans [166], une approche alternative à la modélisation de pseudo-mesures dans le contexte de l'estimation de l'état du système de distribution a été proposée. De même, la capacité des RNA à modéliser des dispositifs de mesure a été utilisée pour prédire les concentrations d'ozone en fonction des conditions météorologiques et des concentrations des précurseurs [77]. Une station de surveillance mobile a été utilisée pour la collecte de données pendant 60 jours. Les données ont été collectées sur le même site que les mesures de l'ozone. Diverses architectures ont été essayées pendant le processus d'apprentissage. Les entrées dans le réseau de neurones étaient les conditions météorologiques (vitesse et direction du vent, humidité relative, température et intensité solaire) et la concentration de polluants primaires (méthane, monoxyde de carbone, dioxyde de carbone, oxyde d'azote, dioxyde d'azote, dioxyde de soufre, méthane, hydrocarbures et poussières). Un algorithme de rétro-propagation avec impulsion a été utilisé pour préparer le réseau de neurones. Il a été constaté que les précurseurs qui sont le monoxyde de carbone, le dioxyde de carbone, l'oxyde d'azote, le dioxyde d'azote et le dioxyde de soufre avaient le plus grand effet sur la concentration d'ozone prévue. De plus, la température a joué un rôle important. Les performances du modèle de réseau de neurones ont été comparées aux modèles de régression linéaire et non linéaire préparés à partir des données collectées. Il a été constaté que le modèle de réseau de neurones donnait systématiquement des prédictions supérieures. D'après les résultats de cette étude, la modélisation de réseaux de neurones artificiels semble être une technique prometteuse pour la prévision des concentrations des polluants.

Les réseaux de neurones ont montré leurs performances dans un autre type d'application de mesure tel que la mesure de formes. [192] permet de prédire la forme des galaxies à partir d'une grosse base de données d'images. Comme vérité terrain, des mesures de forme conventionnelles ont été utilisées. La prévision des paramètres de forme avec un CNN était extrêmement rapide : elle n'a pris que 0,2 milliseconde par galaxie, améliorant ainsi 4 ordres de grandeurs par rapport à l'ajustement et l'optimisation du modèle. La mesure de la forme est l'un des problèmes critiques de la fabrication de systèmes robotiques [243]. Les coordonnées de points qui ont été obtenus changent distinctement, car différents objets à traiter possèdent différentes formes. Il est donc difficile pour les méthodes traditionnelles d'obtenir des informations précises sur les formes originales. Il affecte toujours les résultats de traitement des systèmes de fabrication robotiques. Selon les exigences de la construction navale, [243] propose une méthode dynamique et intelligente de mesure de forme, basée sur la méthode de logique floue et du réseau de neurones. La logique floue est utilisée pour juger la relation entre les points mesurés.

Comme nous nous intéressons à la mesure des conditions météorologiques dégradées, nous présentons dans la section suivante une sélection d'algorithmes de mesure de ce type de conditions.

## 2.8 Algorithmes de mesure des conditions météorologiques

La visibilité météorologique est une mesure de la transparence de l'atmosphère, facteur important pour la sécurité du transport routier, aérien et fluvial. Le problème de la pollution augmente également le besoin de mesurer la visibilité [151]. Comme de nombreux facteurs, tels que la poussière, la fumée, le brouillard et la brume, influencent la visibilité, sa mesure et son estimation restent un problème très complexe. La surveillance de la visibilité comprend principalement l'observation visuelle par des observateurs météorologiques qualifiés, des mesures optiques par indicateur de visibilité et une évaluation numérique sur l'image de la caméra. En mode visuel, l'être humain peut généralement ignorer le bruit causé par différentes conditions météorologiques et créer un effet visuel.

Comparées aux visiomètres, les techniques de surveillance de la visibilité par caméra sont proches de la méthode visuelle et ont été largement utilisées pour estimer la visibilité pendant des décennies [6, 14, 15, 16, 32, 98, 101, 102, 104, 130, 181, 233, 242, 259]. Les méthodes basées sur les caméras peuvent être principalement divisées en deux classes.

D'une part, la première méthode consiste à utiliser les caméras routières pour le transport routier [101, 102, 104, 181, 259]. Dans ces travaux, les références spéciales de la route, telles que la route droite ou les marques au sol, sont nécessaires pour calculer la portée optique météorologique. Comme la caméra routière ne contient que la scène de la route sur une courte distance, ces méthodes sont utilisées pour une visibilité réduite.

D'autre part, la deuxième méthode correspond aux lois en météorologie, telles que l'équation de Koschmieder ou la loi de Beer-Lambert [6, 14, 15, 16, 32, 98, 130, 233, 242]. Dans cette méthode, les caractéristiques liées au coefficient d'extinction ou à la distance de visibilité sont extraites de l'image pour évaluer la visibilité en se basant sur la loi météorologique. Par exemple, Babari et al. [14, 15, 16] ont proposé un modèle basé sur la physique pour exprimer la correspondance entre le contraste et la distance de visibilité météorologique en tenant compte des surfaces lambertiennes de la scène. Carretas et al. [32] ont utilisé deux ensembles d'arbres situés à des distances différentes, mais connues pour calculer le coefficient d'extinction en fonction des différentes longueurs d'onde associées aux canaux (R, G, B) de l'image pour obtenir la visibilité. Dans la méthode d'image numérique, la qualité de l'image et les objectifs de référence sont importants pour l'extraction de caractéristiques et l'évaluation de la visibilité. La qualité de l'image, telle que le contraste, dépend de nombreux facteurs, notamment la source de lumière, les conditions météorologiques et les paramètres de caméras. La difficulté de cette méthode est liée à l'extraction de toutes les caractéristiques concernées pour décrire l'effet complexe des facteurs de la visibilité.

## 2.9 Synthèse

Dans ce chapitre, nous avons montré que les performances de l'apprentissage profond dépassent les performances des techniques traditionnelles de l'apprentissage automatique. Nous avons présenté aussi les différentes utilisations du Deep Learning dans le contexte routier ainsi que l'utilité des réseaux de neurones de classification et de mesure dans les différents domaines de recherche.

Suite au succès du Deep Learning, nous allons l'utiliser dans la suite de notre thèse.

Dans ce cas, nous aurons besoin de réseaux de neurones pour les faire apprendre. Dans le chapitre suivant, nous allons nous concentrer sur la présentation des différents réseaux de neurones que nous utilisons ensuite dans notre analyse.



# Chapitre 3

## Les réseaux de neurones artificiels

### Sommaire

---

|   |           |
|---|-----------|
| <b>3.1 Introduction</b>                     | <b>34</b> |
| <b>3.2 Historique du Deep Learning</b>      | <b>34</b> |
| <b>3.3 Réseaux de neurones artificiels</b>  | <b>36</b> |
| 3.3.1 Principe de base                      | 36        |
| 3.3.2 Apprentissage d'un réseau de neurones | 41        |
| <b>3.4 Réseaux de neurones convolutifs</b>  | <b>43</b> |
| 3.4.1 Couche de convolution                 | 43        |
| 3.4.2 Couche d'échantillonnage (Pooling)    | 47        |
| 3.4.3 Couche complètement connectée         | 48        |
| <b>3.5 Réseaux de neurones utilisés</b>     | <b>49</b> |
| 3.5.1 LeNet                                 | 49        |
| 3.5.2 ResNet                                | 49        |
| 3.5.3 Inception                             | 52        |
| 3.5.4 DenseNet                              | 54        |

---

## 3.1 Introduction

Dans le chapitre précédent, nous avons montré que la méthode du Deep Learning dépasse les méthodes traditionnelles du Machine Learning. Cette méthode sur laquelle nous sommes basés pour la classification des conditions météorologiques, utilise essentiellement des réseaux de neurones profonds.

Dans ce chapitre, nous allons présenter dans un premier lieu l'historique du Deep Learning et comment il a évolué durant les années (section 3.2). Ensuite, nous allons introduire les réseaux de neurones artificiels et convolutifs respectivement dans les sections 3.3 et 3.4 ainsi que leur fonctionnement. Et enfin, nous présentons dans la section 3.5 les différents réseaux de neurones que nous allons utiliser dans notre système de classification des conditions météorologiques dégradées.

## 3.2 Historique du Deep Learning

Le Deep Learning, en tant qu'une branche du Machine Learning, utilise des algorithmes pour traiter les données et imiter le processus de réflexion chez un humain.

L'histoire de Deep Learning remonte à 1943, lorsque Walter Pitts et Warren McCulloch avaient créé un modèle de calcul pour réseaux de neurones basé sur des mathématiques et des algorithmes, appelé logique à seuil - ou *threshold logic*- pour imiter le processus de pensée du cerveau humain. Depuis ce temps, le Deep Learning a évolué régulièrement, avec seulement deux ruptures significatives dans son développement [168].

En 1958, Frank Rosenblatt a créé le perceptron [195], un algorithme de reconnaissance de formes basé sur un réseau de neurones à deux couches utilisant une simple addition et une soustraction.

Les premiers efforts dans le développement d'algorithmes d'apprentissage profond remontent à 1965, lorsque Ivakhnenko et al. [121] utilisaient des modèles avec des fonctions d'activation polynomiales (équations complexes), qui ont ensuite été analysés statistiquement. À partir de chaque couche, les caractéristiques les mieux choisies sur le plan statistique ont ensuite été transférées à la couche suivante (processus lent et manuel).

Au cours des années 70, l'intelligence artificielle a fait l'objet d'un bref recul. Le manque de financement a limité la recherche en apprentissage en profondeur et en intelligence artificielle. Cependant, certains ont poursuivi la recherche sans financement pendant ces années difficiles. Kunihiko Fukushima a utilisé les premiers réseaux de neurones convolutifs conçus avec plusieurs couches de pooling et de convolution. En 1979, il a développé un réseau de neurones artificiels multi-couches, appelé Neocognitron [82], un réseau de neurones artificiels qui a appris à reconnaître les modèles visuels. Il a été utilisé pour les tâches de reconnaissance de caractères manuscrits et autres modèles de reconnaissance de formes et même le traitement du langage naturel. Son travail, qui a été fortement influencé par Hubel et Wiesel, a conduit au développement des premiers réseaux de neurones convolutifs, basés sur l'organisation du cortex visuel chez l'animal. Ce sont des variantes de perceptrons multi-couches conçues pour utiliser un minimum de prétraitement.

De nombreux concepts de Neocognitron continuent d'être utilisés. L'utilisation de nouvelles méthodes d'apprentissage a permis de réaliser divers réseaux de neurones. Un

Neocognitron moderne peut non seulement identifier les motifs/objets avec des informations manquantes (par exemple, un nombre incomplet 5), mais peut également compléter l'image en ajoutant les informations manquantes. La rétropropagation qui est l'utilisation d'erreurs dans l'apprentissage de modèles de Deep Learning, a considérablement évolué en 1970. C'est à ce moment-là que Seppo Linnainmaa [155] a rédigé sa thèse, comprenant un code FORTRAN pour la rétropropagation. Malheureusement, le concept n'a été appliqué aux réseaux de neurones qu'en 1985. C'est à ce moment que Rumelhart, Williams et Hinton [198] ont démontré que la rétropropagation pouvait fournir des représentations de distribution intéressantes en améliorant considérablement les réseaux de neurones existants pour de nombreuses tâches telles que la reconnaissance de forme, la prédiction de mots, etc. Sur le plan philosophique, cette découverte a mis en lumière, dans le cadre de la psychologie cognitive, la question de savoir si la compréhension humaine repose sur la logique symbolique ou sur des représentations distribuées.

En 1989, Yann LeCun [143] a présenté la première démonstration pratique de la rétropropagation aux Bell Labs. Il a combiné des réseaux de neurones convolutifs à des théories récentes de rétropropagation permettant de lire des chiffres manuscrits. Son système a finalement été utilisé pour la lecture de chèques manuscrits et de codes postaux par NCR et d'autres sociétés, traitant entre 10% et 20% des chèques encaissés aux États-Unis à la fin des années 90 et au début des années 2000. C'est également à cette époque que la deuxième rupture du développement de l'intelligence artificielle a eu lieu (1985-1990), ce qui a également permis d'effectuer des recherches sur les réseaux de neurones et le Deep Learning. En 1995, Dana Cortes et Vladimir Vapnik [54] ont mis au point la machine à vecteurs de support (SVM). C'est un système de reconnaissance et de mappage de données similaires. Il peut être utilisé pour la catégorisation de texte, la reconnaissance manuscrite de caractères et la classification d'images en ce qui concerne l'apprentissage automatique et l'apprentissage en profondeur. La LSTM (mémoire à court et long terme) pour les réseaux de neurones récurrents a été développée en 1997 par Sepp Hochreiter et Juergen Schmidhuber [109]. LeCun a contribué à un autre progrès dans le domaine de l'apprentissage profond lorsqu'il a introduit l'algorithme de descente de gradient stochastique en 1998 associé à l'algorithme de rétropropagation [144].

La prochaine étape importante de l'évolution du Deep Learning a eu lieu en 1999, lorsque les ordinateurs ont commencé à accélérer le traitement des données et que les unités de traitement graphique (GPU) ont été développées. Un traitement plus rapide, avec les GPU traitant les images, a multiplié par 1000 les vitesses de calcul sur une période de 10 ans. Pendant ce temps, les réseaux de neurones ont commencé à concurrencer les machines à vecteurs de support. Alors qu'un réseau de neurones pouvait être lent comparé à une machine à vecteurs de support, les réseaux de neurones offraient de meilleurs résultats en utilisant les mêmes données. Les réseaux de neurones ont également l'avantage de continuer à s'améliorer à mesure que de nouvelles données d'apprentissage sont ajoutées.

Vers l'année 2000, le problème du gradient évanescant est apparu. Il a été découvert que les caractéristiques formées dans les couches inférieures n'étaient pas apprises par les couches supérieures, car aucun signal d'apprentissage n'atteignait ces couches. Ce n'était pas un problème fondamental pour tous les réseaux de neurones, il se limitait aux méthodes d'apprentissage basées sur les gradients. Deux solutions ont été utilisées pour résoudre ce problème : l'apprentissage préalable couche par couche et le développement

de la mémoire à court terme.

En 2009, l'atelier NIPS travaillant sur l'apprentissage profond pour la reconnaissance vocale découvre qu'avec un ensemble de données suffisamment important, les réseaux de neurones n'ont pas besoin d'un apprentissage préalable et que les taux d'erreur chutent considérablement. D'où l'apparition de la base de données ImageNet créée par Fei-Fei Li [64], professeur d'intelligence artificielle à Stanford. Il s'agit d'une base de données gratuite de plus de 14 millions d'images étiquetées.

En 2011, la vitesse des GPU avait considérablement augmenté, ce qui permettait d'entraîner des réseaux de neurones convolutifs sans l'apprentissage préalable couche par couche. Avec l'augmentation de la vitesse de calcul, il devenait évident que le Deep Learning présentait des avantages considérables en termes d'efficacité et de rapidité. AlexNet [134], un réseau de neurones convolutif créé par Alex Krizhevsky dont l'architecture a remporté plusieurs concours internationaux en 2011 et 2012, en est un exemple. AlexNet a construit et amélioré LeNet5 [134] (construit par Yann LeCun des années plus tôt). Il ne contenait initialement que huit couches - cinq couches de convolution suivies de trois couches complètement connectées - et a renforcé la vitesse et le dropout en utilisant des unités linéaires rectifiées (ReLU). Son succès a lancé une renaissance du réseau de neurones convolutif dans la communauté de l'apprentissage profond.

### 3.3 Réseaux de neurones artificiels

#### 3.3.1 Principe de base

L'unité de base du calcul dans un réseau de neurones artificiel est le neurone qui a été défini dans [147] en 1959. Un neurone artificiel reçoit des entrées de certains autres neurones ou d'une source externe ayant des valeurs numériques  $x_1, x_2, \dots, x_n$  auxquels il est connecté par des synapses et calcule une sortie  $y$ . Chaque entrée  $x_i$  a un poids associé  $w_i$ , qui est attribué en fonction de son importance relative par rapport aux autres entrées. La valeur d'entrée  $x$  du neurone correspond à la somme pondérée de ses entrées en ajoutant une autre entrée ayant un poids  $b$  appelé biais. Ensuite, le neurone applique une fonction  $f$  sur cette somme, comme illustré à la Figure 3.1.

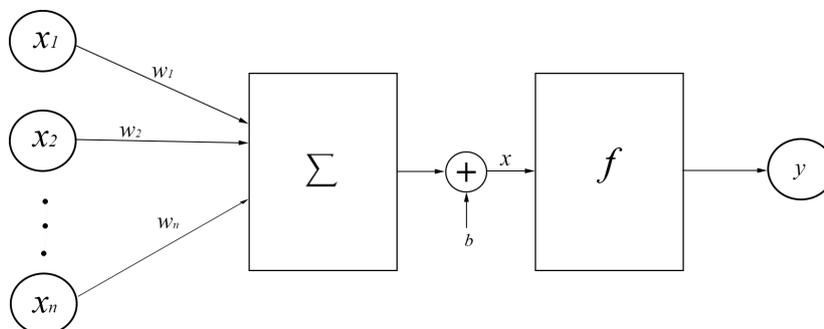


FIGURE 3.1 – Modélisation d'un neurone artificiel.

La sortie du neurone  $y$ , appelée activation de sortie, est calculée selon la formule suivante :

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

où  $f$  est la fonction d'activation, qui sera détaillée ultérieurement.

Le réseau de neurones à propagation avant -ou feedforward neural network- était le premier et le plus simple réseau de neurones artificiel mis au point. Il contient plusieurs neurones disposés en couches. Les neurones des couches adjacentes ont des connexions entre eux. Toutes ces connexions ont des poids qui leur sont associés. Un exemple de réseau de neurones à propagation avant est présenté sur la Figure 3.2.

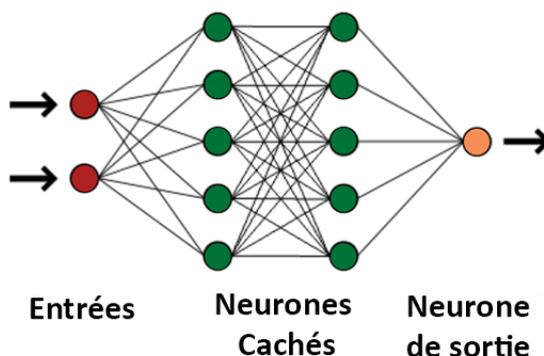


FIGURE 3.2 – Exemple d'un réseau de neurones artificiel.

Comme le montre la Figure 3.2, un réseau de neurones à propagation avant peut être constitué de trois types de neurones :

- neurones d'entrée : les neurones d'entrée fournissent des informations du monde extérieur au réseau et construisent ensemble une « couche d'entrée ». Aucun calcul n'est effectué dans les nœuds d'entrée. Ils transmettent simplement les informations aux nœuds cachés.
- neurones cachés : les neurones cachés n'ont pas de connexion directe avec le monde extérieur (d'où le nom "caché"). Ils effectuent des calculs et transfèrent des informations des neurones d'entrée aux neurones de sortie. Un ensemble de neurones cachés forme une « couche cachée ». Bien qu'un réseau à propagation avant n'ait historiquement qu'une seule couche d'entrée et une seule couche de sortie, il peut avoir aucune ou plusieurs couches cachées, selon un principe de généralisation.
- neurones de sortie : les neurones de sortie construisent ensemble une « couche de sortie ». Ils sont responsables des calculs et du transfert des informations du réseau vers le monde extérieur.

Dans un réseau à propagation avant, les informations ne se déplacent que dans un seul sens, en partant des neurones d'entrée, passant par les neurones cachés (s'ils existent) et en arrivant aux neurones de sortie. Un réseau de neurones peut être assimilé à un graphe orienté acyclique. Deux exemples de réseaux à propagation avant sont présentés dans la Figure 3.3 :

- perceptron mono-couche - Il s'agit d'un réseau de neurones à propagation avant le plus simple qui contient une couche cachée (Figure 3.3a).

- perceptron multi-couches - Il comporte une ou plusieurs couches cachées. Dans le cadre de ces travaux, nous ne traiterons que le cas des perceptrons multi-couches, car ils sont plus utiles que les perceptrons mono-couches dans les applications actuelles (Figure 3.3b).

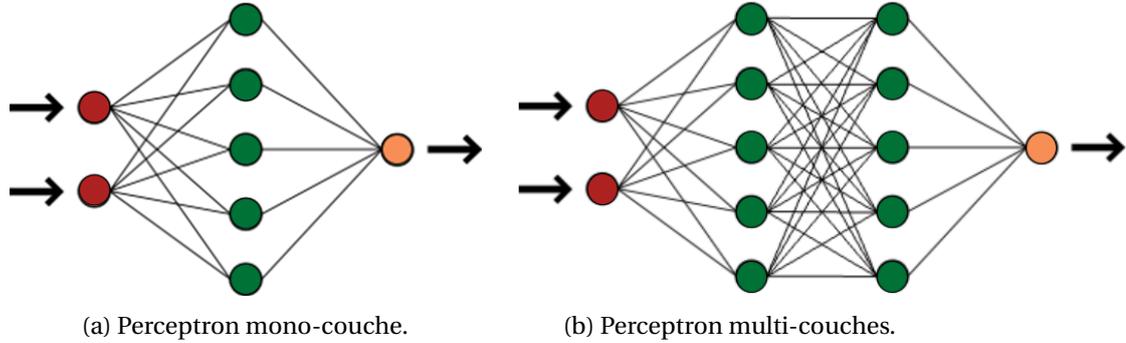


FIGURE 3.3 – Deux types de perceptrons.

En se basant sur l'équation (3.1), un réseau de neurones peut être défini de la manière suivante. Soient :

- $r$  le nombre des neurones sur la couche d'entrée
- $s$  le nombre des neurones sur la couche de sortie
- $l$  le nombre total des couches du réseau de neurones
- $n_k$  le nombre des neurones sur la  $k^{\text{ème}}$  couche avec  $0 \leq k \leq l$  tel que  $n_0 = r$  et  $n_l = s$
- les fonctions affines  $A_i^k(x) = \sum_{j=1}^{n_{k-1}} w_{ij}^k x_j + b_i^k$  avec  $1 \leq k \leq l, 1 \leq i \leq n_k, x \in \mathbb{R}^{n_{k-1}}$
- $f$  la fonction d'activation du réseau de neurones
- le vecteur de sortie  $Y = (y_1, \dots, y_s)$  défini par la récurrence suivante :

$$\forall x \in \mathbb{R}^r, \begin{cases} a_0 = x \\ a_i^k = f(A_i^k(a^{k-1})) & \text{avec } \forall 1 \leq k \leq l, 1 \leq i \leq n_k \\ y_i(x) = A_i^l(a^{l-1}) & \text{avec } \forall 1 \leq i \leq s \end{cases} \quad (3.2)$$

A partir de la Figure 3.2 et en se basant sur les équations de (3.2), nous pouvons représenter le réseau de neurones sous le graphe de la Figure 3.4.

La fonction  $f$ , appelée fonction d'activation, est non linéaire. Elle a pour but d'introduire la non-linéarité dans la sortie d'un neurone. Ceci est important car la plupart des données du monde réel sont non linéaires et donc les neurones apprendront ces représentations non-linéaires. Elle devrait être aussi différentiable pour pouvoir appliquer ensuite l'algorithme de rétropropagation du gradient. Chaque fonction d'activation prend un nombre unique et effectue une certaine opération mathématique sur lui. Il existe plusieurs fonctions d'activation utilisées dans la pratique (Figure 3.5) :

- **la fonction Sigmoïde** : prend une entrée réelle et la réduit entre 0 et 1

$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}} \quad (3.3)$$

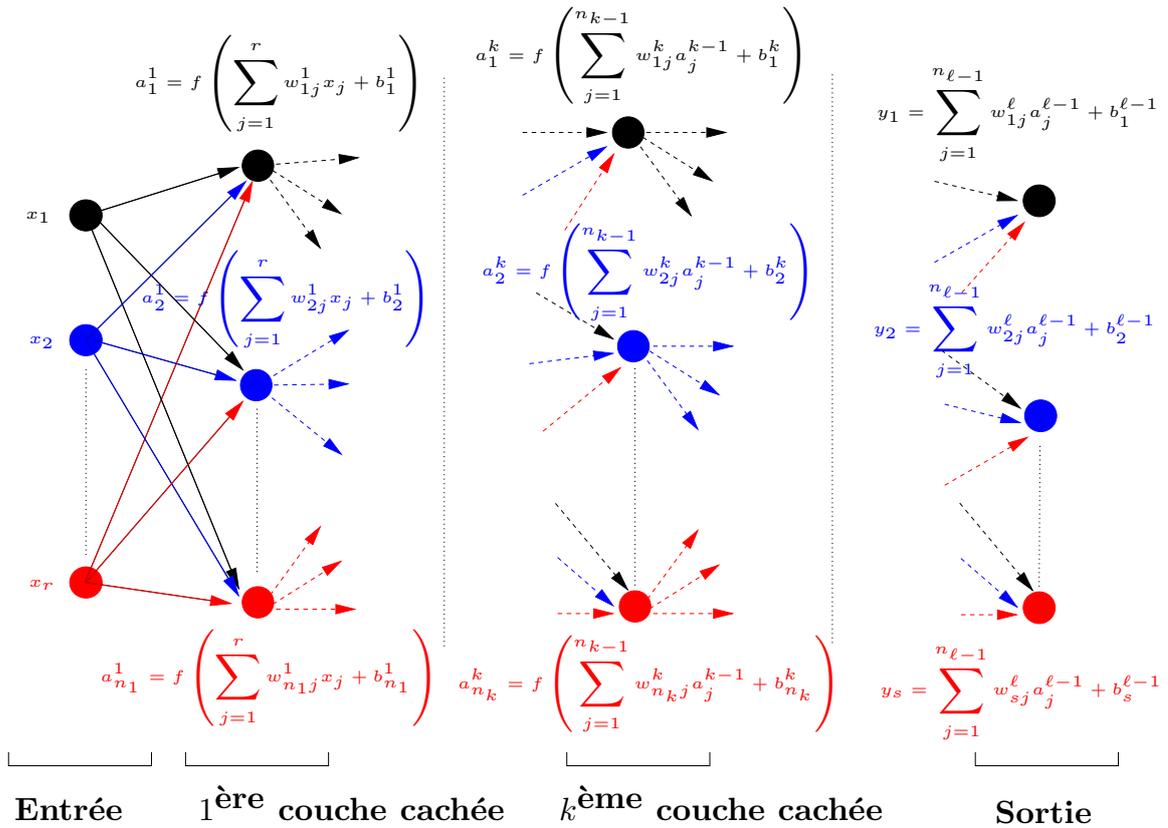


FIGURE 3.4 – Schéma d’un réseau de neurones artificiel à  $l - 1$  couches cachées.

- **la fonction Tangente hyperbolique** : prend une entrée de valeur réelle et la réduit à une valeur dans  $[-1, 1]$

$$f(x) = \tanh(x) = 2\sigma(2x) - 1 \tag{3.4}$$

Ces deux fonctions sont saturantes (fonctions qui présentent des paliers<sup>1</sup>) dont l’inconvénient est d’avoir des gradients évanescents, c’est-à-dire les gradients tendent vers 0. En effet, durant la phase d’apprentissage, le calcul du gradient à chaque itération permet d’avoir une décroissance exponentielle du gradient. Dans le sens inverse, il peut y avoir des gradients explosifs où la valeur présente une croissance exponentielle. En anticipant un peu, en regardant les gradients à calculer par la formule (3.11), nous remarquons qu’ils sont des produits de gradients et donc si les gradients sont inférieurs à 1 le résultat va se tendre exponentiellement vers 0, sinon le résultat va se tendre exponentiellement vers l’infini. Afin d’éviter ce problème de gradients évanescents, l’utilisation de fonctions d’activation non-linéaires et non-saturantes (fonctions qui ne présentent pas des paliers) est recommandée. Pour cela, une opération supplémentaire appelée ReLU (Rectified Linear Unit) a été utilisée après chaque opération de convolution dans [92] montrant ainsi une optimisation des réseaux de neurones artificiels. A noter que cette fonction est non-différentiable mais les procédures pouvaient être adaptées pour pouvoir l’utiliser.

La fonction ReLU est une opération élémentaire (appliquée par élément) et remplace toutes les valeurs négatives par zéro. Elle est définie comme suit :

1. Asymptotes aux infinis

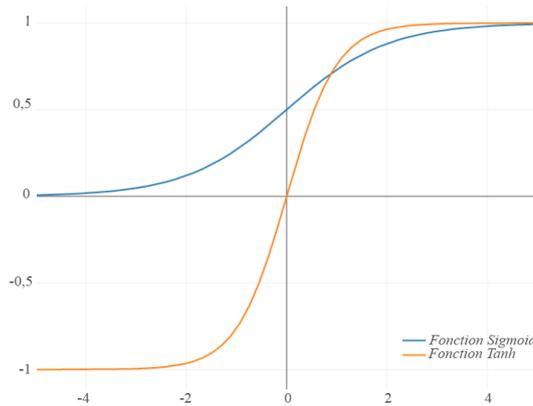


FIGURE 3.5 – Exemples de fonctions d’activation.

$$f(x) = \max(0, x) \tag{3.5}$$

En comparaison avec sigmoïde et tanh, la fonction ReLU accélère le temps d’apprentissage ce qui entraîne une efficacité calculatoire [92].

En 1989, Cybenko [57] a mis en place le théorème d’approximation universelle. Ce théorème montre que, dans le cas d’une fonction d’activation sigmoïdale, les fonctions calculables par un perceptron sont denses dans l’ensemble des fonctions continues par morceaux sur des compacts. Ce théorème a été ensuite généralisé pour l’ensemble des fonctions d’activations par Hornik [111].

**Théorème 1.** Soit  $f$  une fonction bornée, croissante non-constante. Soit  $C_0^n$  l’ensemble des fonctions continues définies sur  $[0, 1]^n$ . Alors :

$\forall \epsilon > 0, \forall F \in C_0^n, \exists N \in \mathbb{N}^*,$  des réels  $v_i, b_i \in \mathbb{R}$  et des vecteurs  $w_i \in \mathbb{R}^n$  avec  $i \in [1, n]$  tels que :

$$\hat{F} : x \rightarrow \sum_{i=1}^N v_i f(w_i^t x + b_i), \tag{3.6}$$

soit une approximation de  $F$  à  $\epsilon$  près, c’est-à-dire :

$$\forall x \in [0, 1]^n, |F(x) - \hat{F}(x)| < \epsilon \tag{3.7}$$

Ce théorème montre que toute fonction relativement régulière (continue par morceaux sur un ensemble de compacts) peut être approchée avec une précision arbitraire par un perceptron monocouche. Donc, toute fonction peut être simulée par un réseau de neurones artificiel.

Cependant, il existe une limitation liée à ce théorème. Bien que l’existence d’un perceptron qui approche n’importe quelle fonction est garantie, aucune indication sur sa méthode de construction n’est fournie. Donc, avoir des poids atteignables par une descente de gradient n’est pas garanti. Cela dit le théorème est valable pour des réseaux de neurones simples à une seule couche, or il existe des études qui ont montré que les réseaux de neurones profonds sont plus performants que les réseaux simples [22, 172]. Dans ce

cas, les chercheurs utilisent différentes méthodes pour construire les réseaux de neurones profonds telles que la méthode essai-erreur [165, 189].

Il existe des études en mathématiques appliquées qui s'intéressent à l'expressivité des réseaux de neurones. En effet, il s'agit d'étudier la capacité de ces réseaux à représenter certaines classes de fonctions. Les recherches ont montré que l'expressivité est proportionnelle à la profondeur du réseau ainsi qu'à sa largeur [154]. En effet, la profondeur d'un réseau de neurones est définie par le nombre de couches qu'il contient et la largeur par le nombre de neurones sur les couches. Avec l'évolution de la performance des réseaux de neurones profonds, plusieurs recherches en mathématiques appliquées se sont concentrées sur les bases théoriques qui ont participé à ce succès [162]. Ceci est très important afin de posséder des fondations solides qui aident à mieux comprendre l'apprentissage profond.

### 3.3.2 Apprentissage d'un réseau de neurones

La phase d'apprentissage consiste à ajuster les poids d'un réseau de neurones afin d'obtenir les meilleurs résultats de régression ou de classification dans notre cas. L'objectif de cette optimisation est de minimiser la fonction de coût  $\psi$  (3.8) qui calcule l'erreur entre les données de la base d'apprentissage  $y$  et les données prédites par le réseau de neurones  $\hat{y}$ .

$$\psi(p) = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (3.8)$$

L'un des algorithmes d'optimisation proposés est l'algorithme de rétropropagation qui se base sur la descente de gradient [141, 199, 237]. Nous montrons brièvement comment cet algorithme permet de calculer le gradient du réseau en se basant sur un autre réseau de neurones qui est adjoint de celui d'origine.

Soit un réseau de neurones ayant  $l + 1$  couches et  $z_k$  la sortie de la couche  $k$ . A partir de la fonction (3.2), nous notons  $g_{k+1}$  la fonction de transfert  $A_i^k$  et  $\alpha_{k+1}$  son vecteur de paramètres. Lors de la phase d'apprentissage, le réseau transfère l'entrée  $e$  vers la sortie  $z_l$  selon la récurrence définie dans (3.9).

$$\begin{cases} z_0^p = e \\ \vdots \\ z_{k+1}^p = g_{k+1}(\alpha_{k+1}, z_k) \\ \vdots \\ z_l^p = g_l(\alpha_l, z_{l-1}) \end{cases} \quad (3.9)$$

où  $p = (\alpha_1, \dots, \alpha_l)$

Avec ces notations, la fonction de coût définie dans (3.8) est exprimée par  $\psi(p) = Q(p, z_l^p)$  où la fonction  $Q$  est dérivable. Le gradient de  $\psi$  peut s'écrire :

$$\nabla_p \psi(p) = \nabla_p Q(p, z_l^p) + \frac{\partial z_l^p}{\partial \alpha_k}(p) \times \nabla_z Q(p, z_l^p) \quad (3.10)$$

La dérivée partielle de  $z_l^p$  dans (3.10) peut s'exprimer par :

$$\frac{\partial z_l^p}{\partial \alpha_k}(p) = \nabla_{\alpha} g_k(\alpha_k, z_{k-1}^p) \times [\nabla_z g_l(\alpha_l, z_{l-1}^p) \times \dots \times \nabla_z g_{k+1}(\alpha_{k+1}, z_k^p)], \quad (3.11)$$

où  $\nabla_{\alpha} g_k$  et  $\nabla_z g_k$  sont les matrices jacobiennes de  $g_k$  pour les variables  $\alpha$  et  $z$  respectivement et  $\times$  présente le produit matriciel.

A partir de (3.11), nous pouvons écrire la récurrence rétrograde suivante :

$$\begin{cases} \lambda_l^p = I \\ \lambda_k^p = \nabla_{\alpha} g_k(\alpha_k, z_{k-1}^p) \times \lambda_{k+1}^p & \text{avec } 1 \leq k \leq l-1 \\ \frac{\partial z_l^p}{\partial \alpha_k}(p) = \nabla_z g_k(\alpha_k, z_{k-1}^p) \times \lambda_k^p & \text{avec } 1 \leq k \leq l \end{cases} \quad (3.12)$$

où  $I$  est la matrice unité.

En se basant sur les deux premières lignes de (3.12) et en les comparant à celles de (3.2), nous retrouvons un réseau de neurones ayant comme fonction de transfert les matrices jacobiennes de  $g_k$  et les  $\lambda_k$  les activations de sorties des neurones. La seule différence entre les deux réseaux est le sens de parcours qui est inversé d'où la rétropropagation de l'erreur.

L'algorithme de descente de gradient permet d'approcher son minimum local en cherchant les points auxquels son gradient est nul. Dans le cas des réseaux de neurones, cela signifie que, à chaque itération, la rétropropagation calcule la dérivée de la fonction de coût  $\psi$  par rapport à chaque poids  $W = (w_1, \dots, w_n)$  et la soustrait de ce dernier selon l'équation (3.13).

$$p_{i+1} = p_i - \alpha \nabla_p \Psi_{\{1, \dots, N\}}(p_i), \quad (3.13)$$

où nous notons désormais  $\Psi_{\omega}(p)$  la fonction :

$$\Psi_{\omega}(p) = \frac{1}{k} \sum_{j=1}^k (y_{i_j} - \hat{y}_{i_j})^2, \quad (3.14)$$

pour tout élément  $\omega = \{i_1, \dots, i_k\}$  de  $\Omega = \mathcal{P}(\{1, \dots, N\})$  l'ensemble des données de la base d'apprentissage avec  $1 \leq i_1 < i_2 < \dots < i_k \leq N$ .

Dans la littérature,  $\omega$  est appelé lot ou *batch*. Les bases de données pouvant être très importantes ( $N$  très grand), nous limitons l'évaluation de la fonction  $\psi$  à des batchs de cardinal bien plus petit que  $N$ . En pratique, nous nous donnons une taille "raisonnable"  $k$  du batch, puis nous tirons au hasard de manière uniforme les indices  $i_1, \dots, i_k$  et nous calculons  $\nabla \Psi_{\omega}(p)$  par :

$$\nabla \Psi_{\omega}(p) = \frac{1}{k} \sum_{j=1}^k \nabla_p \Psi_{\{i_j\}}(p), \quad (3.15)$$

car nous avons :

$$\Psi_{\omega}(p) = \frac{1}{k} \sum_{j=1}^k \Psi_{\{i_j\}}(p), \quad (3.16)$$

Cette approche consiste à échantillonner la base de données  $\Omega$  pour estimer la fonction  $\Psi_{\{1, \dots, N\}}$  qui peut être vue comme une moyenne. L'utilisation des mini-lots permet d'améliorer la convergence du modèle et la rendre plus stable en réduisant le nombre de

passages sur la base d'apprentissage. Cela permet aussi de se contenter de charger d'une manière partielle les données en mémoire.

La valeur retenue par (3.15) pour le gradient de  $\psi$  correspond à la méthode de gradient stochastique.

Le paramètre  $\alpha$  de l'équation (3.13) est le pas de la descente. Il est connu sous le nom de taux d'apprentissage -ou *learning rate*- et il est considéré comme l'un des hyperparamètres importants pour l'optimisation des réseaux de neurones profonds en agissant sur sa convergence. En effet, un taux d'apprentissage trop élevé conduit à des mises à jour des poids importantes et la convergence devient instable. Par contre, pour un taux d'apprentissage faible, la convergence est ralentie avec une possibilité de tomber dans des minima locaux. L'approche populaire utilisée dans l'apprentissage profond pour avoir le taux d'apprentissage optimal est de commencer l'apprentissage avec une valeur élevée afin d'accélérer la descente du gradient et de la réduire par la suite pour améliorer la précision [18]. Pratiquement, il s'agit d'initialiser un  $\alpha_0$  à une valeur élevée au début et la faire diminuer ensuite par un facteur multiplicatif constant  $\gamma$  pendant la phase de l'apprentissage selon la formule (3.17), jusqu'à ce que l'erreur de validation atteigne une valeur stable ou lorsque l'erreur de l'apprentissage ne décroît plus [27]. Cette relation est proposée par :

$$\alpha_{i+1} = \alpha_0(1 + \gamma i)^{-1} \quad (3.17)$$

Parmi les problèmes qui peuvent être rencontrés durant le processus de l'apprentissage, il y a le surapprentissage. Il s'agit d'un mauvais paramétrage des hyperparamètres de la descente de gradient ou à des problèmes liés à la base de données. Suite à un surapprentissage, le réseau de neurones devient capable de capturer des informations qui ne sont pas utiles pour accomplir sa tâche ou il devient incapable de généraliser les caractéristiques des données, ce qui limite sa capacité de reconnaître de nouvelles données. Afin d'éviter ce problème de surapprentissage, des techniques de régularisation ont été proposées. La première technique est le Dropout [214]. Il s'agit d'éliminer certains neurones à chaque itération afin de diminuer le nombre de paramètres du réseau et de mettre à jour un nombre réduit de poids. La deuxième technique est l'augmentation de données. Il s'agit d'augmenter le nombre des échantillons d'apprentissage en ajoutant une certaine variété sans affecter leur sémantique. Par exemple, nous pouvons ajouter des images en introduisant des rotations, de la symétrie verticale ou même changer leurs dimensions.

## 3.4 Réseaux de neurones convolutifs

### 3.4.1 Couche de convolution

Un réseau de neurones convolutif [94, 144] est un réseau de neurones qui utilise une opération mathématique qui s'appelle convolution ou produit de convolution. Il s'agit d'une opération linéaire. Chaque réseau de neurones convolutif contient au moins une couche de convolution. Soient  $f$  et  $g$  deux fonctions définies sur  $\mathbb{R}$ , le produit de convolution entre  $f$  et  $g$  est généralement noté  $f * g$  et il est défini par l'équation suivante :

$$s(x) = (f * g)(x) = \int_{-\infty}^{+\infty} f(t)g(x - t) dt \quad (3.18)$$

Dans un réseau de neurones convolutif, le premier argument  $f$  est assimilé à l'entrée, le deuxième argument  $g$  est assimilé au noyau de convolution et la sortie est assimilée à la carte de caractéristiques. En réalité, les réseaux de neurones convolutifs utilisent une convolution discrète qui est définie par :

$$s(x) = (f * g)(x) = \sum_{a=-\infty}^{+\infty} f(t)g(x - a) \quad (3.19)$$

En apprentissage automatique, l'entrée est toujours un tableau de données multidimensionnel, et le noyau est toujours un tableau de paramètres multidimensionnel qui seront adaptés par l'algorithme d'apprentissage. Ces tableaux sont référencés par des tenseurs. La convolution est toujours utilisée avec une dimension supérieure à 1. La convolution la plus utilisée est une convolution 2D. Dans ce cas, pour une entrée image  $I$  et pour un noyau  $K$ , la convolution discrète s'écrit :

$$S(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i - m, j - n)K(m, n) \quad (3.20)$$

D'après cette formule, l'image  $I$  et le noyau de convolution  $K$  sont parcourus dans des sens opposés. En effet, quand  $m$  augmente, les indices de  $I$  décroissent et les indices de  $K$  augmentent. La seule raison d'utiliser la convolution est sa propriété de commutativité. Cependant, cette propriété n'est pas importante dans l'implémentation du réseau de neurones, car les autres fonctions avec lesquelles la convolution est utilisée ne sont pas commutatives. C'est pour cette raison que la plupart des bibliothèques de réseaux de neurones implémentent la fonction de la corrélation croisée mais ils l'appellent convolution (3.23). Cet opérateur est identique à la convolution sauf qu'il perd la propriété de commutativité.

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i + m, j + n)K(m, n) \quad (3.21)$$

Une couche de convolution est caractérisée par :

- les dimensions des noyaux de convolution, généralement une convolution à une dimension égale à 2 avec des noyaux carrés.
- le nombre des filtres de convolution  $C$ , c'est le nombre de cartes d'activations, ou cartes de caractéristiques, en sortie de la couche. Ces cartes sont représentées sous la forme de tenseurs de dimension 3 ( $H, W, C$ ) avec  $H$  la hauteur des cartes,  $W$  la largeur et  $C$  le nombre de canaux.
- le pas de convolution -ou *stride* -  $s$ . C'est le pas de décalage du noyau de convolution à chaque calcul. Par exemple dans le cas de deux dimensions  $s = (s_1, s_2) = (1, 1)$ , le noyau de convolution sera déplacé d'un seul pixel vers la droite pour chaque mouvement horizontal du noyau et d'un seul pixel vers le bas pour chaque mouvement vertical du noyau lors de la création de la carte de caractéristiques.
- le padding  $p$ . C'est le paramètre permettant de dépasser la taille de l'image pour appliquer la convolution en ajoutant des pixels autour de l'image.

Dans la suite, et afin de faciliter l'analyse et simplifier la visualisation, nous utilisons une convolution 2D ( $N = 2$ ), avec une entrée carrée  $h = w$ , un noyau de convolution carré de dimension  $k$ , un pas  $s$ , et un padding  $p$ .

A noter que les équations suivantes peuvent être généralisées pour le cas d'une convolution N-D et pour les cas non-carrés.

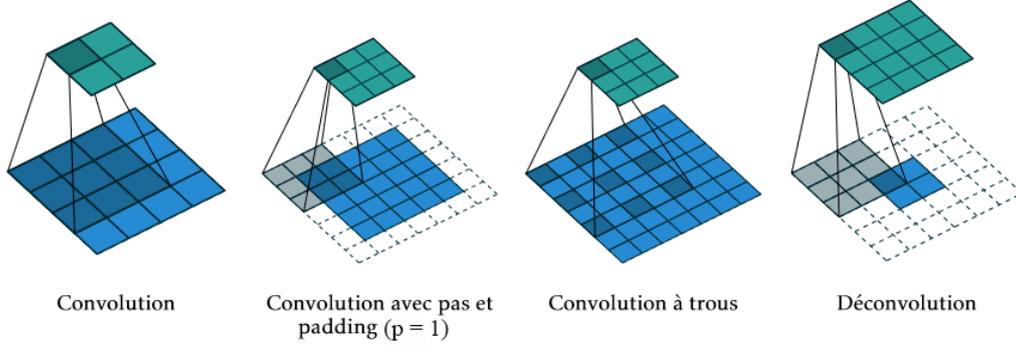


FIGURE 3.6 – Différents types de convolution [72].

Il existe différents types de convolutions (Figure 3.6) :

**Convolution classique ou convolution à pas :** C'est la convolution dont nous avons parlé ci-dessus. Couramment utilisée, elle est définie par la taille du noyau de convolution  $k$ , le pas  $s$  et le padding  $p$ . Il s'agit de réduire la dimension des cartes d'activation produites par un facteur de  $s$  selon l'équation (3.22). Ce type de convolution consiste à faire glisser le noyau de convolution de gauche à droite et du haut au bas avec le pas  $s$  jusqu'à ce qu'il arrive à parcourir toute l'image.

$$o = \frac{h - k + 2p}{s} + 1 \quad (3.22)$$

Soit  $I$  une image de taille  $h \times h$  et  $I'$  de taille  $(h + 2p) \times (h + 2p)$  construite à partir de  $I$  en ajoutant  $p$  pixels aux alentours de l'image, dans le cas où  $p = 0$ , l'image  $I'$  sera identique à l'image  $I$ . L'équation de la convolution devient ainsi :

$$S(i, j) = (K * I')(i, j) = \sum_{m=0}^{k-1} \sum_{n=0}^{k-1} I'(s \cdot i + m, s \cdot j + n) K(m, n) \quad (3.23)$$

Avec  $i \leq h - k + 2p + 1$  et  $j \leq h - k + 2p + 1$

**Convolution à trous ou convolution dilatée [246] :** Elle se base sur un noyau dit éclaté ou dilaté d'un facteur de dilatation  $d$ . Il s'agit de parcourir l'image en sautant certains de ses pixels. Les pixels manquants dans le noyau sont remplacés par des 0. La dimension des cartes d'activation de sortie est calculée comme suit :

$$o = \frac{h - k - (k - 1)(d - 1) + 2p}{s} + 1 \quad (3.24)$$

**Convolution transposée ou déconvolution :** Elle s'oppose à la convolution classique en construisant une image  $I$  à partir des cartes d'activation dont les dimensions sont calculées comme suit :

$$o = (h - 1)s + k - 2p \quad (3.25)$$

A partir de 3.26, la convolution transposée peut être assimilée à une convolution à pas avec  $s = \frac{1}{s'}$  ( $s' \in \mathbb{N}^*$ ). Ce type de convolution est utilisé dans la phase décodage d'un auto-encodeur convolutif [258].

La convolution exploite trois idées importantes susceptibles d'améliorer un système d'apprentissage automatique :

- **les interactions convolutives** : Dans un réseau de neurone traditionnel, les couches utilisent une multiplication matricielle par une matrice de paramètres avec un paramètre séparé décrivant l'interaction entre chaque unité d'entrée et chaque unité de sortie. Cela signifie que chaque unité de sortie interagit avec chaque unité d'entrée. Cependant, les réseaux convolutifs ont généralement des interactions claires-mées. Ceci est accompli en rendant le noyau plus petit que l'entrée. Par exemple, l'image d'entrée peut comporter des milliers ou des millions de pixels, mais grâce aux couches de convolution, nous pouvons détecter de petites caractéristiques utiles, telles que les contours avec des noyaux n'occupant que des dizaines ou des centaines de pixels. Cela permet de stocker moins de paramètres, ce qui réduit les besoins en mémoire et améliore l'efficacité de l'algorithme. Cela signifie également que le calcul de la sortie nécessite moins d'opérations.
- **le partage des paramètres** : Il consiste à utiliser le même paramètre pour plusieurs fonctions dans un modèle. Dans un réseau de neurones traditionnel, chaque élément de la matrice de pondération est utilisé une seule fois lors du calcul de la sortie d'une couche. Il est seulement multiplié par un élément de l'entrée. Le partage de paramètres permet d'avoir des poids liés, c'est-à-dire, la valeur du poids appliqué à une entrée est liée à la valeur d'un autre poids appliqué ailleurs. Le partage de paramètres utilisé par l'opération de convolution signifie qu'au lieu d'apprendre des ensembles de paramètres distincts pour chaque emplacement, nous apprenons un seul ensemble. Cela n'affecte pas le temps d'exécution de la propagation en avant mais il réduit davantage les exigences de stockage du modèle.
- **les représentations équivariantes** : La couche de convolution équivariante à la translation. Une fonction est équivariante signifie que si l'entrée change, la sortie change de la même manière. Autrement, une fonction  $f(x)$  est équivalente à une fonction  $g(x)$  si  $f(g(x)) = g(f(x))$ . Dans le cas d'une couche de convolution, une translation de l'image d'entrée entraîne la translation des cartes d'activation.

Yann LeCun [144] a introduit les couches de convolution au début d'un réseau de neurones afin d'extraire les caractéristiques des images d'une manière pertinente à travers les noyaux de convolutions. À noter que la pertinence des caractéristiques est proportionnelle à l'intervention tardive des convolutions. C'est-à-dire, plus la convolution intervient tard, plus les noyaux se complexifient et deviennent capables de détecter des formes plus détaillées (Figure 3.7). En effet, à la première couche de convolution, les valeurs d'un noyau sont initialisées aléatoirement. Ensuite, ces valeurs seront mises à jour avec le processus d'apprentissage du réseau afin d'améliorer les résultats d'extraction des caractéristiques.

Les couches de convolution sont généralement suivies d'une couche de ReLU (introduite précédemment) afin de modifier les cartes de caractéristiques de sortie appelées ainsi cartes de caractéristiques rectifiées.

L'application de la fonction ReLU à la sortie des couches de convolution possède plusieurs avantages :

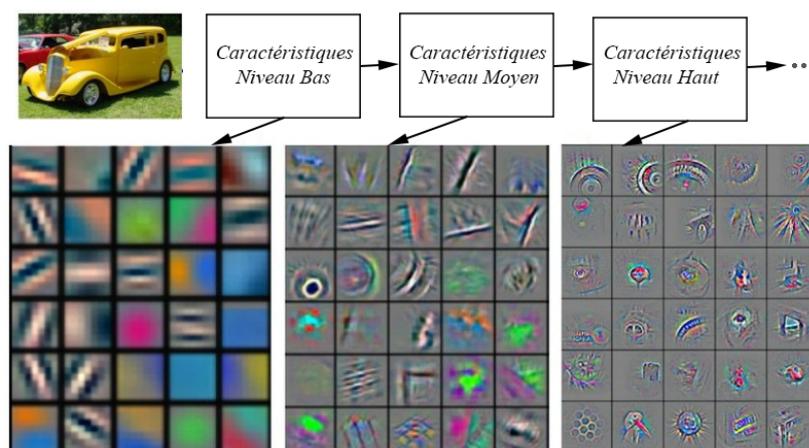


FIGURE 3.7 – Cartes des caractéristiques.

- la convolution réalise des opérations d’additions et de multiplication, ce qui conserve la linéarité de la sortie par rapport à l’entrée. Le fait d’appliquer la fonction ReLU, introduit une certaine non-linéarité (en supprimant les pixels de valeur négative).
- suite à la suppression d’une partie des données, la fonction ReLU permet l’accélération des calculs.
- la mise en valeur des caractéristiques extraites par les couches de convolution en accentuant l’écart entre eux (valeurs négatives).

### 3.4.2 Couche d’échantillonnage (Pooling)

Semblable à la couche de convolution, la couche d’échantillonnage est chargée de réduire la taille spatiale des cartes de caractéristiques, mais elle conserve les informations les plus importantes. Il existe différents types d’échantillonnage dont l’échantillonnage maximum -ou *Max Pooling*-, l’échantillonnage moyen -ou *Average Pooling*-, etc.

L’échantillonnage consiste à appliquer un noyau de taille  $n \times n$  sur la carte d’activation en le faisant glisser avec un pas préalablement défini (le pas est généralement égal à la taille du noyau  $n$  pour éviter le phénomène de chevauchement). Le *Max Pooling* renvoie la valeur maximale de la partie de l’image couverte par le noyau. Au lieu de prendre le maximum, nous pourrions prendre la moyenne de tous les éléments couverts par le noyau, cela est assuré par le *Average Pooling*. Le *Max Pooling* permet de supprimer le bruit. D’autre part, le *Average Pooling* effectue simplement une réduction de la dimensionnalité en tant que mécanisme de suppression de bruit. Par conséquent, en pratique, le *Max Pooling* est beaucoup plus utilisé que le *Average Pooling* puisqu’il fonctionne mieux.

La forme la plus courante est une couche de Pooling avec des noyaux de taille  $2 \times 2$  appliqués avec un pas de 2 réduisant ainsi la dimension de l’entrée par 2, en éliminant 75% des activations et en laissant la dimension de profondeur inchangée. Dans la Figure 3.8, le nombre des activations passe de 16 à 4 en appliquant le pooling.

Plus généralement, la couche de pooling :

- accepte un tenseur de taille  $W_1 \times H_1 \times D_1$
- nécessite deux hyperparamètres : la taille  $F$  et le pas  $s$  avec  $F < W_1$  et  $F < H_1$

— produit un tenseur de taille  $W_2 \times H_2 \times D_2$  où :

- $W_2 = \frac{(W_1-F)}{s} + 1$
- $H_2 = \frac{(H_1-F)}{s} + 1$
- $D_2 = D_1$

— introduit zéro paramètre puisqu'il calcule une fonction fixe de l'entrée.

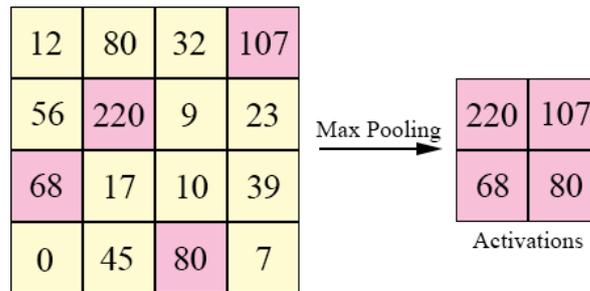


FIGURE 3.8 – Exemple d'une opération de pooling de taille  $2 \times 2$ .

L'échantillonnage a pour fonction de réduire progressivement la taille spatiale de la donnée d'entrée. En effet, il permet de :

- rendre les représentations d'entrée (carte des caractéristiques) plus petites et plus faciles à gérer.
- réduire la puissance de calcul requise pour traiter les données par réduction des dimensions et donc il contrôle le sur-apprentissage.
- extraire les caractéristiques dominantes qui sont invariantes en rotation et en position, maintenant ainsi l'efficacité du processus d'apprentissage du modèle.
- aider à obtenir une représentation équivariante de l'image. Ceci est très puissant puisqu'il permet de détecter des objets dans une image, peu importe où ils se trouvent.

### 3.4.3 Couche complètement connectée

La couche complètement connectée est un Perceptron multi-couches traditionnel qui utilise une fonction d'activation (par exemple softmax) sur le vecteur de sortie afin d'ajouter la non-linéarité. Le terme « complètement connecté » implique que chaque neurone de la couche précédente est connecté à chaque neurone de la couche suivante. Leurs activations peuvent donc être calculées avec une multiplication matricielle suivie d'un offset de biais.

La sortie des couches de convolution et d'échantillonnage représente des caractéristiques de haut niveau de l'image d'entrée. L'objectif de la couche complètement connectée est d'utiliser ces caractéristiques pour classer l'image d'entrée du réseau en différentes classes en fonction de la base de données d'apprentissage.

Outre la classification, l'ajout d'une couche complètement connectée est généralement un moyen peu coûteux pour faire apprendre des combinaisons non-linéaires des caractéristiques. La plupart des cartes de caractéristiques des couches de convolution et d'échantillonnage peuvent être utiles pour la tâche de classification, mais une combinaison de ces fonctionnalités peut être encore meilleure.

Les couches entièrement connectées sont généralement suivies d'un Dropout [214]. Ce dernier agit sur les poids de ces couches afin de désactiver un certain nombre de neurones pour réduire le nombre de paramètres. Cela permet de contrôler le sur-apprentissage qui peut être causé par un nombre important de paramètres.

La somme des probabilités en sortie de la couche entièrement connectée est  $\sum_i \hat{y} = 1$ . Cela est garanti en utilisant Softmax comme fonction d'activation dans la couche en sortie de la couche entièrement connectée. La fonction Softmax prend un vecteur de scores arbitraires à valeurs réelles et le réduit à un vecteur de valeurs comprises entre zéro et un qui égal à un. Elle s'écrit sous la forme :

$$\hat{y}_i = \text{softmax}(x)_i = \frac{e^{x_i}}{\sum_j e^{x_j}} \quad (3.26)$$

### 3.5 Réseaux de neurones utilisés

Dans cette section, les réseaux de neurones convolutifs utilisés dans cette thèse sont présentés.

#### 3.5.1 LeNet

L'un des premiers réseaux de neurones qui ont contribué à propulser le domaine de l'apprentissage profond est LeNet [144] qui a été conçu par Yann LeCun en 1998. À cette époque, l'architecture LeNet était utilisée principalement pour reconnaître les chiffres et appliquée à la reconnaissance de nombres manuscrits sur des chèques. L'architecture de LeNet est présentée dans la Figure 3.9. Elle contient deux blocs constitués d'une couche de convolution suivie d'une couche de pooling, et deux couches complètement connectées semblables aux perceptrons multi-couches classiques.

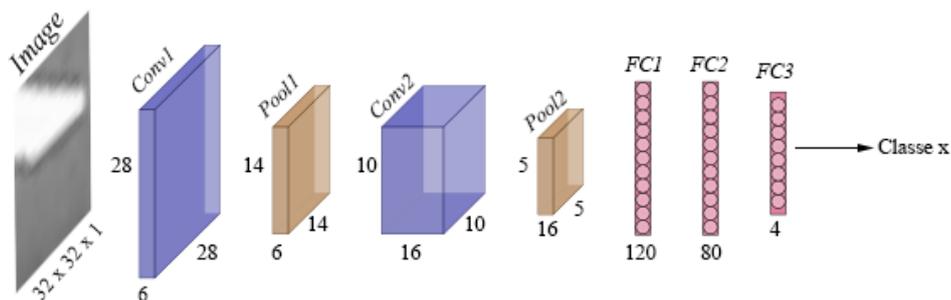


FIGURE 3.9 – Architecture du réseau de neurones LeNet pour un exemple de classification en 4 classes météorologiques (adapté de [11]).

#### 3.5.2 ResNet

Le réseau de neurones résiduel -ou *Residual Network*- plus connu par le nom ResNet [105] apparaît suite au problème de rétropropagation du gradient et l'augmentation de l'erreur d'apprentissage. En effet, lorsque le réseau de neurones est trop profond, le gradient se réduit à zéro et devient très faible pour mettre à jour les poids des différentes couches

d'où la difficulté de l'apprentissage. Avec ResNet, l'optimisation des réseaux profonds est assurée en utilisant des connexions résiduelles. Cela permet aux gradients de passer dans deux couches de convolution, mais aussi de passer directement à travers un saut aux couches suivantes en reliant l'entrée de la  $n$ ème couche à la sortie de la  $(n + a)$ ème couche (Figure 3.10).

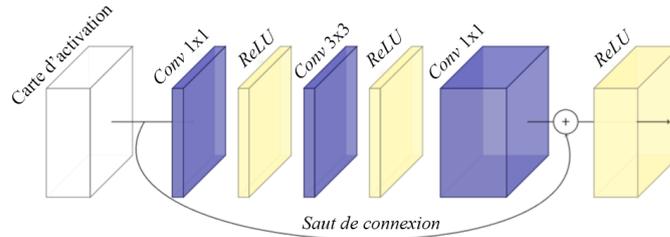


FIGURE 3.10 – Bloc convolutif résiduel (adapté de [11]).

Ce bloc résiduel peut être modélisé par la relation suivante :

$$x_l = H_l(x_{l-1}) + x_{l-1}, \quad (3.27)$$

avec  $x_l$  la sortie du  $l^{\text{ème}}$  bloc résiduel et  $H$  une fonction composée qui représente l'ensemble des couches présentes dans le bloc résiduel (par exemple les 3 couches de convolution et les 2 couches de ReLU dans la Figure 3.10).

Les résultats de classification basés sur cette architecture ont prouvé que l'apprentissage d'un réseau profond est plus facile que l'apprentissage de simples réseaux de neurones convolutifs profonds et que le problème du gradient est résolu. Dans [105], les auteurs présentent 5 versions de ResNet : ResNet-18, ResNet-34, ResNet-50, ResNet-101, ResNet-152 (Tableau 3.1). La différence entre ces différentes architectures se situe au niveau du nombre des couches (18 couches, 34 couches, etc.). Le Tableau 3.1 présente les détails de chaque architecture. Prenons l'exemple de ResNet-34 (Figure 3.11). Cette architecture contient une couche de convolution  $conv1$  avec un noyau carré  $k = 7$ , un pas  $s = 2$  et un nombre de filtres de convolution  $C = 64$ . Cette couche est suivie d'une couche de  $max\ pooling$  de  $3 \times 3$  avec un pas de 2. Cette dernière est suivie de 4 couches de convolution ( $conv2_x$ ,  $conv3_x$ ,  $conv4_x$  et  $conv5_x$ ). Chaque couche contient un certain nombre de blocs convolutifs résiduels constitués de deux couches de convolution de  $3 \times 3$  contournées par l'entrée. Ces blocs ont l'avantage de conserver les dimensions en hauteur et en largeur pendant toute la couche  $((H, W, C) = (56, 56, 64))$  pour la couche  $conv2_x$ . Par contre, en passant d'une couche à l'autre, les dimensions changent suite à une augmentation du pas de la couche de convolution de 1 à 2, ce qui réduit sa taille par 2  $((H, W, C) = (28, 28, 128))$  pour la couche  $conv3_x$ . D'habitude, dans les réseaux de neurones convolutifs simples, la réduction de la taille de convolution est assurée par une couche de pooling. Enfin, ces 4 couches sont suivies d'une couche de pooling moyen et d'une couche complètement connectée de dimension 1000 (pour le cas d'une classification sur la base de données ImageNet qui contient 1000 classes).

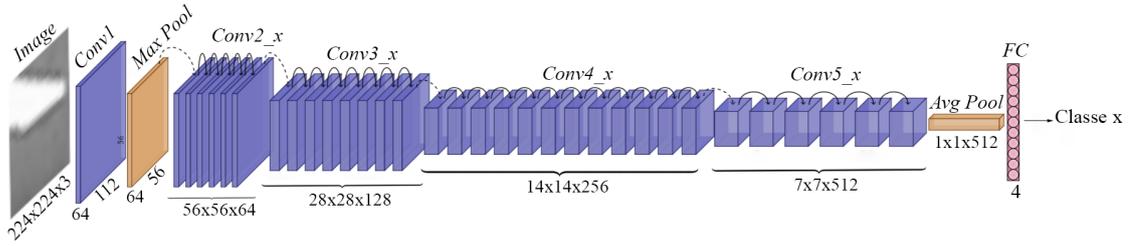


FIGURE 3.11 – Architecture du réseau ResNet-34 pour un exemple de classification en 4 classes météorologiques (adapté de [11]).

| Nom de la couche | Taille de sortie | 18 couches  | 34 couches  | 50 couches  | 101 couches  | 152 couches  |
|------------------|------------------|---|---|---|--|--|
| conv1            | 112x112          | 7x7, 64, pas 2  |   |   |  |  |
| conv2_x          | 56x56            | 3x3 max pooling, pas 2  |   |   |  |  |
|                  |                  | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$   | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$   | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$    | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$     | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$     |
| conv3_x          | 28x28            | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$  | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$   | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$   |
| conv4_x          | 14x14            | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$ |
| conv5_x          | 7x7              | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$  | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$  |
|                  | 1x1              | Moy pooling, 1000-d fc, softmax   |   |   |  |  |

TABEAU 3.1 – Différentes architectures de ResNet (adapté de [105]).

En se basant sur (3.27), ce réseau ResNet-34 peut être écrit sous la forme suivante :

$$\left\{ \begin{array}{l}
 x_0 = e \\
 x_1 = f_{c,w}(x_0) \\
 x_2 = f_{p,w}(x_1) \\
 \vdots \\
 x_{t+1} = H_{1,w}(x_t) + x_t \quad \text{avec } t = 2 \cdots 4 \\
 \vdots \\
 x_{t+1} = H_{2,w}(x_t) + x_t \quad \text{avec } t = 5 \cdots 8 \\
 \vdots \\
 x_{t+1} = H_{3,w}(x_t) + x_t \quad \text{avec } t = 9 \cdots 14 \\
 \vdots \\
 x_{t+1} = H_{4,w}(x_t) + x_t \quad \text{avec } t = 15 \cdots 17 \\
 \vdots \\
 x_{18} = f_{s,w}(x_{17})
 \end{array} \right. \quad (3.28)$$

Avec

- $e$  : les données d'entrée du réseau
- $w$  : le vecteur de poids dont dépend chaque couche
- $f_{c,w}$  : la fonction exprimant la couche de convolution conv1

- $f_{p,w}$  : la fonction exprimant la couche max pooling
- $f_{s,w}$  : la fonction exprimant la couche de sortie du réseau
- $H_{l,w}$  : la fonction exprimant le bloc résiduel de la couche de convolution  $l$

Dans le cadre de cette thèse, le réseau ResNet-152 sera utilisé en l'adoptant si besoin.

### 3.5.3 Inception

L'un des réseaux les plus complexes est Inception qui a été construit par Google. Il a été conçu de façon à améliorer les performances des réseaux de neurones profonds à la fois en termes de rapidité et de précision. Son évolution a conduit à la création de plusieurs versions du réseau. Les versions populaires sont les suivantes : GoogLeNet ou Inception-v1 [219], Inception-v2 [220], Inception-v3 [220], et Inception-v4 [218]. Chaque version est une amélioration itérative par rapport à la précédente. La compréhension des mises à niveau peut nous aider à créer des classifieurs personnalisés optimisés en vitesse et en précision. L'architecture de ce réseau se base essentiellement sur des modules d'inceptions, d'où son nom. Chaque version de ce réseau a abouti à la construction d'un nouveau module. Inception-v1 a présenté le premier module d'inception. Celui-ci a été créé suite à 3 problèmes. Premièrement, la difficulté de transmettre les mises à jour de gradient à travers un réseau trop profond. Deuxièmement, le fait d'empiler naïvement de grandes opérations de convolution, augmente le temps de calcul. Et troisièmement, vu que l'emplacement des informations dans une image est très varié, il devient difficile de choisir la bonne taille du noyau pour l'opération de convolution. Un noyau plus gros est préférable pour les informations distribuées plus globalement, et un noyau plus petit est préféré pour les informations distribuées plus localement. Pour cela, un module d'inception a été mis en place permettant de tester plusieurs configurations de convolution avec des filtres de tailles multiples ( $1 \times 1$ ,  $3 \times 3$ ,  $5 \times 5$ ) sur le même niveau à travers des convolutions en parallèle. Ensuite, une couche de pooling est appliquée. Pour rendre le réseau moins coûteux en termes de temps de calcul, les auteurs limitent le nombre de canaux d'entrée en ajoutant une convolution supplémentaire de  $1 \times 1$  avant les convolutions  $3 \times 3$  et  $5 \times 5$ . Cependant, cette couche est ajoutée après la couche de pooling. Aussi, ils ont factorisé les convolutions avec des filtres de  $5 \times 5$  en deux convolutions de  $3 \times 3$ , car les couches de  $5 \times 5$  sont 2,78 fois plus coûteuses que les couches de  $3 \times 3$ . Les sorties sont ensuite concaténées et envoyées au module suivant. Cette dernière modification a été introduite lors de la construction du réseau Inception-v2, d'où la construction du module d'inception A (Figure 3.12a). Un autre type de factorisation consistait à remplacer les convolutions de taille de filtre  $n \times n$  en une combinaison de convolutions  $1 \times n$  et  $n \times 1$  ce qui limite la complexité de calcul. En fixant  $n$  à 7, le module d'inception B (Figure 3.12b) prend place lors de la construction de Inception-v3. Le fait d'empiler les filtres les uns après les autres réduit excessivement les dimensions et donc il y aura une perte d'informations. Dans ce cas une autre technique a été introduite afin de remplacer un filtre par deux autres en parallèle, d'où la construction du module d'inception C (Figure 3.12c). Ces 3 modules d'inception ainsi que 3 autres blocs ont contribué à la création du réseau Inception-v4 (Figure 3.13) que nous utiliserons dans la section suivante.

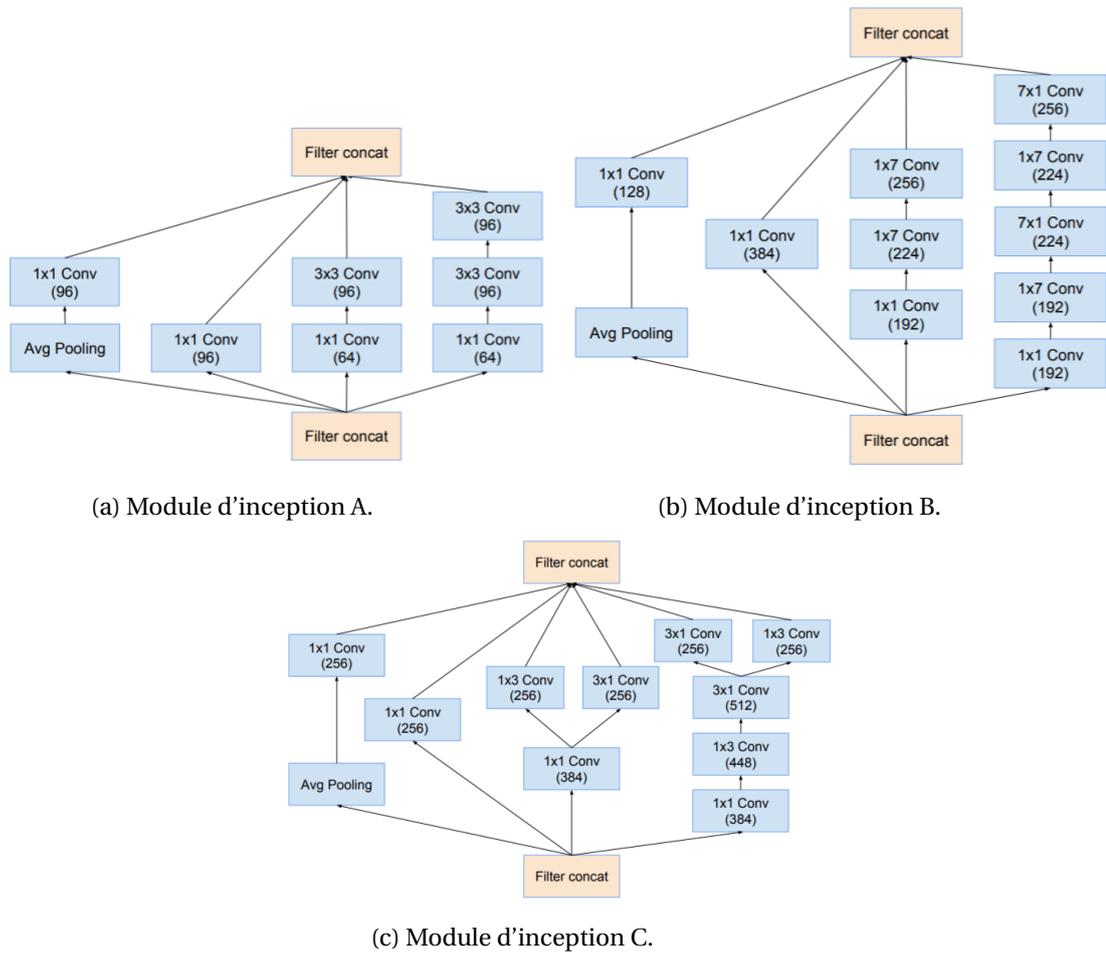


FIGURE 3.12 – Modules d'inception du réseau de neurones Inception-v4 [218].

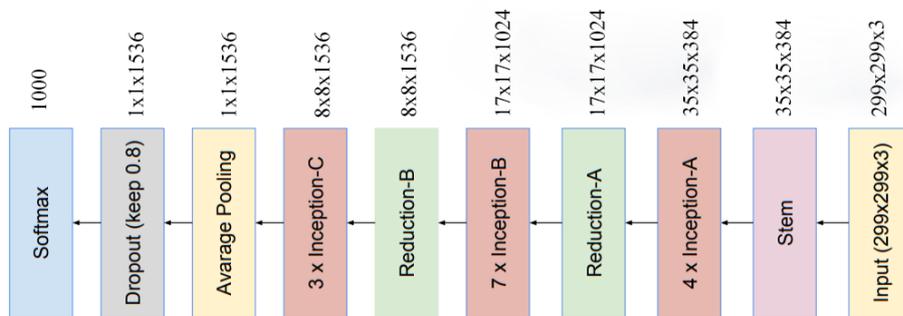


FIGURE 3.13 – Architecture du réseau Inception-v4 (adapté de [218]).

### 3.5.4 DenseNet

Dans [115], une nouvelle architecture portant le nom de DenseNet a été proposée. Les réseaux denses simplifient le modèle de connectivité entre les couches introduites dans d'autres architectures comme ResNet. En effet, leur architecture assure un flux maximum d'informations et de gradients en connectant les couches directement les unes avec les autres. De cette manière, toutes les entrées atteignent toutes les couches suivantes et les gradients se propagent facilement des couches supérieures aux couches inférieures. En se connectant de cette manière, les réseaux DenseNet nécessitent moins de paramètres qu'un CNN traditionnel équivalent, car il n'est pas nécessaire d'apprendre les cartes de caractéristiques redondantes. En outre, certaines variantes de ResNet ont prouvé que de nombreuses couches contribuent à peine et peuvent être supprimées. Au lieu de cela, les couches DenseNet sont très étroites et ajoutent simplement un petit ensemble de nouvelles cartes de caractéristiques. Ce regroupement de cartes de caractéristiques ne peut pas être fait lorsque leur taille est différente. Par conséquent, les réseaux DenseNet sont divisés en blocs denses (Figure 3.14), dans lesquels les dimensions des cartes de caractéristiques restent constantes dans un bloc, mais le nombre de filtres change entre elles. DenseNet fait la première différence avec ResNet ici. Les blocs denses n'additionnent pas les cartes de caractéristiques sortantes de la couche avec les cartes de caractéristiques entrantes, mais les concatènent (d'où le nom dense). Les couches entre ces blocs denses s'appellent des couches de transition et permettent la réduction du nombre et de la dimension spatiale des cartes de caractéristiques (division par 2) en appliquant une normalisation par lots, une convolution  $1 \times 1$  et une couche de pooling  $2 \times 2$ . Le Tableau 3.2 présente les différentes versions de DenseNet proposées par [115]. Dans notre cas, nous utilisons le réseau DenseNet-121 (Figure 3.15).

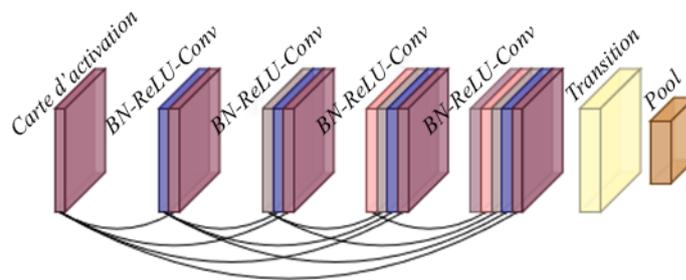


FIGURE 3.14 – Bloc convolutif dense (adapté de [11]).

Soit  $Y_0$  l'entrée du bloc dense (BD1), et  $Y_1$  sa sortie. Ce bloc contient  $l$  couches et chaque couche est caractérisée par une certaine fonction non-linéaire  $H_l$ . Cette fonction peut être une fonction composée de différentes opérations (par exemple ReLU, Pooling, convolution, etc). Soit  $y_l$  la sortie de la couche  $l$ . Comme il a été indiqué précédemment, il existe des connexions directes entre une couche et toutes les couches qui la précèdent. Autrement dit, chaque couche  $l$  reçoit les cartes de caractéristiques de toutes les couches précédentes comme des données d'entrée. Cela, peut être noté selon la relation suivante :

$$y_l = H_l([y_0, y_1, \dots, y_{l-1}]) \quad (3.29)$$

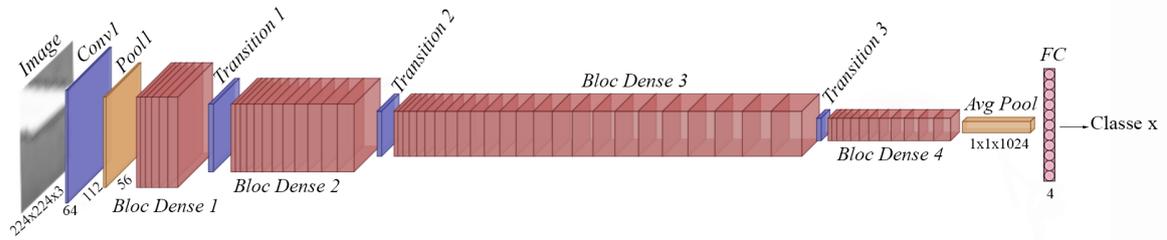


FIGURE 3.15 – Architecture du réseau DenseNet-121 pour un exemple de classification en 4 classes météorologiques (adapté de [11]).

| Nom de la couche         | Taille de sortie | DenseNet-121   | DenseNet-169   | DenseNet-201   | DenseNet-264   |
|--------------------------|------------------|--|--|--|--|
| Convolution              | 112x112          | 7x7 conv, pas 2  |  |  |  |
| Pooling                  | 56x56            | 3x3 max pooling, pas 2   |  |  |  |
| Bloc Dense (1)           | 56x56            | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$  | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$  | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$  | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$  |
| Couche de transition (1) | 56x56            | 1x1 conv   |  |  |  |
|                          | 28x28            | 2x2 moy pooling, pas 2   |  |  |  |
| Bloc Dense (2)           | 28x28            | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$ | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$ | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$ | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$ |
| Couche de transition (2) | 28x28            | 1x1 conv   |  |  |  |
|                          | 14x14            | 2x2 moy pooling, pas 2   |  |  |  |
| Bloc Dense (3)           | 14x14            | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$ | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$ | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$ | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$ |
| Couche de transition (3) | 14x14            | 1x1 conv   |  |  |  |
|                          | 7x7              | 2x2 moy pooling, pas 2   |  |  |  |
| Bloc Dense (4)           | 7x7              | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$ | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$ | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$ | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$ |
| Couche de classification | 1x1              | 7x7 moy pooling global   |  |  |  |
|                          |                  | 1000D complètement connectée, softmax  |  |  |  |

TABLEAU 3.2 – Différentes architectures de DenseNet (adapté de [115]).

où  $[y_0, y_1, \dots, y_{l-1}]$  indique la concaténation des cartes d'activations produites par les couches  $0, 1, \dots, l-1$ .

A partir de (3.29), la modélisation du bloc DB1 peut être écrite comme suit :

$$\left\{ \begin{array}{l} y_0 = Y_0 \\ y_1 = H_1(y_0) \\ y_2 = H_2([y_0, y_1]) \\ y_3 = H_3([y_0, y_1, y_2]) \\ \vdots \\ y_l = H_l([y_0, y_1, y_2, \dots, y_{l-1}]) = Y_1 \end{array} \right. \quad (3.30)$$

Prenons l'exemple du réseau DenseNet-121, à partir de la description de son architecture dans le Tableau 3.2 et de la modélisation d'un bloc dense (3.30), ce réseau peut être écrit comme suit :

$$\left\{ \begin{array}{l} x_0 = e \\ x_1 = f_{c,w}(x_0) \\ x_2 = f_{p,w}(x_1) = Y_0 \\ x_3 = Y_1 \\ x_4 = f_{c,w}(x_3) \\ x_5 = f_{p,w}(x_4) = Y_2 \\ x_6 = Y_3 \\ x_7 = f_{c,w}(x_6) \\ x_8 = f_{p,w}(x_7) = Y_4 \\ x_9 = Y_5 \\ x_{10} = f_{c,w}(x_9) \\ x_{11} = f_{p,w}(x_{10}) = Y_6 \\ x_{12} = Y_7 \\ x_{13} = f_{s,w}(x_{12}) \end{array} \right. \quad (3.31)$$

Avec

- $e$  : les données d'entrée du réseau
- $w$  : le vecteur de poids dont dépend chaque couche
- $f_{c,w}$  : la fonction exprimant la couche de convolution
- $f_{p,w}$  : la fonction exprimant la couche de pooling
- $f_{s,w}$  : la fonction exprimant la couche de sortie du réseau

# Chapitre 4

## Description des bases de données météorologiques

### Sommaire

---

|   |           |
|---|-----------|
| <b>4.1 Introduction</b> . . . . .   | <b>58</b> |
| <b>4.2 Les conditions météorologiques dégradées</b> . . . . .             | <b>58</b> |
| 4.2.1 La pluie . . . . .  | 58        |
| 4.2.2 Le brouillard . . . . .   | 59        |
| <b>4.3 Les bases de données existantes et leurs limitations</b> . . . . . | <b>61</b> |
| 4.3.1 Présentation des bases de données existantes . . . . .              | 61        |
| 4.3.2 Présentation des bases de données utilisées . . . . .               | 64        |
| 4.3.3 Limitations des bases de données . . . . .                          | 66        |
| <b>4.4 Création de la base de données</b> . . . . .                       | <b>68</b> |
| 4.4.1 Logiciel de déclenchement automatique d'acquisition . . . . .       | 69        |
| 4.4.2 Gestion des données . . . . .                                       | 72        |
| 4.4.3 La base de données mise en place . . . . .                          | 73        |
| <b>4.5 Synthèse</b> . . . . .   | <b>80</b> |

---

## **4.1 Introduction**

L'utilisation du Deep Learning pour faire la classification et la mesure des conditions météorologiques dégradées nécessite l'utilisation d'une base de données météorologique d'images. La section 4.2 de ce chapitre présente les conditions météorologiques dégradées par caméra prises en compte dans le cadre de la thèse telles que la pluie et le brouillard. Ensuite, une étude des bases de données météorologiques déjà existantes est réalisée dans la section 4.3. Enfin, la section 4.4 présente la base de données mise en place afin de réaliser l'objectif de notre étude. Cette section présente les outils utilisés pour la création de cette base de données tels que le logiciel de déclenchement automatique d'acquisition et le logiciel de gestion des données.

## **4.2 Les conditions météorologiques dégradées**

Avant la présentation de la base de données météorologiques utilisée, il est nécessaire de présenter les conditions météorologiques qui participent à la construction de cette base. Dans cette section, nous allons identifier les différents types de conditions météorologiques et les paramètres physiques de la météorologie pris en considération dans notre étude.

La météorologie est la science qui étudie les phénomènes atmosphériques et qui sert à établir des prévisions. Parmi les conditions atmosphériques, il y a la température, le vent, les nuages et les précipitations. Ces conditions peuvent varier considérablement et dépendent en grande partie du climat, des saisons et de divers autres facteurs.

Des paramètres macroscopiques et microphysiques sont associés à ces conditions atmosphériques. Les paramètres macroscopiques sont la visibilité du brouillard et l'intensité de la pluie, et les paramètres microscopiques comprennent la taille et le diamètre des gouttes. Tous ces paramètres ont une influence sur le fonctionnement des systèmes de perception basés sur des caméras.

Dans le cas de la thèse, nous nous intéressons aux paramètres macroscopiques, nous nous intéressons aussi aux précipitations parce qu'elles agissent sur la visibilité. Certains types de précipitations sont associés aux conditions météorologiques estivales et d'autres, aux conditions météorologiques hivernales. La forme que prend la précipitation lorsqu'elle tombe à la surface de la terre dépend de la température de l'air dans les nuages et au niveau du sol. Selon [1], toutes les précipitations hivernales commencent sous forme de vapeur d'eau condensée dans les nuages, quelle que soit leur forme au sol. Associés à l'hiver, la neige et la neige roulée tombent sous forme de précipitation gelée. Les précipitations estivales comprennent la pluie et la grêle. D'autres formes, telles que la bruine, le grésil et le brouillard, ne sont pas nécessairement relatives à la saison. Le Tableau 4.1 présente une comparaison entre quelques types de précipitations.

Dans le cadre de la thèse, nous n'allons prendre en considération que le brouillard et la pluie. Le cas de la neige ne sera pas abordé.

### **4.2.1 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

| 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                         |

TABLEAU 4.1 – Types des précipitations en fonction de leur diamètre et leur vitesse de chute.

Terre. La pluie se produit lorsque la température des nuages et la température du sol sont au-dessus de zéro. Elle peut prendre trois formes. On parle simplement de pluie lorsque les gouttes mesurent environ 0,5 mm de diamètre, de la bruine lorsque les gouttes sont plus petites que cela et de la virga lorsque les gouttes sont si petites qu'elles n'atteignent pas le sol.

La pluie est caractérisée d'un point de vue microscopique (taille des gouttes) et d'un point de vue macroscopique (intensité de pluie – ou rainfall rate – ( $R_R$ )). Dans la construction de la base météorologique, seule l'intensité de la pluie va être prise en considération pour caractériser les conditions de pluie. Il s'agit d'une mesure physique de l'intensité des précipitations en calculant la quantité de pluie qui tombe pendant un intervalle de temps donné sur une surface donnée. L'intensité de pluie est notée  $I_R$  et elle est exprimée en  $mm.h^{-1}$  ( $1mm.h^{-1} = 1L.m^{-2}.h^{-1}$ ). Cette intensité varie entre  $0mm.h^{-1}$  au cas où il n'y a pas de pluie à plusieurs dizaines de  $mm.h^{-1}$  pour une pluie forte. D'après la norme française NF P 99-320 [5] définissant les perturbations météorologiques routières, il existe différents types de pluie classés selon leur intensité (Tableau 4.2). Ces intensités de pluie correspondent à des valeurs couramment rencontrées en France métropolitaine.

| 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                                    |

TABLEAU 4.2 – Différentes classes de pluie selon la norme NF P 99-320 [5].

Pour la construction de la base de données, la pluie a été caractérisée en utilisant quelques capteurs dédiés à la mesure physique des conditions météorologiques et qui sont installés sur le site autoroutier du col de la Fageole (au bord de l'autoroute A75), tels que le capteur temps présent VAISALA PWD12 et le disdromètre optique OTT Parsivel. Les caractéristiques de ces capteurs seront décrites dans la suite.

#### 4.2.2 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. Le brouillard est physiquement caractérisé d'un point de vue microscopique (distribution de la taille des gouttes) et d'un point de vue macroscopique (visibilité à travers une épaisseur de brouillard)[74]. Seule la visibilité va être prise en considération lors de la construction de la base de données pour caractériser le brouillard. En effet, les caméras classiques que nous utilisons ne permettent pas d'atteindre la précision du micron dans l'image.

Le brouillard est caractérisé par une grandeur physique appelée la visibilité météorologique, notée  $V_{met}$  et exprimée en mètres. Cette dernière est définie par la loi de Koschmieder. La loi de Koschmieder décrit l'atténuation de la luminance dans une atmosphère contenant du brouillard [104, 174]. Elle propose une équation (4.1) reliant la luminance apparente  $L$  d'un objet situé à la distance  $D$  de l'observateur à sa luminance intrinsèque  $L_0$  :

$$L = L_0 e^{(-\beta D)} + L_\infty (1 - e^{(-\beta D)}) \quad (4.1)$$

où  $\beta$  est le coefficient d'extinction de l'atmosphère et  $L_\infty$  est la luminance atmosphérique. En présence de brouillard, elle correspond à la luminance du fond sur lequel l'objet peut être observé. En s'appuyant sur cette équation, Duntley [13, 174] a établi une loi d'atténuation de contraste (4.2), indiquant qu'un objet présentant le contraste intrinsèque  $C_0$  par rapport au fond sera perçu à la distance  $D$  avec le contraste  $C$  suivant (4.2) :

$$C = \frac{|L_0 - L_\infty|}{L_\infty} e^{(-\beta D)} = C_0 e^{(-\beta D)} \quad (4.2)$$

Le contraste  $C$  désigne le contraste apparent de l'objet à une distance  $D$  d'un objet de luminance  $L$  sur fond de ciel  $L_\infty$ . Cette expression sert de base pour la définition d'une dimension standard appelée "distance de visibilité météorologique"  $V_{met}$ , c'est-à-dire la plus grande distance à laquelle un objet noir ( $C_0 = 1$ ) d'une dimension appropriée peut être vu dans le ciel à l'horizon, avec le seuil de contraste fixé à 5% (l'œil humain n'est pas capable de distinguer l'objet du fond pour un seuil inférieur à 5%). Cette valeur est définie par la Commission Internationale de l'Eclairage ou International Commission on Illumination (CIE)[71] et l'Organisation mondiale de la météorologie – ou World Meteorological Organization – (WMO)[148]). La distance de visibilité météorologique est donc une dimension standard qui caractérise l'opacité d'une couche de brouillard. Cette définition donne l'expression (4.3) :

$$V_{met} = \frac{1}{\beta} \ln(0.05) \cong \frac{3}{\beta} \quad (4.3)$$

La visibilité météorologique varie entre 0  $m$  pour un brouillard très dense et des milliers de mètres pour une absence de brouillard. Il existe deux types de brouillard. Le brouillard météorologique qui est défini par une visibilité inférieure à 1000  $m$  et le brouillard routier qui prend lieu au cas où la visibilité est au-dessous de 400  $m$ . D'après la norme française NF P 99-320 [5], le brouillard routier est classé en 4 classes (Tableau 4.3).

Lors de la construction de la base de données, la distance de visibilité météorologique a été mesurée à partir des capteurs de mesure physique qui sont installés sur le site autoroutier du col de la Fageole (au bord de l'autoroute A75), tels que le capteur temps présent VAISALA PWD12 et le Visibilimètre VS20. Les caractéristiques de ces capteurs seront décrites dans la suite.

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

TABLEAU 4.3 – Différents types et classes de brouillard selon la norme NF P 99-320 [5].

## 4.3 Les bases de données existantes et leurs limitations

Dans le chapitre 2, nous avons parlé de différentes applications dans le contexte routier. La réalisation de ces applications exige une mise en place des bases de données d'images spécifiques pour chaque type d'applications.

### 4.3.1 Présentation des bases de données existantes

Parmi les bases de données les plus célèbres, on peut citer le référentiel de détection de route KITTI-ROAD en temps réel [81, 169] qui contient 600 images extraites de la base originale KITTI [88]. Cette base est utilisée pour la segmentation de la route et des voies [183] ainsi que pour la détection des objets pour la conduite autonome [38, 113, 222]. Dans le contexte routier, l'application la plus répandue est la détection des piétons. Pour cela, différentes bases de données ont été créées telles que la base Cerema-AWP [59] qui comprend environ 60 000 images de 10 piétons acquises dans la plateforme PAVIN Brouillard & Pluie du Cerema [52]. Aussi, les auteurs de [156] ont utilisé des images couleur et thermiques appartenant à la base KAIST [117] pour ce type de détection. Li et al. [152], de leur part, ont mis en place une base de données CUHK03 labellisée automatiquement et contenant 13 164 images de 1 360 piétons pour la détection de piétons. Il existe aussi la base Caltech Pedestrian [66, 67] sur laquelle la plupart des algorithmes de détection de piétons ont été validé [8, 113, 163, 227]. Zhang et al. [252] ont créé une base de 500 images acquises par un téléphone intelligent et annotée manuellement pour faire la détection des fissures routières.

Les informations météorologiques peuvent constituer un facteur important pour de nombreuses applications de vision par ordinateur, comme la détection et la reconnaissance d'objets ainsi que la catégorisation de scènes. Par exemple, la Figure 4.1 montre que la Tour Eiffel présente des apparences visuelles, radicalement différentes selon les conditions météorologiques, ce qui entraîne des difficultés considérables pour la reconnaissance des objets. Une fois que les caractéristiques météorologiques peuvent être estimées, un détecteur d'objet peut adapter ses paramètres à différents types de temps afin de réduire l'influence des variations visuelles [45]. Pour cela, nous avons besoin de bases de données d'images météorologiques.

Les bases de données disponibles publiquement pour la classification météorologique sont limitées aux niveaux d'échelle et de quantité [153]. La base de données fournie dans [196] inclut les conditions météorologiques de temps clair, pluie faible et pluie forte. Ces images sont orientées vers la classification de pluie, et ont été capturées par une caméra

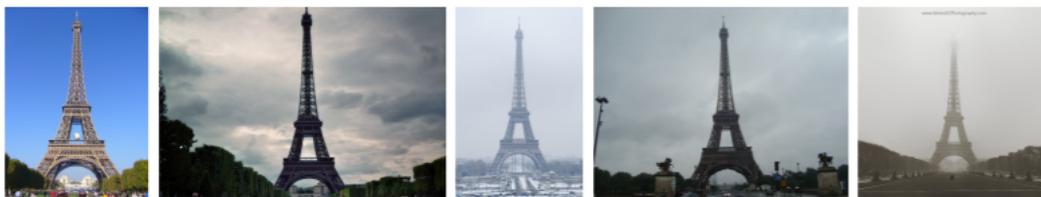


FIGURE 4.1 – La Tour Eiffel sous différentes conditions météorologiques [45].

embarquée. D'autres travaux se sont concentrés sur la classification de la pluie, tels que [75] où la base Cerema-AWR a été mise en place en contenant 30 400 images divisées en deux types de pluie; naturelle et simulée pour différents paramètres de caméras. Comme la pluie, les auteurs de [247] se sont concentrés sur l'étude des différents niveaux de brouillard (temps clair, brouillard faible et brouillard fort) en utilisant une base de données constituée de 321 images collectées à partir de six caméras installées sur trois stations météorologiques différentes situées autour de la côte d'Irlande. Les images sont prises à partir de plates-formes statiques et fixes.

Lu et al. [160] ont fourni une base de données de classification météorologique avec des scènes plus naturelles. Mais seules les images ensoleillées et nuageuses sont impliquées. C'est la base weather dataset qui contient des images météorologiques comprenant 10 000 images divisées en deux classes. Les images ont été collectées à partir de trois bases de données existantes Sun Dataset [241], Labelme Dataset [200] et Flickr. Zhang et al. [256, 257] ont décrit une base de données multi-classes contenant quatre classes de conditions météorologiques appelée MWI (Multi-class Weather Image) et constituée de 20 000 images collectées à partir des albums web et des films comme Flickr, Picasa, MojiWeather, Poco, Fengniao. Il s'agit des images avec différents fonds d'écran et avec une labellisation manuelle.

Dans [153], les auteurs ont construit une nouvelle base qui comprend 65 000 images collectées de Flickr et divisées en six classes météorologiques : temps ensoleillé, nuageux, pluvieux, neigeux, brumeux et orageux. Dans [45] aussi, les auteurs ont créé une grosse base de 183 000 images réparties en cinq classes soleil, nuage, neige, pluie, brouillard et qui s'appelle Image2Weather dataset. Il existe une autre base appelée Cerema-AWP [59] qui contient différentes conditions météorologiques dégradées, tels que le brouillard et la pluie avec deux intensités différentes chacun. Ces conditions sont capturées pendant le jour et pendant la nuit ce qui permet de détecter les piétons sous ces conditions dégradées.

Jacobs et ses collègues [123] ont initié un projet de collecte d'images de scènes en extérieur capturées par des webcams statiques sur une longue période. Les images collectées forment la base de données Archive of Many Outdoor Scene (AMOS) [123]. Dans [122], ils ont augmenté la base AMOS avec un alignement automatique de la scène et une labellisation des objets. La base de données AMOS + C [120] a été proposée comme le premier jeu de données d'image à grande échelle associé aux informations météorologiques, qui a été étendu sur la base AMOS en collectant des données météorologiques via l'horodatage et la géolocalisation des images. Avec la base de données AMOS + C, Jacobs et ses collègues ont exploré les relations entre l'apparence de l'image, la position du soleil et les conditions météorologiques [120]. Beaucoup de travaux se sont basés sur AMOS pour la création de leurs propres bases de données tels que [139] et [44].

## CHAPITRE 4. DESCRIPTION DES BASES DE DONNÉES MÉTÉOROLOGIQUES

Dans [139], la base de données créée contient 8 571 images provenant de 101 webcams. À partir d'elle, les auteurs arrivent à prédire la saison, l'heure ainsi que les conditions météorologiques. La base de données [91] comprend des images capturées en continu par 10 caméras dans 10 environnements différents pendant deux années consécutives. Une seule image est capturée à 11 h chaque jour. À partir de cette base, et en se basant sur les images de AMOS, [44] ont construit une base contenant 53 000 images capturées par neuf caméras.

Les auteurs de [175] ont introduit la base SkyFinder qui contient des images étiquetées et acquises à partir de 53 webcams dans une large gamme de conditions météorologiques et d'éclairage. Cette base contient environ 90 000 images d'extérieur. Pour la prédiction de la température, [44] n'ont utilisé que 35 417 images capturées par 44 caméras à 11 h chaque jour. Cette sélection réduit les variations de la lumière du soleil dues à la capture à différents moments de la journée. Un récapitulatif des différentes bases de données est présenté dans le Tableau 4.4.

| Base de données                        | Nb d'images | Résolution                | Annotation  | Point de vue | Conditions météorologiques                                       | Applications                        |
|--|-------------|---------------------------|-------------|--------------|--|-------------------------------------|
| KITTI [88]                             | > 15 000    | 1392 x 512                | -           | Dynamique    | Temps clair  | Détection de la route et des objets |
| Cerema-AWP [59]                        | 60 000      | 1024 x 631                | Automatique | Statique     | Temps clair, brouillard, pluie, jour et nuit                     | Détection des piétons               |
| KAIST [117]                            | 95 000      | 640 x 480                 | Manuelle    | Dynamique    | Temps clair jour et nuit   | Détection des piétons               |
| CHUK03 [152]                           | 13 000      | -                         | Automatique | Statique     | x  | Détection des piétons               |
| Caltech [66][67]                       | 250 000     | 640 x 480                 | Automatique | Dynamique    | Temps clair  | Détection des piétons               |
| [252]                                  | 500         | 3264 x 2448               | Manuelle    | Statique     | Temps clair  | Détection des fissures routières    |
| [196]                                  | 500 000     | -                         | -           | Dynamique    | Temps clair, pluie faible et pluie forte                         | Classification du temps             |
| Weather dataset [160]                  | 10 000      | Entre 600 et 1 500        | Manuelle    | Statique     | Temps ensoleillé et nuageux                                      | Classification du temps             |
| MWI [256][257]                         | 20 000      | -                         | Manuelle    | Statique     | Temps clair, pluie, nuage, brume                                 | Classification du temps             |
| Multi-classe Weather dataset [153]     | 65 000      | -                         | Manuelle    | Statique     | Temps ensoleillé, nuageux, pluvieux, neigeux, brumeux et orageux | Classification du temps             |
| Cerema-AWR [75]                        | 30 000      | 1024 x 768                | Automatique | Statique     | Temps clair et pluie   | Classification du temps             |
| Image2Weather [45]                     | 183 000     | -                         | -           | Dynamique    | Temps ensoleillé, nuageux, pluvieux, neigeux et brouillard       | Classification du temps             |
| The Transient Attribute Database [139] | 8 500       | 640 x 360 et 4000 x 3 000 | Manuelle    | Dynamique    | Temps ensoleillé, chaud, humide, brumeux et nuageux              | Classification du temps             |
| AMOS [123]                             | 100M        | 640 x 480 ou plus         | x           | Statique     | x  | Classification du temps             |

TABLEAU 4.4 – Caractéristiques des bases de données existantes.

D'après le Tableau 4.4, les différentes bases de données météorologiques présentent un manque au niveau de l'annotation des données météo. En effet, la plupart de ces bases sont classifiées selon des conditions météorologiques dégradées, mais cette classification n'est pas fine (pas d'indication des niveaux des intensités des conditions). Aussi, pour les bases qui présentent différentes conditions météorologiques, l'annotation est manuelle et faite par des volontaires. Vu ces contraintes, nous avons mis en place au Cerema une nouvelle base de données d'images météorologiques annotée automatiquement en se basant sur des capteurs dédiés à la mesure physique pour une annotation fiable. Avant de présenter la base de données que nous avons créée, nous décrivons dans la partie suivante les bases de données utilisées dans le cadre de cette thèse.

### 4.3.2 Présentation des bases de données utilisées

À partir du Tableau 4.4, une sélection de trois bases de données publiques a été faite afin de les utiliser dans la suite de cette étude.

#### La base Cerema-AWP

La base de données Cerema-AWP (Cerema Adverse Weather Pedestrian) [59] était initialement utilisée pour la détection des piétons. Mais comme elle contient différentes conditions météorologiques, nous en tenons compte dans notre étude. Il s'agit d'une base météorologique d'images composée de 62 828 images acquises sur la plateforme PAVIN Brouillard & Pluie du Cerema [52] C'est le seul endroit en Europe à produire des conditions météorologiques défavorables contrôlées telles que la pluie et le brouillard. Cette plateforme est équipée de plusieurs capteurs dédiés à la mesure physique. La distance de visibilité météorologique est mesurée avec le transmissiomètre Degreane Horizon TR30. Les intensités de pluie sont mesurées à l'aide du pluviomètre à augets LSI DQA136 et du disdromètre optique OTT Parsivel. La base de données a été construite à l'aide d'une caméra Sony DFW-X700 avec une résolution de  $1024 \times 631$ .

Cerema-AWP est divisée en 10 sous-bases acquises pendant le jour et pendant la nuit. Pour ces deux conditions, il existe 2 intensités de pluie et 2 densités de brouillard (Figure 4.2). Pour les brouillards forts (JB1) et les brouillards faibles (JB2), la visibilité est comprise entre  $50m$  et  $60m$  et entre  $70m$  et  $80m$  respectivement. Pour les pluies fortes (JP1) et les pluies faibles (JP2), l'intensité des précipitations varie de  $40mm.h^{-1}$  à  $50mm.h^{-1}$  et de  $20mm.h^{-1}$  à  $30mm.h^{-1}$ . La distribution des intensités de pluie et de brouillard est la même pour toutes les conditions. Cette base est annotée automatiquement : chaque image est associée aux données météorologiques qui lui correspondent. Elle est disponible gratuitement sur le site <https://ceremadlcfmds.wixsite.com/cerema-databases>.



FIGURE 4.2 – Exemple des conditions météorologiques présentes dans la base Cerema-AWP. Les conditions de jour en haut et les conditions de nuit en bas. De gauche à droite : condition normale, brouillard faible, brouillard fort, pluie faible, pluie forte.

#### La base Multi-classe Weather

La base Multi-classe Weather, notée MWD [153] a été créée suite à la collecte des images par 12 personnes à partir de Flickr et Google en utilisant les mots clés "outdoor" et "weather". Elle contient 65000 images dont 60000 sont divisées sur six classes météorologiques : temps ensoleillé, nuageux, pluvieux, neigeux, brumeux et orageux.

L'annotation des images est manuelle et il est demandé aux annotateurs de conserver les images contenant des scènes extérieures et avec des résolutions raisonnables. Les images visuellement similaires ont été rejetées en calculant les distances de l'histogramme de couleur pour toutes les paires d'images.

La base MWD a été divisée en deux groupes. Des images annotées avec des classes et d'autres avec des attributs météorologiques.

Le premier groupe a été utilisé pour la classification météorologique, où les conditions météorologiques critiques d'une image sont identifiées. Dans ce groupe, chaque image devrait appartenir à une seule classe. Pour cela, les images qui ne contiennent pas de gouttes de pluie apparentes, de neige ou de brouillard sont classées soit dans ensoleillés soit dans nuageux. Par contre, les images avec pluie, neige et orage sont affectées aux conditions météorologiques défavorables. Les images étiquetées comme "météo ambiguë" par une personne ou par une autre ont été rejetées. Au final, 10000 images de chaque classe ont été retenues (Figure 4.3a).

Le deuxième groupe est annoté pour prendre en considération la reconnaissance des attributs météorologiques, qui vise à trouver toutes les conditions météorologiques dans une image. En comparaison avec la classification, la reconnaissance d'attribut permet une reconnaissance plus flexible des conditions météorologiques. Les images appartenant à ce groupe sont susceptibles de présenter deux conditions météorologiques ou plus, car elles créent une ambiguïté lors de l'annotation de classe. Après l'annotation, si une image est étiquetée avec au moins un attribut ambigu, elle est éliminée. À la fin, 5000 ont été sélectionnées pour ce groupe (Figure 4.3b).

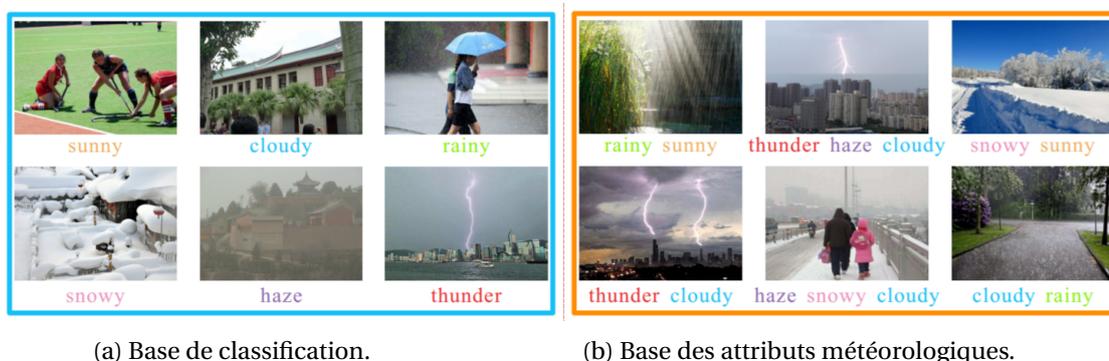


FIGURE 4.3 – Exemples d'images de la base MWD [153].

### La base Image2Weather

Image2Weather, notée I2W, [45] est une base de données météorologique contenant 183 798 images dont 119 141 images sont classées selon 5 classes : ensoleillé, nuageux, pluvieux, brouillard et neigeux. Par rapport aux bases de données existantes (Tableau 4.4), cette base contient une collection d'images à grande échelle. I2W a été construite à partir d'une sélection d'images déjà existante appartenant à la base de données EC1M (*European City 1 Million*) [12]. À partir de là, les auteurs de [45] ont mis en place une procédure automatique d'annotation des images.

À partir de l'URL et l'identifiant de l'image disponibles dans EC1M, ils utilisent *Flickr* pour obtenir l'image elle-même et ses méta-données associées, telles que l'heure et l'em-

placement. En fonction de la position prise (la latitude et la longitude), les informations d'élévation correspondantes sont acquises par le biais de *Google Maps*. En fonction de la longitude et de la latitude d'une image, ils utilisent *Weather Underground* pour extraire les propriétés météorologiques correspondantes. Les informations météorologiques sur le site Web *Weather Underground* proviennent de plus de 60 000 stations météorologiques. Pour environ 80% des images de la base, la distance les séparant des stations les plus proches est inférieure à quatre kilomètres. En utilisant ces trois outils, la base I2W s'est construite avec 28 propriétés disponibles pour chaque image, dont la date et l'heure, la température, la condition météorologique, l'humidité, la visibilité, l'intensité de pluie, etc.

Par souci d'estimation météorologique, les auteurs ne se concentrent que sur les photos capturées à l'extérieur où la région du ciel occupe plus de 10% de la photo entière. Globalement, les images recueillies couvrent une majeure partie de l'Europe. La plage de température va de  $-25^{\circ}\text{C}$  à  $45^{\circ}\text{C}$  et la plage d'humidité de 4% à 67%. La Figure 4.4 illustre des images de la base I2W. Il est clair que l'estimation du type du temps présente un défi en raison des variations visuelles faibles surtout entre les classes nuageuses et pluvieuses.

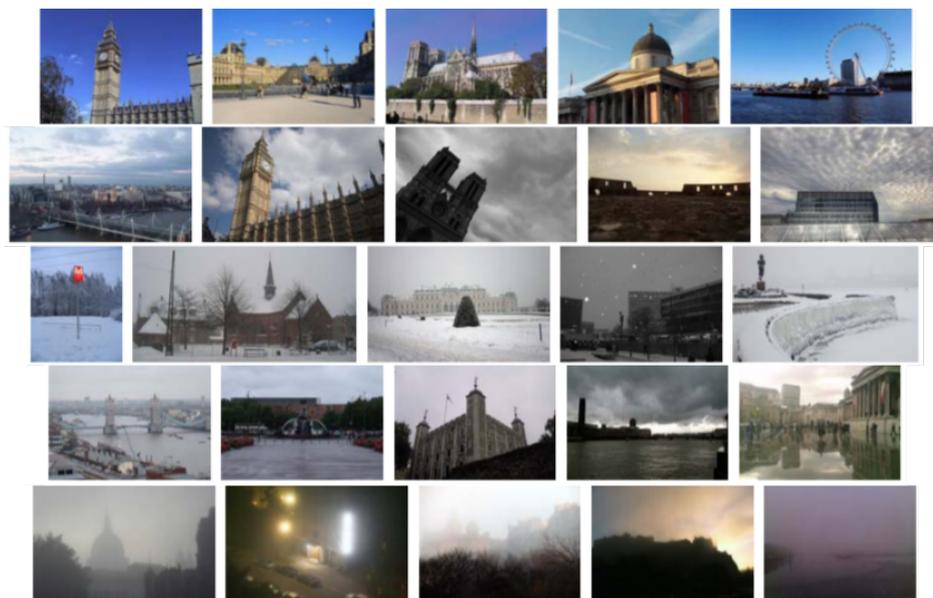


FIGURE 4.4 – Exemples d'images de la base Image2Weather [45]. Les conditions météorologiques du haut en bas : ensoleillé, nuageux, neigeux, pluvieux et brouillardoux.

### 4.3.3 Limitations des bases de données

Lors de l'étude qui a été faite sur les bases de données de l'état de l'art, nous avons rencontré beaucoup de difficultés. En effet, il existe de nombreux articles qui ne présentent pas l'approche expérimentale utilisée pour faire la classification. Ils ne sont pas assez documentés et ne contiennent pas des explications qui permettraient aux lecteurs d'utiliser les bases de données avec fiabilité. De ce fait, il devient très compliqué de se comparer avec des travaux précédents ayant des résultats dits satisfaisants.

Il existe d'autres difficultés qui sont liées aux bases de données météorologiques. Soit elles ne sont pas publiques telles que [247, 256, 257], soit, si elles sont accessibles, elles ne

disposent pas de toutes les informations qui permettent de les étudier (par exemple les conditions météorologiques, spatiales et temporelles dans lesquelles elles étaient prises). Dans ce contexte, une analyse qualitative de base de données a été présentée dans [205]. Les auteurs se sont concentrés sur l'évaluation de la qualité administrative de la base Cerema-AWP [59] afin de savoir si elle est bien documentée quand elle est diffusée sur le web, si elle dispose de toutes les informations liées à son acquisition, si sa labellisation est bien structurée, etc. Par contre, cette analyse ne discute pas du côté qualitatif/expérimental de la base liée aux données météorologiques.

Les bases de données météorologiques existantes ne sont pas utilisables rigoureusement. En effet, la plupart d'entre elles sont collectées à partir des sites web comme Google Image ou Flickr, ces images sont ensuite labellisées à la main telle que la base MWD [153]. Aussi, les images de la base I2W, collectées du web, même si elles sont labellisées automatiquement avec des informations météorologiques prises par des capteurs, celle-ci contient toujours des images ambiguës (la base disponible au public n'est pas la même que celle présentée dans [45] : les auteurs ont ajouté une 6<sup>ème</sup> classe nommée "Autres" qui contient des images présentant des doutes de classification). Les différentes étapes de filtrage ne sont pas décrites précisément et il y persiste donc des ambiguïtés. La Figure 4.5 présente quelques exemples d'images telles qu'elles sont classifiées dans les deux bases MWD et I2W. Un autre problème rencontré lié à la base I2W est le déséquilibre entre les classes météorologiques où il y a un grand écart dans le nombre des images (70 501 images pour la classe ensoleillée et 357 images pour la classe brouillard). Ce déséquilibre peut entraîner des difficultés pour le réseau de neurones lors de la phase de l'apprentissage.

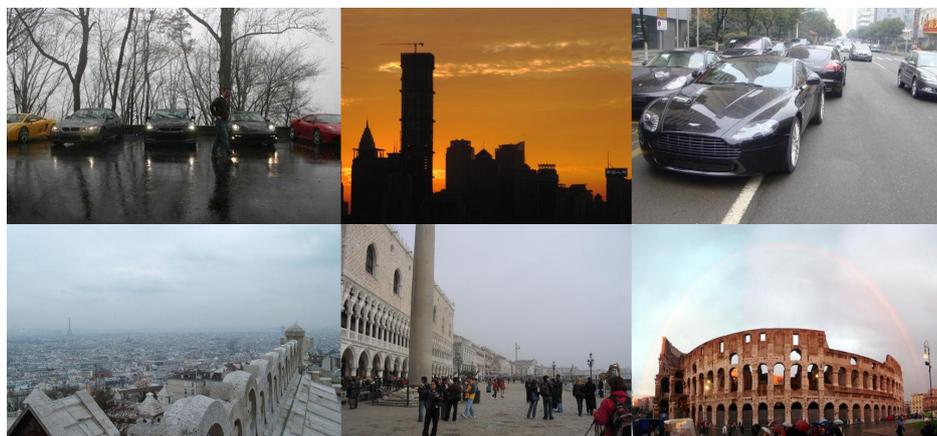


FIGURE 4.5 – Exemples d'images présentant des ambiguïtés. En haut la base MWD et en bas la base Image2Weather. De gauche à droite : ensoleillé, brouillard et pluie.

Vu ces problèmes de labellisation, et puisque la classification du temps exige au minimum des données fiables, nous avons construit une base de données rigoureuse qui se base sur des capteurs dans l'attribution des classes météorologiques installés sur le même site que la caméra d'acquisition. De cette manière, nos données garantissent une meilleure objectivité.

## 4.4 Création de la base de données

La création d'une base météorologique d'images comporte deux phases : l'acquisition des images et l'enregistrement des données météorologiques. Pour cela, nous utilisons le site autoroutier du col de la Fageole (au bord de l'autoroute A75) comme notre station météo (Figure 4.6). Ce site est équipé de deux caméras routières ainsi que d'un ensemble de capteurs météorologiques dédiés à la mesure physique. Dans le cadre de notre thèse, nous utilisons deux capteurs pour la mesure de l'intensité de pluie et deux autres pour la mesure de la distance de visibilité météorologique.



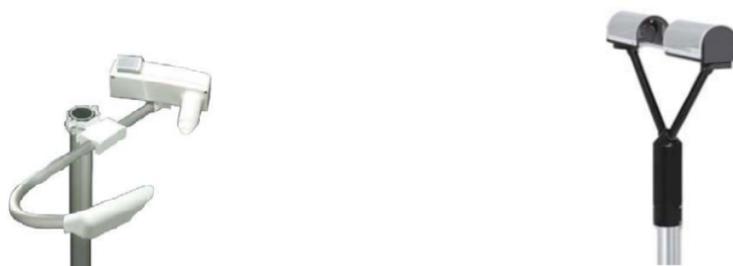
FIGURE 4.6 – Le site autoroutier du col de la Fageole.

Pour la mesure de l'intensité de pluie, nous avons utilisé le capteur temps présent VAISALA PWD12 et le disdromètre optique OTT Parsivel (Figure. 4.7). Les caractéristiques de ces capteurs sont présentées dans le Tableau 4.5.

| Capteur météorologique | $I_R$ minimale            | $I_R$ maximale             | Incertitude                   |
|------------------------|---------------------------|----------------------------|-------------------------------|
| OTT Parsivel           | $0,001 \text{ mm.h}^{-1}$ | $1200 \text{ mm.h}^{-1}$   | $\pm 5\%$                     |
| VAISALA PWD12          | $0 \text{ mm.h}^{-1}$     | $999,99 \text{ mm.h}^{-1}$ | $\leq 0,05 \text{ mm.h}^{-1}$ |

TABEAU 4.5 – Caractéristiques des capteurs de mesure de l'intensité de pluie.

Pour la mesure de la distance de visibilité météorologique, nous avons utilisé le capteur temps présent VAISALA PWD12 et le Visibilimètre VS20 (Figure. 4.8). Comme l'indique le



(a) Capteur temps présent VAISALA PWD12.

(b) Disdromètre optique OTT Parsivel.

FIGURE 4.7 – Capteurs de mesure de l'intensité de pluie.

Tableau 4.6, la plage du brouillard routier ( $V_{met}$  au-dessous de 400  $m$ ) est bien comprise dans la plage de mesure des capteurs.



(a) Capteur temps présent VAISALA PWD12.

(b) Visibilimètre VS20.

FIGURE 4.8 – Capteurs de mesure de la visibilité météorologique.

| Capteur météorologique | $V_{met}$ minimale | $V_{met}$ maximale | Incertitude |
|------------------------|--------------------|--------------------|-------------|
| Visibilimètre VS20     | 10 $m$             | 2000 $m$           | $\pm 10\%$  |
| VAISALA PWD12          | 10 $m$             | 2000 $m$           | $\pm 10\%$  |

TABLEAU 4.6 – Caractéristiques des capteurs de mesure de la visibilité météorologique.

#### 4.4.1 Logiciel de déclenchement automatique d'acquisition

L'acquisition des images et des données météorologiques se fait à partir d'un logiciel mis en place sur le site. Il est développé en C++ et utilise QT comme bibliothèque pour gérer l'interface graphique. Comme le montre la Figure 4.9, cette interface affiche les différentes informations de mesures relevées par les capteurs comme la température, l'éclairement, la visibilité, l'intensité de pluie, etc. Le déclenchement de l'acquisition des données est automatique et se fait toutes les minutes en temps réel. La Figure 4.10 montre le principe du fonctionnement du logiciel de déclenchement automatique de l'acquisition des images. Il rapporte alors quelles sont les conditions météorologiques en cours afin de valider son bon fonctionnement en continu [74].

L'enregistrement permanent des images demande un volume de stockage important. Pour cela, le logiciel a été réglé pour ne faire l'acquisition qu'en cas de besoin à travers des

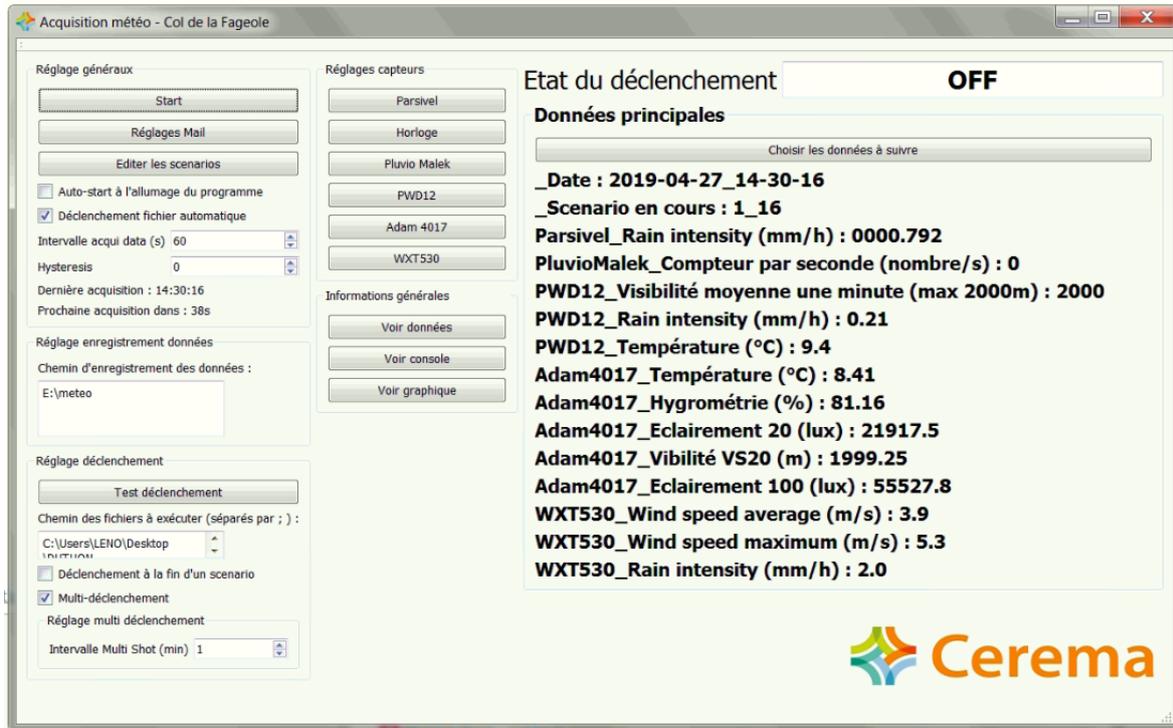


FIGURE 4.9 – Interface graphique du logiciel d'acquisition de la base météorologique d'images.

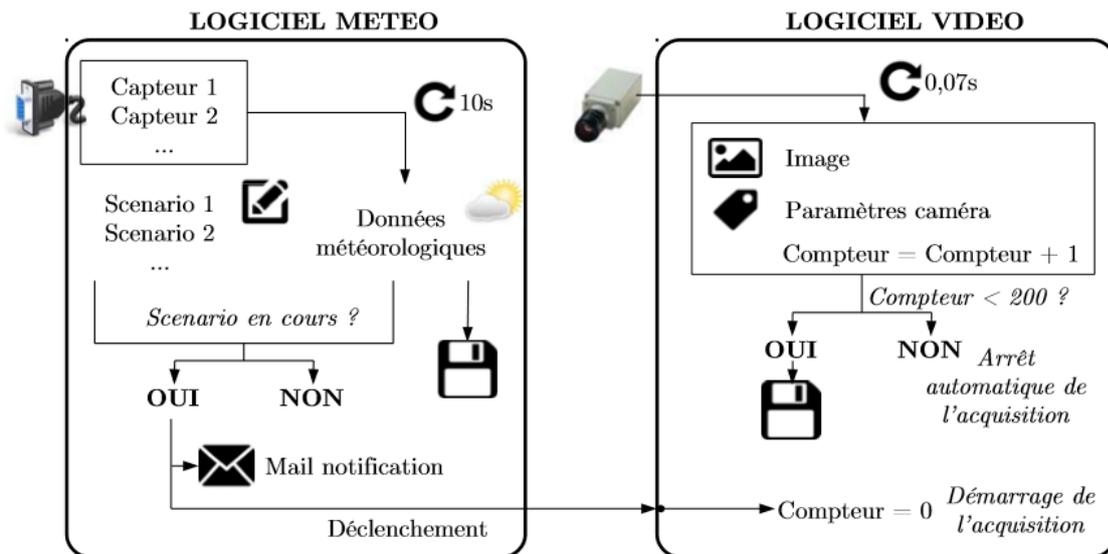


FIGURE 4.10 – Principe du fonctionnement du logiciel de déclenchement automatique de l'acquisition des images [74].

scénarios 4.11. Les scénarios dépendent des informations envoyées par l'ensemble des capteurs. Si les conditions définies pour chaque scénario sont remplies, l'acquisition se déclenche. Parmi ces conditions, il y a les grandeurs des conditions météorologiques (intensité ou distance de visibilité météorologique) ou des codes météorologiques normalisés. Il y a aussi la plage horaire à laquelle l'acquisition est permise (par exemple pour spécifier une acquisition pendant le jour ou pendant la nuit). Il existe aussi un système de comptage qui permet de limiter le nombre de déclenchements de chaque scénario à une valeur bien déterminée. Ce système permet de ne pas enregistrer beaucoup d'images inutiles dans le cas où le scénario est validé d'une manière fréquente. À noter qu'il est possible de cumuler beaucoup de conditions en un seul scénario et que le nombre de définitions des scénarios est illimité. Les images sont enregistrées d'une manière brute sans aucune compression. Cela permet un bon traitement des images indépendamment des effets de compression ou de modification.

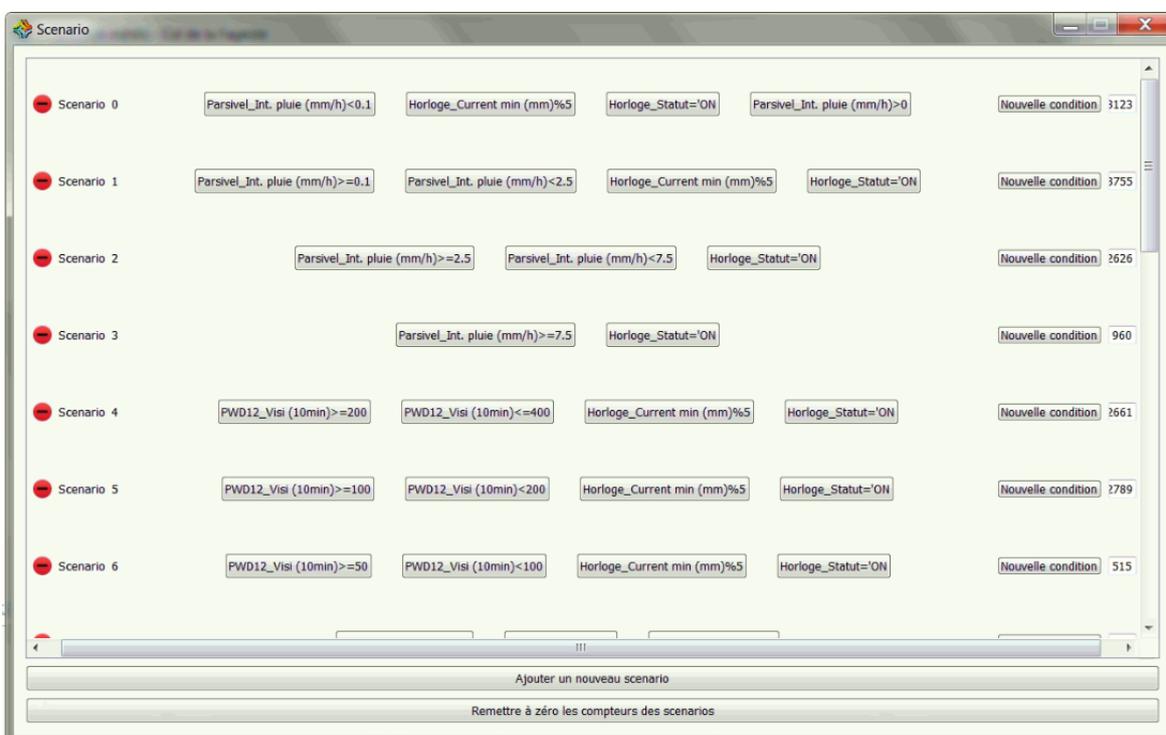


FIGURE 4.11 – Interface graphique du système d'édition des scénarios de déclenchement de l'acquisition.

Dans le cadre de la thèse, la collecte des images se fait comme suit :

- l'acquisition se fait de 7h à 2h du matin.
- un scénario pour chaque classe de pluie selon la norme française NF P 99-320 [5] (Tableau 4.2).
- un scénario pour chaque classe de brouillard selon la norme française NF P 99-320 [5] (Tableau 4.3).
- pour les scénarios qui ont des occurrences nombreuses (intensité de pluie inférieure à  $2,5 \text{ mm.h}^{-1}$  et les classes 1, 2 et 3 de brouillard), un enregistrement toutes les 5 minutes a lieu.

- pour les scénarios qui ont des occurrences très peu nombreuses (intensité de pluie supérieure à  $2,5 \text{ mm.h}^{-1}$  et les classes 4 et 5 de brouillard), un enregistrement toutes les minutes a lieu.
- dans le cas où il n'y a ni brouillard ni pluie, une acquisition régulière toutes les 10 minutes a lieu.
- les données météorologiques s'enregistrent toutes les minutes.

À chaque fois que le scénario se déclenche, un enregistrement de 100 images a lieu.

#### 4.4.2 Gestion des données

Le traitement et la manipulation des bases de données météorologiques sont des tâches complexes à gérer, vu le nombre important des données et des images. D'une part, un enregistrement des données météorologiques toutes les minutes durant 20h chaque jour, conduit à plus de 400 000 lignes de données par an. Chaque ligne de données contient plusieurs informations et valeurs acquises à partir de l'ensemble des capteurs installés sur le site. D'autre part, une acquisition de 100 images au moins toutes les 10 minutes mène à 4 000 000 d'images acquise par an, ce qui demande des ressources importantes en mémoire. Aussi, il est nécessaire d'utiliser le référencement puisqu'il est impossible de parcourir l'ensemble des images de la base. Cette méthode consiste à stocker les noms des chemins d'accès à chaque image de la base et permet d'éviter de parcourir toutes les images de la base à chaque utilisation. La construction d'une base de données météorologiques d'images consiste à associer à chaque image les données météorologiques qui lui correspondent. Pour cela, il est nécessaire d'avoir un logiciel qui permet de rassembler les images avec les données météorologiques acquises à la même minute ou à l'instant le plus proche.

Pour répondre à toutes ces exigences, un logiciel de gestion de données basé sur une base SQL a été mis en place. Avant de commencer par la gestion des données, il faut présenter l'espace de stockage de la base de données. Il s'agit d'un serveur de stockage en réseau ou *Network Attached Storage (NAS)* qui est un espace de stockage de données en commun de l'équipe. Cet espace a été mis en place pour répondre aux deux critères : l'impossibilité de stocker toutes les données localement vu leur taille importante et la nécessité d'une sauvegarde automatique des données pour assurer leur sécurité.

Revenons au logiciel de gestion de données. Ce dernier a été créé par Pierre Duthon dans le cadre de sa thèse [74]. Il est implémenté en C++ et basé sur une base SQL. L'avantage d'utiliser le SQL est la possibilité de manipuler des milliers de lignes de données ainsi que l'utilisation de requêtes pour faire le tri ou encore la recherche des données selon des critères bien déterminés. Comme il a été expliqué dans la Figure 4.12 [74], la première étape consiste à scanner et enregistrer les données dans une base SQL. Les images et les données météorologiques sont traitées indépendamment et un tableau pour chaque type de données est créé (*Image* et *Meteo*). La deuxième étape consiste à associer les données météorologiques aux images. La mise en correspondance se fait à partir de la date et l'heure d'acquisition de l'image et la date et l'heure d'acquisition de la ligne de données météorologiques la plus proche. L'identifiant de cette ligne est alors enregistré dans le tableau des données images *Image*. Finalement, le logiciel permet l'actualisation rapide des données tout en évitant la relecture des données existantes. Grâce à ce logiciel, l'accès

aux données de la base peut être fait rapidement en utilisant des requêtes et sans passer par les données brutes. Aussi, et à partir de l'identifiant extrait à la deuxième étape, il est possible de mettre en place des requêtes qui prennent en considération les métadonnées (ou paramètres de caméra) et les données météorologiques au même temps.

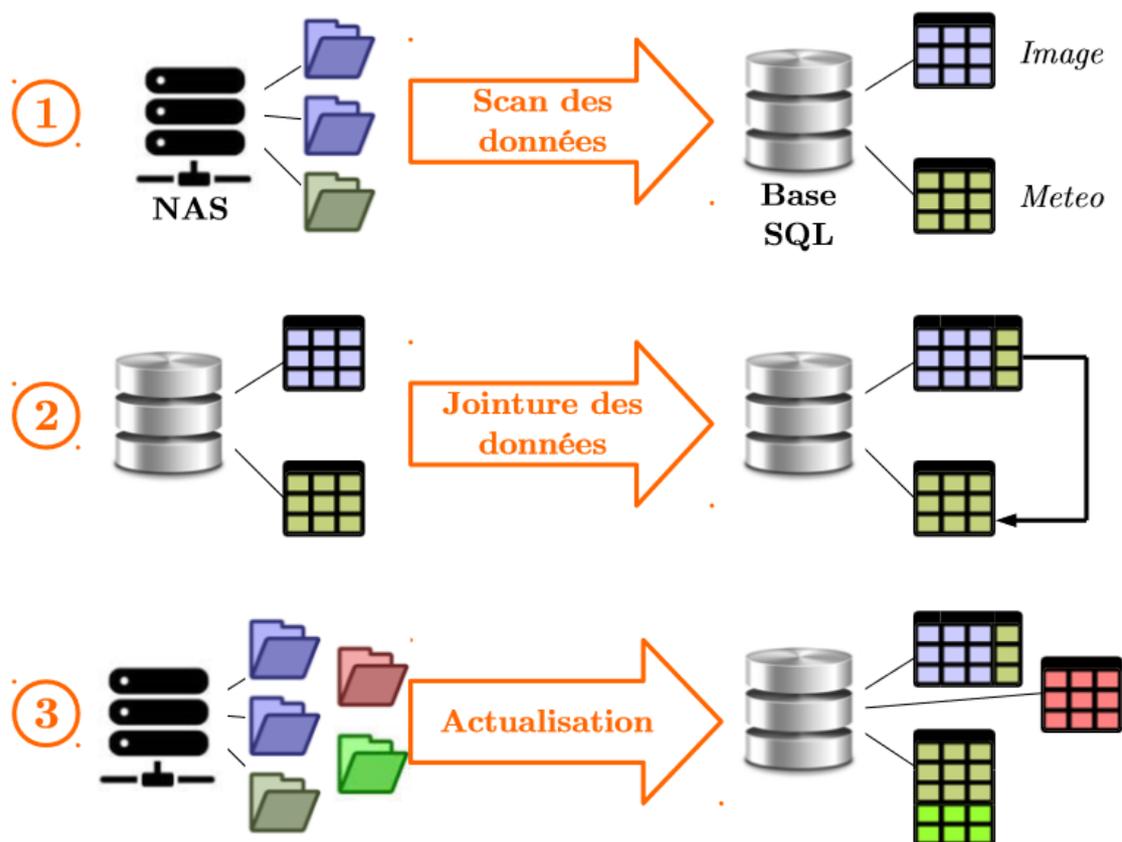


FIGURE 4.12 – Principe de fonctionnement du logiciel de gestion des données de la base [74].

#### 4.4.3 La base de données mise en place

La base de données météorologique d'images mise en place s'appelle Cerema-AWH pour *Adverse Weather Highway*. Elle est acquise au site autoroutier de la Fageole (A75) appartenant à la Direction interdépartementale des routes Massif-Central (voir Figure 4.6). La spécificité de ce site est sa localisation. En effet, il est situé à une altitude de 1 114 m. Grâce à cette altitude, des conditions météorologiques dégradées naturelles peuvent être obtenues (par exemple, la pluie très forte, brouillard fort et chutes de neige fréquentes). Cette base sera la base de tous nos apprentissages, et donc elle doit contenir tout type de conditions dégradées avec différentes intensités.

La base de données est alimentée depuis février 2017 suite à une acquisition d'images de  $1008 \times 648$  pixels avec une caméra routière de type AVT Pike F421B à une fréquence de 1,9 images par seconde. Les mêmes réglages d'optique et de caméra sont utilisés pour l'ensemble des images acquises. La caméra est fixée à une hauteur de 4,5 m du sol. Chaque

image de cette base est labellisée grâce aux données météorologiques relevées sur le site permettant de caractériser divers niveaux de précipitations pour la pluie et ainsi que pour le brouillard. En effet, l'intensité de la pluie est mesurée grâce au capteur temps présent VAISALA PWD12 et le disdromètre optique OTT Parsivel et la distance de visibilité météorologique est mesurée grâce au capteur temps présent VAISALA PWD12 et le Visibilimètre VS20. De cette façon, nous aurons l'avantage d'enregistrer les images et les conditions météorologiques au même endroit contrairement aux autres bases où l'acquisition des images se fait en un endroit et le prélèvement des données météorologiques se fait à la station météorologique la plus proche.

À l'écriture de ces lignes, la base de données comporte plus que 2 millions d'images. Il s'agit d'un nombre très important d'images qu'il est quasi impossible de toutes les utiliser durant les phases d'apprentissage, de test et de validation, car cela demanderait des ressources importantes en mémoire. Pour cela, des critères de tri ont été mis en place afin de diminuer le nombre des images ainsi que pour vérifier la base et la construire proprement. La sélection a été faite selon des critères liés aux images, comme la taille de l'image qui est fixée à  $1008 \times 648$  pixels, la luminosité (les images toutes sombres ont été éliminées), ou encore les images avec des objets indésirables (par exemple des insectes passant sur l'objectif de la caméra). Il existe d'autres critères liés à la labellisation où seulement les images avec un écart météo-image (écart entre l'heure d'acquisition de l'image et celle de la donnée météo associée) inférieur ou égal à 3 minutes sont retenues. Aussi, les images qui présentent des données météorologiques erronées suite à des problèmes de fonctionnement des capteurs ont été enlevées. Une autre sélection a été faite de telle sorte que les deux capteurs météorologiques utilisés pour labelliser les images (2 capteurs pour la pluie et 2 pour le brouillard) aient un écart relatif inférieur à 60%.

En effet, il est important de ne garder que les données météorologiques où les capteurs sont corrélés afin d'être sûr des valeurs qu'ils délivrent. Dans la construction de notre base, nous avons choisi de fixer un seuil de corrélation de 60% ce qui nous permet de garder 69% des images de brouillard et 58% des images de pluie. La Figure 4.13 montre les nuages de points obtenus pour les mesures de la visibilité enregistrées par le capteur temps présent VAISALA PWD12 et le Visibilimètre VS20 (Figure. 4.8) pour deux types de brouillard (fort et faible). De même, la Figure 4.14 montre les nuages de points obtenus pour les mesures de l'intensité de pluie enregistrées par le capteur temps présent VAISALA PWD12 et le disdromètre optique OTT Parsivel (Figure. 4.7) pour deux types de pluie (forte et faible).

Suite à ces différents tris de la base de données, beaucoup d'images ainsi que leurs données météorologiques ont été enlevées. La Figure 4.15 présente quelques exemples des images de la base Cerema-AWH. Dans le cadre de la thèse, le cas de la neige n'est pas abordé vu les ambiguïtés qui peuvent prendre place lors de sa détection (par exemple, considérer que le temps est neigeux lors de la chute de la neige seulement ou aussi lors de sa présence sur la chaussée). Cela n'empêche pas que les données météorologiques et les images de la neige sont présentes dans notre base de données brute, ce qui permettra d'effectuer des recherches et des études futures sur la neige.

Dans le cadre de la constitution de la base Cerema-AWH, 5 classes météorologiques ont été intégrées : les conditions normales de jour (JCN), brouillard fort (JB1), brouillard faible (JB2), pluie forte (JP1) et pluie faible (JP2). Comme le montre le Tableau 4.7, le brouillard routier (selon la norme NF P 99-320 [5]) a été considéré comme le brouillard fort (JB1) avec une distance de visibilité météorologique inférieure ou égale à 400m. Le brouillard

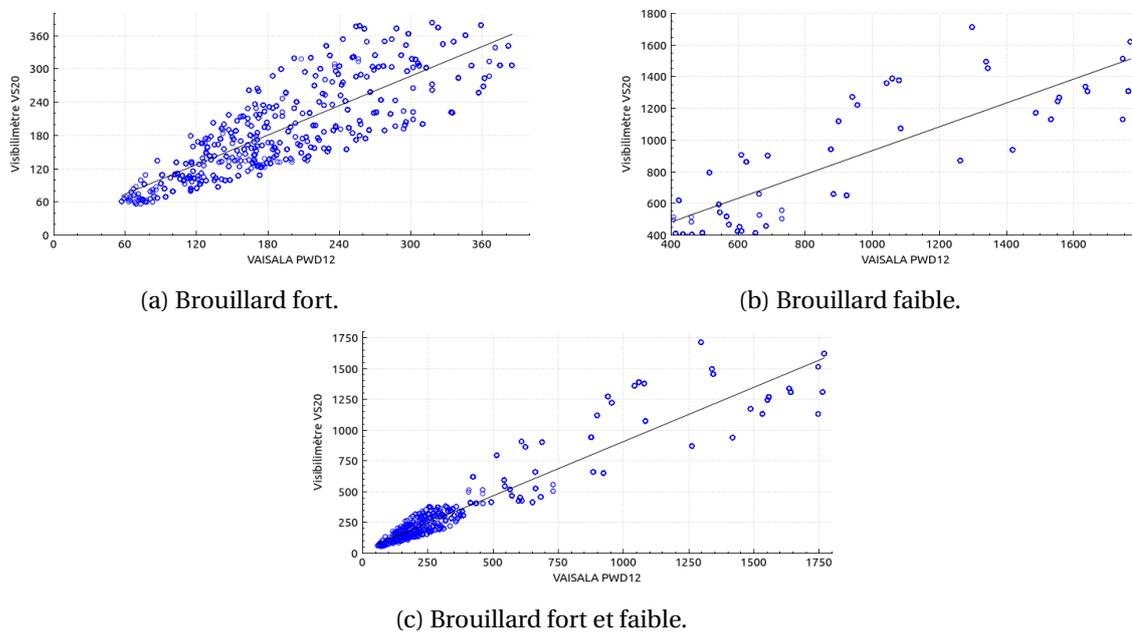


FIGURE 4.13 – Corrélation des capteurs mesurant la distance de visibilité météorologique du brouillard.

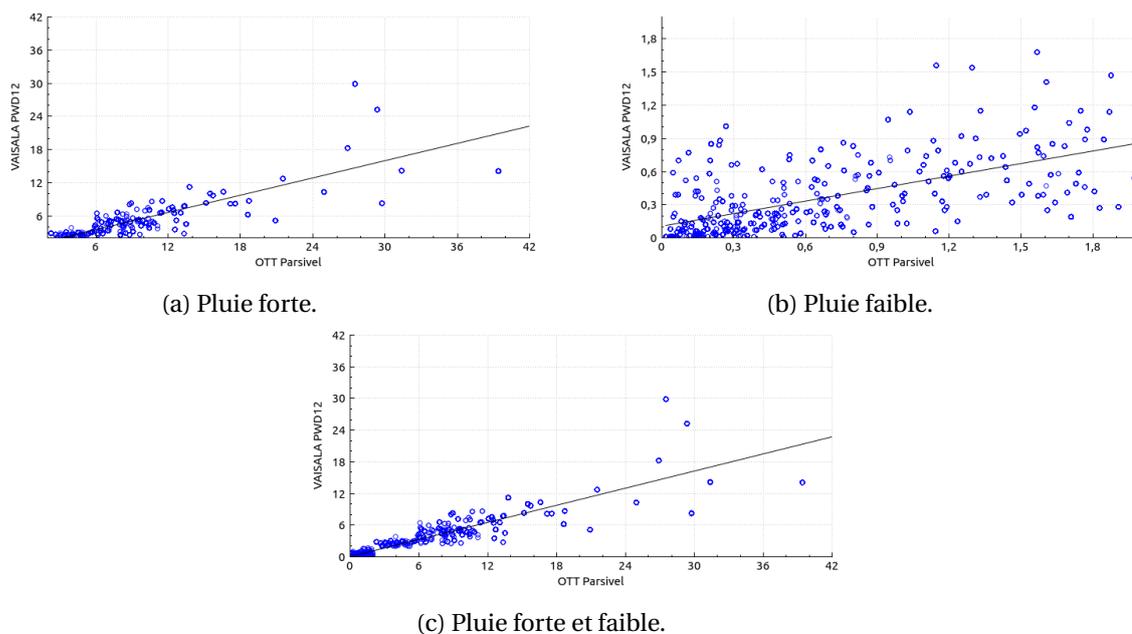


FIGURE 4.14 – Corrélation des capteurs mesurant l'intensité de la pluie.

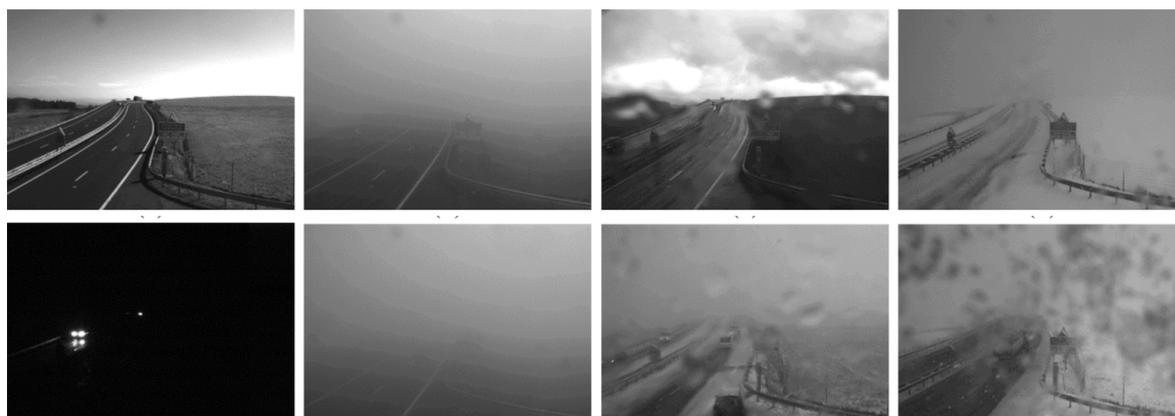


FIGURE 4.15 – Exemples d’images issues de la base de données Cerema-AWH. Les conditions météorologiques présentées de gauche à droite : conditions normales, brouillard, pluie et neige.

faible est tout brouillard présentant une distance de visibilité météorologique qui varie entre  $400m$  et  $1800m$ . La série chronologique des distances de visibilité météorologique présentes dans la base Cerema-AWH pour le cas du brouillard de jour durant la période d’acquisition (entre février 2017 et mars 2018) est présentée dans la Figure 4.16, la Figure 4.17 présentant quant à elle la distribution des distances de visibilité météorologique.

Les images dédiées à la phase d’apprentissage et celles dédiées à la phase de test sont sélectionnées selon le jour d’acquisition où les images d’apprentissage sont les images acquises pendant les jours pairs et les images de test sont les images acquises pendant les jours impairs afin d’éviter d’apprendre et de tester sur les mêmes images ce qui conduirait à biaiser le résultat de classification et de mesure.

D’après la Figure 4.17, nous remarquons que nous possédons des images qui correspondent à toutes les visibilités comprises entre  $50m$  et  $400m$  pour le brouillard fort (JB1). Par contre, dans le cas du brouillard faible (JB2), il existe un trou dans l’histogramme de distribution où la base d’apprentissage manque des images ayant des visibilités entre  $800m$  et  $1000m$ . Cela peut poser un problème lors de la prédiction de la mesure de la condition météorologique surtout que la base de test contient des images ayant cette gamme de brouillard. En revanche, ce manque de données n’influe pas dans le cas de la classification où nous nous intéressons plutôt à la classe de la condition météorologique dégradée.

Concernant la pluie, l’intensité limite divisant les deux classes a été fixée à  $2mm.h^{-1}$ . En effet, ce choix de la valeur de l’intensité de pluie revient à la quantité de données disponible sur la base de données brutes où le nombre des images avec une faible intensité de pluie est beaucoup plus grand que celui des images avec une intensité de pluie élevée. La Figure 4.18 donne une idée des intensités de pluie présentes dans la base Cerema-AWH durant la période d’acquisition et la Figure 4.19 présente la distribution des intensités de pluie selon le nombre des images et leur répartition pour la phase d’apprentissage et celle de test. Dans le cas de la pluie forte (JP1), pour la base d’apprentissage nous possédons des images contenant des pluies ayant une intensité qui atteint à  $70 mm.h^{-1}$  avec une absence de données entre  $40 mm.h^{-1}$  et  $50 mm.h^{-1}$ . Par contre, la base de test ne contient que des images contenant des intensités de pluie qui ne dépassent pas  $30 mm.h^{-1}$ . Cela peut être considéré comme un avantage qui donne à l’algorithme la capacité de prédire des intensités de pluie peu fréquentes ou rares. Nous remarquons aussi que la distribution

| Sous base | Intensité de pluie (mm/h) | Distance de visibilité météorologique (m) | Nb d'images d'apprentissage | Nb d'images de test |
|-----------|---------------------------|---|-----------------------------|---------------------|
| JCN       | $R = 0$                   | $V = 2000$                                | 66 070                      | 70 106              |
| JB1       | $R = 0$                   | $V \leq 400$                              | 23 018                      | 7 535               |
| JP1       | $R \geq 2$                | $V > 0$                                   | 6 100                       | 9 800               |
| JB2       | $R = 0$                   | $400 < V < 1800$                          | 2 500                       | 2 300               |
| JP2       | $R < 2$                   | $V > 0$                                   | 4 989                       | 8 995               |

TABEAU 4.7 – Caractéristiques de la base de données Cerema-AWH.

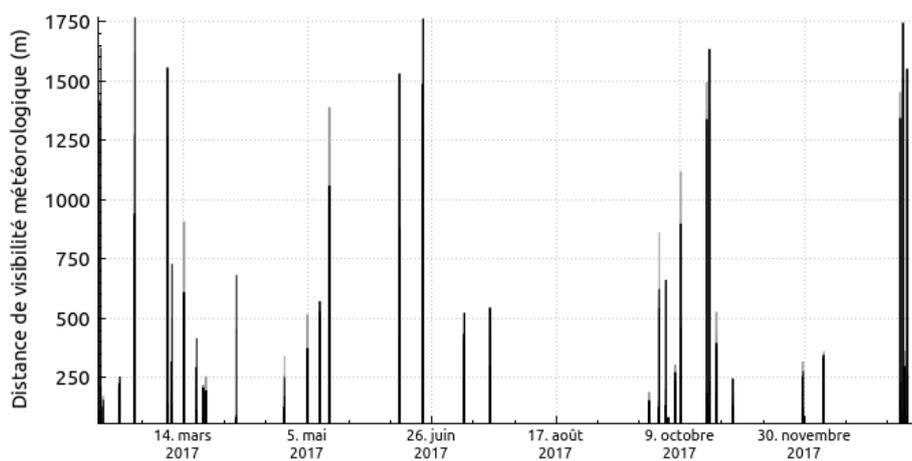


FIGURE 4.16 – La série chronologique des distances de visibilité météorologique présentes dans la base Cerema-AWH.

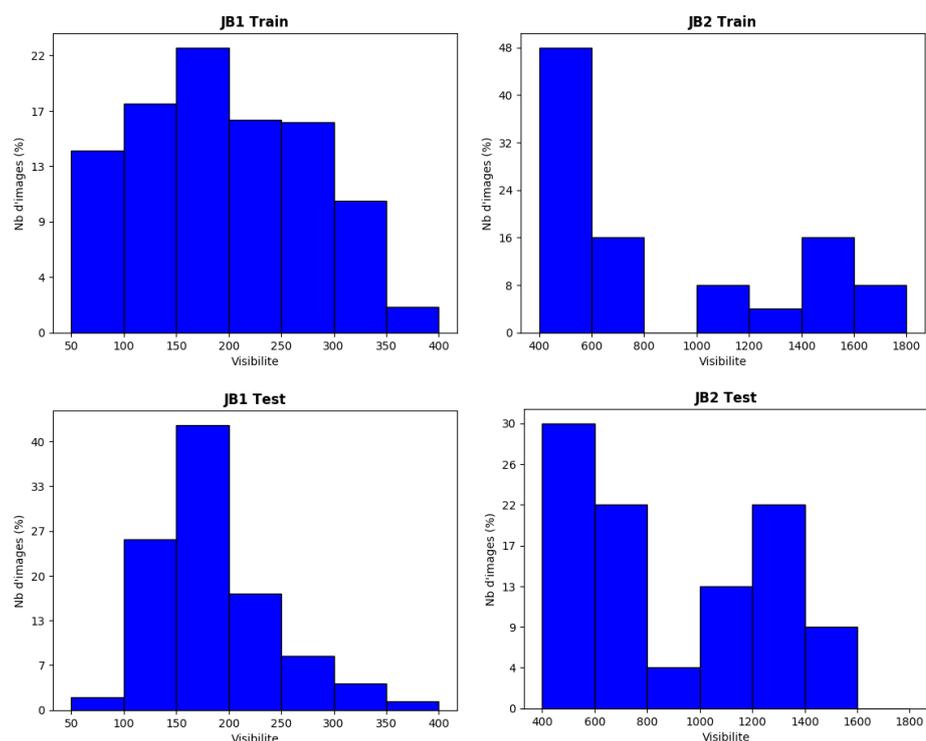


FIGURE 4.17 – Distribution de la distance de visibilité météorologique sur la base Cerema-AWH.

des intensités entre  $2 \text{ mm.h}^{-1}$  et  $3 \text{ mm.h}^{-1}$  pour les deux bases d'apprentissage et de test est à peu près identique. Pour la pluie faible (JB2), toutes les intensités de pluie inférieures à  $2 \text{ mm.h}^{-1}$  existent pour la base d'apprentissage ainsi que celle de test.

Comme l'objectif de cette thèse est de faire la classification et la mesure des conditions météorologiques dégradées, une base de mesure a été créée. Cette dernière dérive de la base décrite dans le Tableau 4.7.

Cette base s'appelle Cerema-AWH-M destinée à la mesure de la distance de visibilité météorologique du brouillard (Tableau 4.8). Ce type de base n'est pas divisé en un nombre défini de classes vu que l'objectif final n'est pas la prédiction de la classe météorologique, mais la mesure de la grandeur physique de la visibilité. Pour cela, la prise en considération de ces valeurs de visibilité est très importante.

| Sous-base | Intensité de pluie (mm/h) | Distance de visibilité météorologique (m) | Nb d'images d'apprentissage | Nb d'images de test |
|-----------|---------------------------|---|-----------------------------|---------------------|
| JB1       | R = 0                     | $V \leq 400$                              | 2 500                       | 1 500               |
| JB2       | R = 0                     | $400 < V < 1800$                          | 2 500                       | 1 500               |

TABLEAU 4.8 – Caractéristiques de la base de données Cerema-AWH-M.

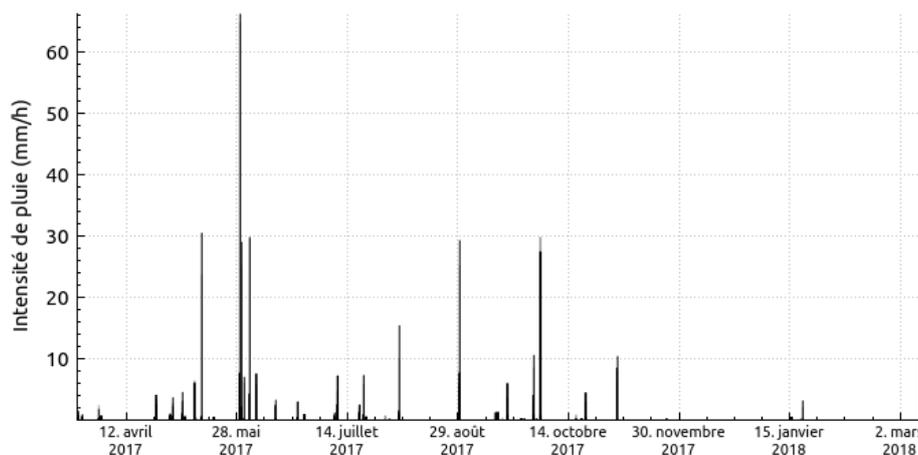


FIGURE 4.18 – La série chronologique des intensités de pluie présentes dans la base Cerema-AWH.

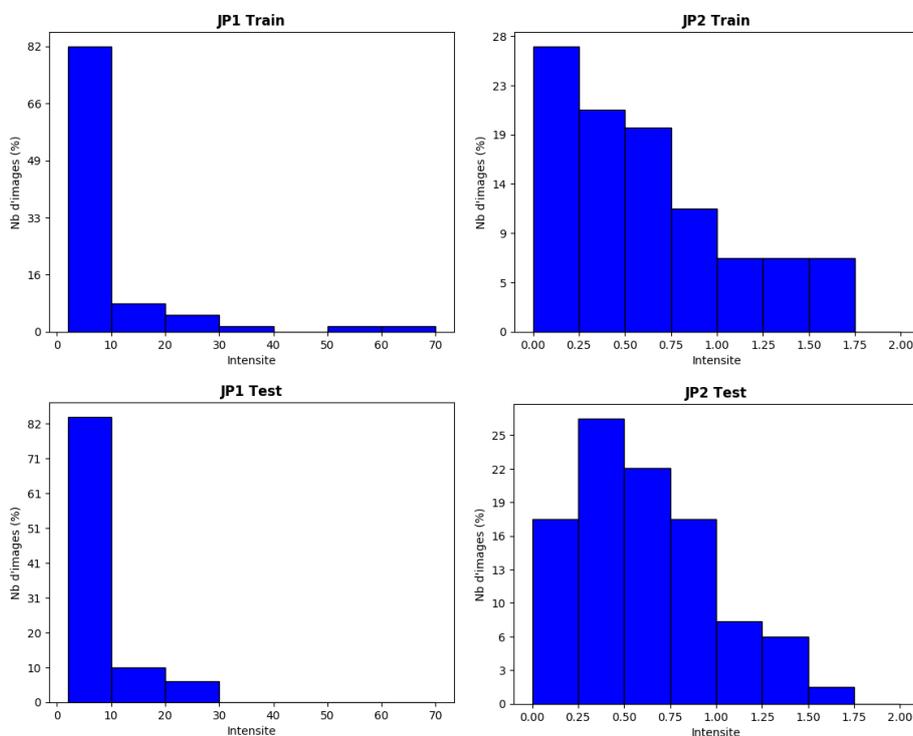


FIGURE 4.19 – Distribution de l'intensité de pluie sur la base Cerema-AWH.

## 4.5 Synthèse

Il existe des conditions météorologiques dégradées dans le contexte routier qui peuvent être critiques et qui peuvent affecter la mobilité telles que le brouillard routier avec une distance de visibilité météorologique inférieure à  $400m$  ou encore la pluie forte avec des intensités de pluie supérieures à  $7,5mm.h^{-1}$ . Afin de pouvoir faire une classification fine de ces conditions météorologiques, une base de données a été mise en place grâce au site autoroutier de la Fageole (A75). Il s'agit d'un site réel suffisamment instrumenté pour pouvoir faire l'acquisition simultanée des images et des données météorologiques. La base créée à grande échelle s'appelle Cerema-AWH et elle comprend toutes les conditions météorologiques telles que le brouillard, la pluie, la neige, le jour et la nuit. En faisant la comparaison avec les bases de données météorologiques existantes, Cerema-AWH comprend un nombre très important d'images ainsi qu'une labellisation automatique en associant chaque image aux données météorologiques correspondantes. Ces données sont enregistrées à partir des capteurs météorologiques situés sur le site de l'acquisition des images. Bien que dans le cadre de la thèse certaines conditions n'étaient pas prises en compte comme la nuit et la neige, ces conditions peuvent faire l'objet de prochaines études. À noter aussi qu'il existe d'autres bases données météorologiques qui sont en cours d'acquisition comme la base acquise à Egletons et la deuxième base du site autoroutier de la Fageole.

Comme la base de données Cerema-AWH est bien construite et les réseaux de neurones sont bien décrits, le chapitre prochain présentera l'ensemble des analyses effectuées dans le cadre de ces travaux de thèse pour la classification des conditions météorologiques dégradées.

# Chapitre 5

## Classification des conditions météorologiques par caméra routière

### Sommaire

---

|  |            |
|--|------------|
| <b>5.1 Introduction</b>  | <b>82</b>  |
| <b>5.2 Justification de l'usage de réseaux de neurones convolutifs</b>                             | <b>82</b>  |
| 5.2.1 Influence du contraste dans la classification des conditions météorologiques                 | 83         |
| 5.2.2 Sélection des hyper-paramètres optimaux  | 85         |
| 5.2.3 Influence des différentes couches d'un CNN par construction d'une architecture instrumentale | 87         |
| 5.2.4 Sélection des réseaux de neurones convolutifs  | 91         |
| <b>5.3 Influence des données d'entrée et de sortie</b>   | <b>92</b>  |
| 5.3.1 Influence du changement de scènes  | 93         |
| 5.3.2 Influence du nombre de données d'entrée  | 94         |
| 5.3.3 Influence du nombre de classes   | 95         |
| 5.3.4 Utilisation des bases de données publiques   | 96         |
| <b>5.4 Transfert de bases de données</b>   | <b>97</b>  |
| 5.4.1 Transfert entre Cerema-AWH et les bases publiques  | 99         |
| 5.4.2 Transfert entre des bases mélangées  | 99         |
| <b>5.5 Mesure de la distance de visibilité météorologique</b>                                      | <b>103</b> |
| <b>5.6 Synthèse</b>  | <b>106</b> |

---

## 5.1 Introduction

Afin d'assurer la sécurité routière et faciliter la mobilité, il est nécessaire de connaître les conditions météorologiques en temps réel. En effet, la dégradation de ces conditions peut entraîner des accidents ou encore des congestions. En contexte routier, il existe des systèmes de vision artificielle tels que les caméras de surveillance routière, les radars ou les systèmes d'aide à la conduite automobile qui peuvent être influencés par ces conditions dégradées, ce qui peut entraîner des dysfonctionnements suite à des limitations de visibilité et des réductions dans la qualité de l'image [60, 76, 196, 209, 244] car ils sont généralement développés pour un fonctionnement sous des conditions normales [196].

De nos jours, la reconnaissance des conditions météorologiques se fait à partir des stations météorologiques qui sont équipées de capteurs spécifiques dédiés à la mesure physique tels que les transmissiomètres, les pluviomètres à augets, les capteurs de température, les capteurs temps présent, etc. Le problème avec ces stations météorologiques est qu'elles sont coûteuses en termes de matériels et installations, et elles demandent des interventions de maintenance. Pour remédier à cela, une utilisation des caméras routières pour identifier les conditions météorologiques dégradées peut être une solution intéressante. En effet, ces dernières sont installées partout sur le territoire, ce qui constitue un réseau dense de capteurs pour la mesure des conditions météorologiques au contraire des stations météorologiques qui ne sont pas trop présentes sur les bords de la route. Dans [191], les auteurs ont montré que la mesure des conditions météorologiques en plusieurs points, même si elle est imprécise, est plus utile qu'une mesure précise à des points isolés. Aussi, le fait de passer par les caméras routières réduit les coûts, car les caméras sont déjà installées et la seule intervention qui pourra prendre place est une intervention logicielle. Un autre avantage peut être cité, les caméras capables de mesurer les conditions météorologiques seront aussi capables de fonctionner en présence de conditions défavorables et donc elles parviendront à exécuter leurs fonctions finales comme la détection des incidents ou des objets sur la route.

Dans la section 5.2 de ce chapitre, nous présentons une étude "académique" permettant d'étudier les différentes caractéristiques des réseaux de neurones convolutifs et de justifier leur utilisation dans notre contexte. Ensuite, dans la section 5.3, nous présentons une analyse spécifique sur les données d'entrée et de sortie de notre système de classification météorologique. Dans la section 5.4, nous présentons la méthode de transfert de bases que nous avons adoptée pour étudier la portabilité de notre système. Et finalement, nous présentons une approche de mesure de la distance de visibilité météorologique dans la section 5.5.

## 5.2 Justification de l'usage de réseaux de neurones convolutifs

Nous souhaitons étudier dans cette section l'influence de différents composants d'un réseau de neurones sur la classification du temps météorologique. Pour ce faire, nous construisons un réseau de neurones instrumental, appelé ClariNet, qui présentera différentes versions selon qu'il contienne des couches de convolution, différents nombres de filtres de convolution et différents nombres de neurones. Nous aboutirons à la conclu-

sion de l'effet bénéfique d'introduire des couches de convolution pour la classification météorologique. À partir de l'article [23] réalisant un benchmark de différents CNN sur la classification, nous retiendrons trois réseaux (ResNet-152, DenseNet-121 et Inception-v4) pour lesquels nous comparons les scores de classification météorologique.

Nous allons utiliser le score de justesse appelé encore score de classification -ou *accuracy*- comme métrique d'évaluation de la performance du système lors de la classification. Ce score permet de calculer la proportion des prédictions correctes effectuées par le modèle. Il varie entre 0 pour un algorithme qui ne détecte jamais la bonne classe et 1 pour un algorithme qui réussit toujours à prédire la bonne classe. Ce score est calculé selon l'équation (5.1). Dans la suite, ce score est exprimé en pourcentage (%).

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

où nous notons :

- VN : vrai négatif si l'élément est correctement prédit négatif
- VP : vrai positif si l'élément est correctement prédit positif.
- FP : faux positif si l'élément est prédit comme positif alors qu'il ne l'était pas.
- FN : faux négatif si l'élément est prédit comme négatif alors qu'il ne l'était pas.

Ce score de justesse est calculé sur la base de test. Cette base est toujours différente de la base d'apprentissage au sens où les images de la base de test ne se trouvent jamais dans la base d'apprentissage même si elles sont acquises sur le même site. Cela peut apporter de la similarité dans les images. Cette différence entre la base d'apprentissage et la base de test peut être vue aussi au niveau des scènes provenant de sites différents et de caméras différentes. Dans ce cas une variabilité de scènes peut être ajoutée au système de classification météorologique. Dans ce chapitre, nous utilisons ces deux sens de "bases de données différentes" :

- différentes dans le sens où aucune image de l'une est présente dans l'autre.
- différentes dans un sens plus large : les images proviennent de "sites" différents.

### 5.2.1 Influence du contraste dans la classification des conditions météorologiques

Comme notre classification concerne le brouillard et la pluie, et comme ces deux types de conditions affectent la perte de visibilité dans l'image et notamment la perte de contraste (selon la loi d'atténuation de contraste (4.2) où la distance de visibilité météorologique intervient sur le contraste), nous avons mis en place une étude qui permet de vérifier l'influence de cette loi physique dans le domaine de traitement d'images. À noter que la loi d'atténuation de contraste a été présentée dans le cas du brouillard, mais elle peut être valable pour le cas de la pluie qui introduit une chute de visibilité et donc une chute de contraste. En la comparant avec le brouillard, cette chute est beaucoup moins forte mais elle a quand même un effet appelé effet occultant de l'atmosphère.

Dans cette étude, nous utilisons deux types de patchs appartenant à des images de la base Cerema-AWP (déjà décrite dans la section 4.3) : un patch uniforme qui ne contient qu'une zone sombre de la chaussée (Figure 5.1) et un patch contrasté qui a été choisi sur

une zone de la chaussée avec un marquage au sol (Figure 5.1). Ces patchs ont été utilisés comme entrées du réseau LeNet. Chacun d'entre eux a une dimension de  $28 \times 28$  pixels, car il s'agit de la taille historique des images de la base MNIST [65] qui a servi à l'apprentissage et au test de LeNet.

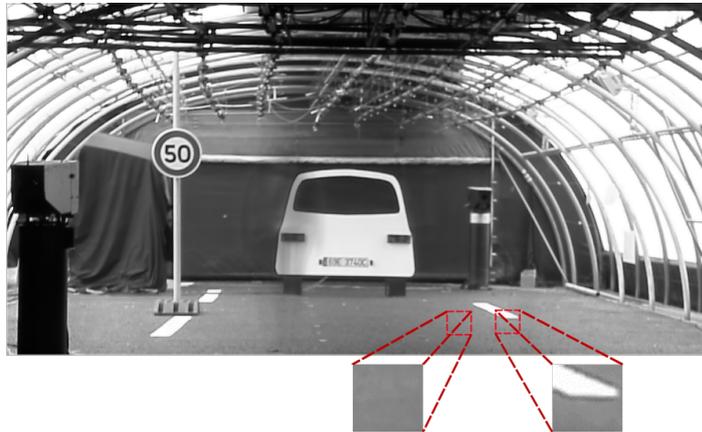


FIGURE 5.1 – Différents types de patchs de la base Cerema-AWP. Patch contrasté (à droite). Patch uniforme (à gauche).

La classification a été faite selon six classes météorologiques  $Cl = \{JCN, JB1, JP1, JB2, JP2, NCN\}$ . Pour rappel, ces conditions sont *conditions normales de jour* (JCN), *brouillard fort de jour* (JB1), *brouillard faible de jour* (JB2), *pluie forte de jour* (JP1), *pluie faible de jour* (JP2) et *conditions normales de nuit* (NCN). Après avoir appris LeNet avec les deux types de patchs, les scores de justesse suivants ont été obtenus : 65% pour les patchs uniformes et 99% pour les patchs contrastés.

Afin de bien comprendre ce résultat de classification, une analyse plus détaillée est décrite dans le Tableau 5.1. Ce dernier présente le pourcentage d'images prédites pour chaque classe par rapport à la vérité terrain. À partir de là, plusieurs remarques peuvent être tirées concernant la classification en utilisant des patchs uniformes. La classe de la nuit (NCN) est la meilleure classe prédite où l'algorithme arrive à bien classer toutes les images. Les deux classes de brouillard (JB1 et JB2) sont moins bien classifiées. En effet, l'algorithme confond le brouillard faible et le brouillard fort vu que leurs visibilités sont proches (visibilités comprises entre 50m et 60m pour JB1 et entre 70m et 80m pour JB2), mais non avec les autres classes (JP1, JCN et NCN). Par contre, ce problème se pose essentiellement pour la classification de la pluie (JP1 et JP2) et les conditions de jour (JCN) où l'algorithme fait la confusion avec toutes les classes possibles.

Deux conclusions peuvent être tirées du Tableau 5.1 et concernent la classification du temps en se basant sur des patchs uniformes. D'une part, l'algorithme n'est pas suffisamment précis pour distinguer entre deux classes météorologiques proches telles que JB1/JB2, JP1/JP2 et JP2/JCN. D'autre part, l'algorithme a du mal à détecter les conditions de pluie et les conditions normales du jour.

En comparant ce tableau avec celui de la classification avec des patchs contrastés (Tableau 5.2), nous concluons que la classification des conditions météorologiques défavorables peut être plus efficace et performante en utilisant des patchs contrastés que des patchs uniformes comme données d'entrée. Cela confirme clairement la loi d'atténuation

|            |     | Vérité terrain |      |      |      |      |       |
|------------|-----|----------------|------|------|------|------|-------|
|            |     | JCN            | JB1  | JP1  | JB2  | JP2  | NCN   |
| Prédiction | JCN | 61,7           | 0,0  | 8,0  | 0,0  | 5,5  | 0,0   |
|            | JB1 | 7,4            | 87,2 | 9,5  | 78,2 | 2,7  | 0,0   |
|            | JP1 | 5,3            | 0,0  | 41,9 | 0,0  | 12,5 | 0,0   |
|            | JB2 | 1,6            | 12,7 | 0,6  | 21,2 | 0,9  | 0,0   |
|            | JP2 | 22,6           | 0,1  | 38,8 | 0,5  | 76,3 | 0,0   |
|            | NCN | 1,3            | 0,0  | 1,2  | 0,0  | 2,0  | 100,0 |

TABLEAU 5.1 – Résultats détaillés sur la prédiction des classes météorologiques avec des patches uniformes de la base Cerema-AWP.

de contraste (4.2). Dans les expériences suivantes nous utilisons des données d'entrée contrastées.

|            |     | Vérité terrain |      |      |      |      |      |
|------------|-----|----------------|------|------|------|------|------|
|            |     | JCN            | JB1  | JP1  | JB2  | JP2  | NCN  |
| Prédiction | JCN | 100,0          | 0,0  | 0,5  | 0,0  | 0,1  | 0,1  |
|            | JB1 | 0,0            | 98,6 | 0,0  | 0,9  | 0,0  | 0,0  |
|            | JP1 | 0,0            | 0,0  | 99,5 | 0,0  | 0,0  | 0,0  |
|            | JB2 | 0,0            | 1,4  | 0,0  | 99,1 | 0,0  | 0,0  |
|            | JP2 | 0,0            | 0,0  | 0,0  | 0,0  | 99,9 | 0,0  |
|            | NCN | 0,0            | 0,0  | 0,0  | 0,0  | 0,0  | 99,9 |

TABLEAU 5.2 – Résultats détaillés sur la prédiction des classes météorologiques avec des patches contrastés de la base Cerema-AWP.

### 5.2.2 Sélection des hyper-paramètres optimaux

Comme il a été mentionné dans la section 3.4, l'optimisation des poids d'un réseau de neurones se fait à partir d'un algorithme de rétropropagation qui se base sur la descente de gradient stochastique (voir (3.13)). Parmi les hyper-paramètres de la méthode de descente de gradient, nous trouvons le taux d'apprentissage  $\alpha_0$  et le coefficient  $\gamma$  (voir (3.17)). Entraîner le réseau de neurones avec les valeurs optimales de  $\alpha_0$  et  $\gamma$  permet d'obtenir un réseau de neurones de classification performant.

Afin de choisir l'ensemble des hyper-paramètres optimaux, nous avons lancé une série d'apprentissages en faisant varier à chaque fois le couple  $(\alpha_0, \gamma)$  avec les valeurs du taux d'apprentissage  $\alpha_0$  suivantes :  $10^{-3}$ ,  $5.10^{-3}$ ,  $10^{-2}$  et les valeurs de gamma  $\gamma$  suivantes :  $10^{-4}$ ,  $10^{-3}$ ,  $10^{-2}$ ,  $10^{-1}$ . Nous avons gardé le même réseau de neurones que dans la section précédente, à savoir LeNet pour faire les apprentissages sur la base de données Cerema-AWH avec 4 et 6 classes météorologiques sur des patches contrastés. La Figure 5.2 montre le score de classification pour chaque couple  $(\alpha_0, \gamma)$ . Pour chaque couple d'hyper-paramètres, 10 apprentissages ont été lancés et le meilleur score a été enregistré. Les courbes de la Figure 5.2 montrent que plus le taux d'apprentissage augmente, plus le score augmente. Cette augmentation de score dépend aussi de la diminution de la valeur de  $\gamma$ . Dans les deux types d'expériences, le même comportement des couples d'hyper-paramètres a été noté. En outre, nous remarquons qu'il n'est pas utile d'utiliser un taux d'apprentissage très faible ou un  $\gamma$  très élevé. À partir de cette étude, il suffit de choisir un taux d'apprentissage  $\alpha_0 \geq 5.10^{-3}$  et  $\gamma \leq 10^{-3}$  pour obtenir les meilleurs scores de classification avec le réseau de neurones LeNet.

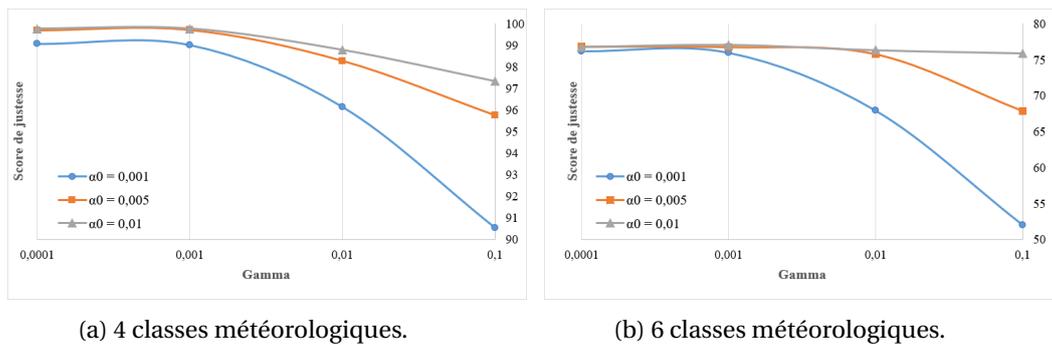


FIGURE 5.2 – Résultats d'apprentissage avec différents couples d'hyper-paramètres sur la base de données Cerema-AWH.

Ce choix a été validé par la littérature pour les réseaux de neurones ResNet-152, Denset-121 et Inception-v4 (voir Tableau 5.3) à partir des références [105, 115, 218]. Dans la suite, nous prendrons en considération ces valeurs des hyper-paramètres dans tous les apprentissages que nous effectuons.

| Architecture | $\alpha_0$       | $\gamma$       |
|--------------|------------------|----------------|
| LeNet        | $\geq 5.10^{-3}$ | $\leq 10^{-3}$ |
| ResNet-152   | $10^{-2}$        | $10^{-3}$      |
| DenseNet-121 | $5.10^{-2}$      | $10^{-3}$      |
| Inception-v4 | $10^{-3}$        | $10^{-3}$      |

TABEAU 5.3 – Hyper-paramètres utilisés pour l'apprentissage des réseaux de neurones.

Cette méthode de sélection du couple optimal d'hyper-paramètres a été adoptée pour tous les réseaux de neurones utilisés dans la suite.

### 5.2.3 Influence des différentes couches d'un CNN par construction d'une architecture instrumentale

L'objectif ici est d'étudier l'impact des couches de convolution et des couches complètement connectées ainsi que l'influence de leurs caractéristiques sur le score de classification. Le point de départ de cette étude était basé sur les travaux de [74]. En effet, l'auteur utilise un réseau de neurones simple pour faire la classification des conditions météorologiques selon 6 et 4 classes sur la base Cerema-AWP. Sa méthode est d'extraire un ensemble de descripteurs d'images "à la main" et de les utiliser en tant qu'entrées du réseau de neurones de type perceptron. Un score de justesse de 87% pour une classification de 4 classes est obtenu.

En s'inspirant de cette méthode, une première version de réseau de neurones appelée "ClariNet1" est proposée. L'architecture de **ClariNet1** est semblable à celle de [74] qui prend en entrée des données de taille  $28 \times 28$  pixels et qui contient deux couches de neurones complètement connectées (FC1 et FC2). Dans un premier temps, nous avons ajouté une couche de convolution (Conv1). Le rôle principal des couches de convolution est d'extraire automatiquement les descripteurs d'images qui serviront après à la phase d'apprentissage, et donc cette couche a été introduite afin de remplacer l'ensemble des descripteurs extraits "à la main" par [74] que sont le maximum de l'intensité de l'image, son minimum, sa moyenne et son écart type. La Figure 5.3 montre l'architecture de **ClariNet1**.

Dans notre étude, nous allons analyser l'influence du nombre de couches de convolution. Pour cela, nous avons créé un autre réseau ayant une couche de convolution de plus par rapport à **ClariNet1**. Ce réseau s'appelle **ClariNet2** dont l'architecture est présentée dans la Figure 5.4.

Afin de différencier les architectures les unes des autres, nous considérons la notation suivante (A, B, C, D) avec :

- $A \in \{3, 10, 50\}$  est le nombre de filtres sur la première couche de convolution si elle existe, sinon  $A = 0$ .
- $B \in \{3, 10, 50\}$  est le nombre de filtres sur la deuxième couche de convolution si elle existe, sinon  $B = 0$ .
- $C \in \{100, 500, 5000\}$  est le nombre de neurones sur FC1.
- $D \in \{50, 250, 2500\}$  est le nombre de neurones sur FC2.

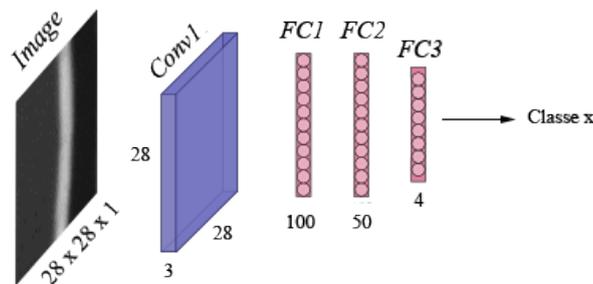


FIGURE 5.3 – Architecture du réseau ClariNet1. Cette architecture correspond à la configuration (3,0,100,50) pour un exemple de classification en 4 classes météorologiques.

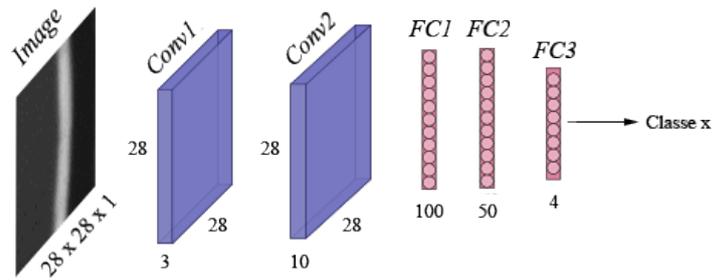


FIGURE 5.4 – Architecture du réseau ClariNet2. Cette architecture correspond à la configuration (3, 10, 100, 50) pour un exemple de classification en 4 classes météorologiques.

Tous les apprentissages de cette partie sont réalisés avec des patches contrastés de  $28 \times 28$  pixels sur la base **Cerema-AWH** avec une classification de **4 classes**  $C_l = \{JCN, JB, JP, NCN\}$ . Ces apprentissages ont été lancés avec toutes les configurations de **ClariNet1** et **ClariNet2** en faisant varier les paramètres A, B, C et D. Les scores de justesse obtenus par l'ensemble des apprentissages ont été représentés dans l'ensemble des courbes des figures 5.5 et 5.6 pour les deux réseaux **ClariNet1** et **ClariNet2** respectivement.

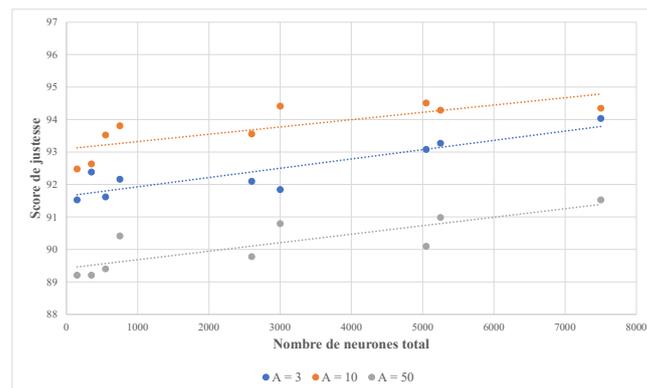
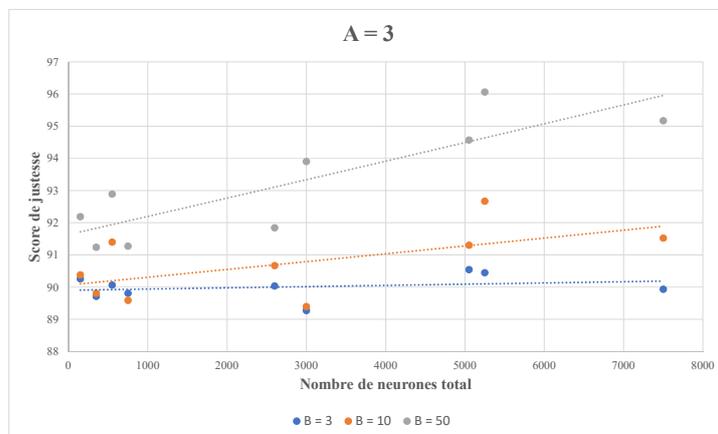


FIGURE 5.5 – Impact du nombre de neurones et du nombre des filtres sur le score de justesse en 4 classes météorologiques avec ClariNet1 sur la base Cerema-AWH.

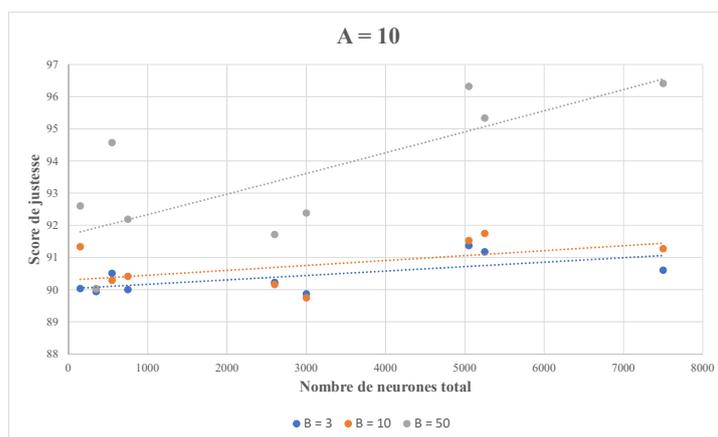
### a. Influence du nombre de neurones sur FC1 et FC2

Nous étudions l'impact du nombre de neurones sur les couches FC1 et FC2. Les courbes de la Figure 5.5 présentent, les scores de justesse obtenus par l'ensemble des tests en fonction du nombre total des neurones sur les deux couches FC1 et FC2. Chaque courbe correspond à un nombre de filtres fixe de la couche de convolution  $A = \{3, 10, 50\}$ .

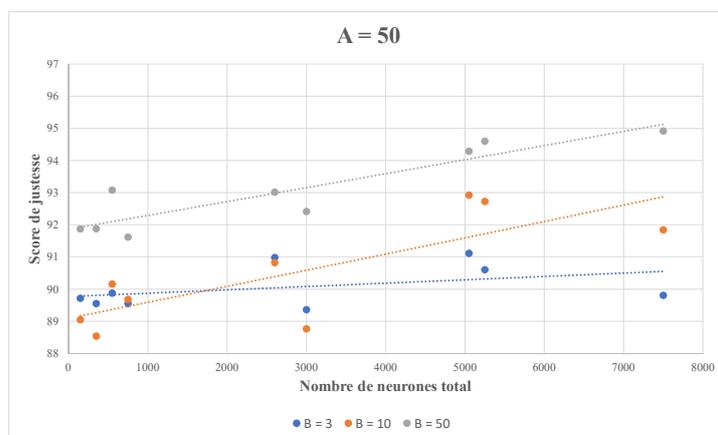
D'après la Figure 5.5, nous remarquons que toutes les courbes de tendance présentent une augmentation du score en fonction du nombre de neurones. Cette augmentation est beaucoup plus marquée pour les courbes obtenues par le réseau ClariNet2 (Figure 5.6). Cela montre que, pour un nombre fixe de filtres sur les couches de convolution, le fait d'augmenter le nombre de neurones sur les couches complètement connectées augmente le score de justesse (par exemple de 91% à 96% dans la Figure 5.6a). Ce résultat est confirmé par toutes les autres courbes qui correspondent à un nombre de filtres différent sur la première couche de convolution (courbes 5.6b et 5.6c). Donc, indépendamment de



(a) Architecture avec 3 filtres sur la première couche de convolution.



(b) Architecture avec 10 filtres sur la première couche de convolution.



(c) Architecture avec 50 filtres sur la première couche de convolution.

FIGURE 5.6 – Impact du nombre de neurones et du nombre des filtres sur le score de justesse en 4 classes météorologiques avec ClariNet2 sur la base Cerema-AWH.

l'architecture, un nombre important de neurones peut augmenter les performances de classification d'un réseau de neurones. Cette hypothèse a été déjà utilisée par le réseau de neurones VGG [210] qui utilise 4096 neurones sur l'une de ses couches FC.

### **b. Influence du nombre des couches de convolution**

Nous étudions l'impact de l'augmentation du nombre de couches de convolution. D'après la Figure 5.5, le meilleur score de justesse obtenu est 94,5% avec le réseau **ClariNet1**. Par contre, avec le réseau **ClariNet2**, nous arrivons à atteindre un score de justesse de 96,5% (Figure 5.6b). Ces deux scores sont bien meilleurs que le score obtenu dans [74] qui est égal à 87% en utilisant un simple réseau de neurones de type perceptron sans couches de convolution. Cela montre, d'une part, l'importance des couches de convolution dans la classification des conditions météorologiques dégradées, et d'autre part, l'importance de l'augmentation du nombre de ce type de couches pour l'optimisation des résultats de classification.

### **c. Influence du nombre des filtres de convolution**

Nous étudions l'impact du nombre de filtres sur les couches de convolution. D'après les courbes de la Figure 5.5, nous remarquons que pour un nombre élevé de filtres ( $A = 50$ ), les scores de classifications sont moins bons que ceux avec  $A = 3$  et  $A = 10$ . De même, nous remarquons que pour un nombre faible de filtres, les résultats de classification ne sont pas satisfaisants. Nous retrouvons ces remarques dans les courbes de la Figure 5.6. En effet, si nous comparons l'intervalle des scores de justesse des trois graphes 5.6a, 5.6b et 5.6c, nous remarquons que pour un nombre de filtres sur la première couche de convolution égal à  $A = 50$ , nous obtenons des scores maximaux de justesse inférieurs à ceux obtenus pour  $A = 3$  qui sont aussi inférieurs aux scores obtenus pour  $A = 10$ . On peut en conclure qu'il faut avoir des filtres de convolutions sans trop augmenter leur nombre sur la première couche du réseau.

Maintenant, nous étudions l'impact du nombre de filtres sur la deuxième couche de convolution du réseau de neurones **ClariNet2**. D'après les courbes de la Figure 5.6, nous remarquons que les courbes qui correspondent à un nombre de filtres sur la deuxième couche de convolution  $B = 50$  présentent les meilleurs scores de justesse par rapport à  $B = 10$  et  $B = 3$ . Cela montre qu'en augmentant le nombre de filtres sur la deuxième couche, les performances du réseau de neurones augmentent.

Cette hypothèse est confirmée par la plupart des réseaux de neurones de l'état de l'art. En effet, ces réseaux ont tendance à utiliser plusieurs couches de convolutions en augmentant le nombre de leurs filtres d'une couche à l'autre (exemple ResNet décrit dans le Tableau 3.1).

Cette analyse et mise en place des réseaux de neurones instrumentaux **ClariNet1** et **ClariNet2** nous a permis de constater l'importance d'utiliser des réseaux de neurones convolutifs vu l'augmentation du score de classification qu'ils introduisent. Cette analyse rapide nous conforte dans l'utilisation des réseaux de neurones profonds afin d'optimiser notre système de classification des conditions météorologiques dégradées.

### 5.2.4 Sélection des réseaux de neurones convolutifs

La sélection des différents DNNs est basée sur l'étude réalisée dans [23]. Dans la Figure 5.7, les auteurs de [23] comparent différents réseaux de neurones convolutifs profonds selon trois critères :

- le score de justesse (équation (5.1)) sur la base ImageNet [64] avec 1000 classes;
- le temps de calcul par apprentissage qui dépend du nombre total de paramètres à apprendre traduit en taille des fichiers à collecter en mémoire (exprimée en Mo). Cette information est très utile pour comprendre la quantité minimale de mémoire GPU requise pour chaque modèle;
- la consommation en mémoire. Il s'agit d'évaluer la consommation totale en mémoire, qui comprend toute la mémoire allouée, c'est-à-dire la mémoire allouée pour le modèle et la mémoire requise lors du traitement du lot d'images.

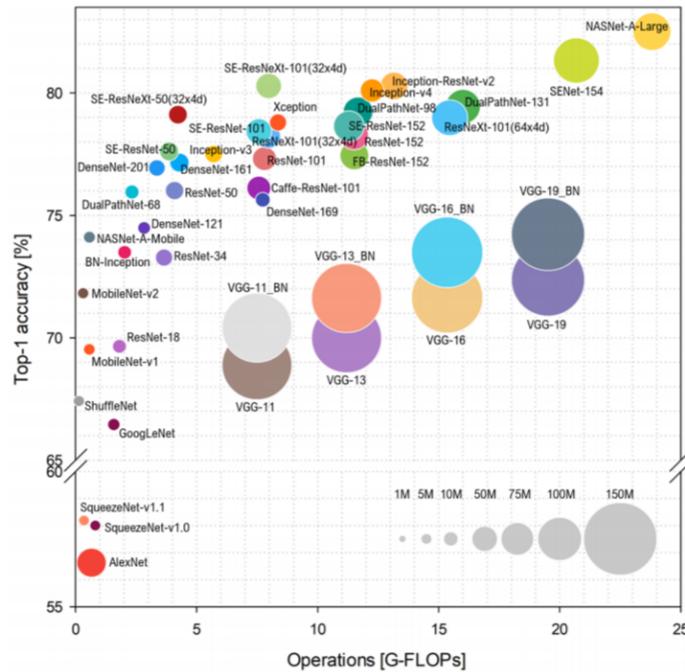


FIGURE 5.7 – Comparaison des architectures des DNNs en fonction du score de justesse et de la complexité du calcul [23].

D'après la Figure 5.7, nous avons sélectionné les modèles qui sont les moins complexes en calcul et qui possèdent un score de justesse élevé (Tableau 5.4). La valeur de la consommation en mémoire concerne un lot de 16 images. À partir de ce tableau, nous avons éliminé les réseaux qui consomment plus de mémoire (1,4 Go). Parmi les réseaux qui restent, nous avons éliminé celui qui atteint le score de justesse le plus faible. Donc finalement, trois DNN suivants ont été sélectionnés : Inceptions-v4 [218], ResNet-152 [105] et DenseNet-121 [115].

Afin d'évaluer ces différents réseaux profonds, une classification en cinq classes météorologiques  $C1 = \{JCN, JB1, JP1, JB2, JP2\}$  a été mise en place pour classifier les images de la base Cerema-AWH.

| Architecture   | Score de justesse | Consommation (GB) |
|----------------|-------------------|-------------------|
| NasNet-A-Large | 82,5              | 1,4               |
| SENet-154      | 81,32             | 1,4               |
| Inception-v4   | 80,1              | 1                 |
| Xception       | 78,79             | 1,4               |
| ResNet-152     | 78,25             | 1                 |
| DenseNet-121   | 74,47             | 0,7               |
| MobileNet-v2   | 71,81             | 0,7               |

TABLEAU 5.4 – Étude comparative des architectures de l'état de l'art.

Le Tableau 5.5 présente les différents résultats de classification obtenus en utilisant les réseaux de neurones profonds sélectionnés.

|                   | Architectures |              |              |       |           |
|-------------------|---------------|--------------|--------------|-------|-----------|
|                   | ResNet-152    | DenseNet-121 | Inception-v4 | LeNet | ClariNet2 |
| <b>Cerema-AWH</b> | 82,6          | 80,5         | 81,3         | 75,2  | 77,3      |

TABLEAU 5.5 – Résultats de classification des conditions météorologiques en 5 classes avec différentes architectures sur la base Cerema-AWH.

D'après les résultats du Tableau 5.5, nous remarquons que l'architecture des DNNs montre ses performances sur la classification des conditions météorologiques. Les scores de justesse de ResNet-152, DenseNet-121 et Inceptions-v4 dépassent les scores de LeNet et du réseau instrumental ClariNet2. La différence entre l'architecture de ces réseaux et celles de LeNet et ClariNet2 se présente dans le nombre des couches de convolutions et dans la complexité du modèle. Comme il a été décrit dans la section 3.5 plusieurs techniques ont été appliquées sur ces DNNs afin d'optimiser leurs performances. Ces résultats font appel à la conclusion tirée dans la section 5.2.3 et qui exige l'utilisation des réseaux de neurones profonds. Nous utilisons ces trois réseaux de neurones dans la suite de notre étude.

### 5.3 Influence des données d'entrée et de sortie

Dans cette section, nous allons procéder à une série d'analyses en utilisant les trois réseaux de neurones profonds sélectionnés précédemment : ResNet-152, DenseNet-121 et Inception-v4 (voir la définition précise en section 3.5) sur des images de taille  $224 \times 224$  pour ResNet-152, DenseNet-121 et  $299 \times 299$  pour Inception-v4. Ces analyses permettent d'étudier l'influence du changement de scènes (fixe ou aléatoire), l'influence du nombre de données d'entrée du réseau de neurones (nombre d'images de la base de données), l'influence de l'augmentation du nombre de classes météorologiques sur la classification et enfin l'influence de l'utilisation des bases de données publiques.

### 5.3.1 Influence du changement de scènes

Avant de commencer l'ensemble des apprentissages pour la classification météorologique, il vaut mieux avoir une idée de l'impact du changement de scène dans la donnée d'entrée sur le résultat de classification météorologique. Pour cela, nous allons fixer la base de test pour ne faire varier que la base d'apprentissage.

D'après la présentation de la base Cerema-AWH (section 4.4), cette dernière est acquise par une caméra routière qui est fixée sur le bord de l'autoroute A75. La scène étant fixe, les images diffèrent les unes par rapport aux autres avec les mouvements des voitures, la pousse des herbes et les changements météorologiques.

Dans cette partie, nous étudions deux variantes de scènes : une scène fixe et une scène aléatoire. La scène fixe consiste à extraire une zone de  $224 \times 224$  pixels ( $299 \times 299$  pixels pour l'apprentissage de Inception-v4) au centre de l'image de la base Cerema-AWH de taille initiale  $640 \times 480$ , qui sera l'entrée du réseau de neurones (Figure 5.8). La phase d'apprentissage et la phase de test se font sur cette même zone.



FIGURE 5.8 – Exemple d'une zone fixe extraite du centre d'une image de la base Cerema-AWH.

La scène aléatoire consiste à extraire une zone aléatoire de  $224 \times 224$  pixels ( $299 \times 299$  pixels pour l'apprentissage de Inception-v4) à partir de l'image originale de la base Cerema-AWH. Dans ce cas, les images d'entrée du réseau ne seront plus les mêmes en termes de scène. La phase de test se fait sur des images fixes extraites au centre de l'image, c'est la même base de test que celle du cas de la scène fixe. Cela permet de ne se concentrer que sur l'impact de l'apprentissage sur scènes fixes/aléatoires.

La classification porte sur cinq catégories météorologiques  $C1 = \{JCN, JB1, JP1, JB2, JP2\}$ . À noter que les conditions normales de nuit (NCN) ont été enlevées dans tous les tests à venir car il s'est avéré que cette classe est toujours bien classifiée et donc nous ne nous sommes intéressés qu'aux conditions météorologiques dégradées de jour. Les résultats de classification sont présentés dans le Tableau 5.6.

D'après le Tableau 5.6, il est clair que le passage d'une scène fixe à une scène aléatoire a pour effet de dégrader les scores de classification pour les trois réseaux de neurones. Nous avons également fait cette analyse sur le réseau LeNet qui a donné un score de justesse de 87,2% en scène fixe et 70,4% en scène aléatoire, ce qui confirme aussi cette dégradation qui est indépendante de l'architecture du réseau et qui ne dépend que de la donnée d'entrée. En effet, pour le cas de la scène fixe, l'algorithme ne prend en considération que le changement des conditions météorologiques, ce qui lui permet de bien apprendre les

|                        | Architectures |              |              |
|------------------------|---------------|--------------|--------------|
|                        | ResNet-152    | DenseNet-121 | Inception-v4 |
| <b>Scène fixe</b>      | 89,8          | 89,3         | 86,7         |
| <b>Scène aléatoire</b> | 82            | 79,6         | 70           |

TABLEAU 5.6 – Résultats du changement de scènes sur la base Cerema-AWH avec 5 classes météorologiques.

caractéristiques spécifiques à chaque classe, ce qui n'est pas le cas pour la scène aléatoire où l'algorithme doit prendre en considération tous les changements de scènes y compris les changements météorologiques.

Bien que le changement de scènes limite les performances des réseaux de classification, il est intéressant de continuer à l'utiliser car c'est le cas d'application qu'on vise à la fin. En effet, l'avantage d'avoir un réseau de neurones appris sur des scènes aléatoires est qu'il sera capable de s'adapter à des bases de données de test différentes de celles de l'apprentissage en termes de scène vue. Cela sera utile lors de l'application de la méthode de transfert de bases (apprendre sur une base et tester sur une autre). Dans ce cas, une amélioration des résultats de classification avec les scènes aléatoires est exigée.

Dans la suite, les différentes analyses sont effectuées sur des bases d'apprentissage et de test ayant des scènes aléatoires.

### 5.3.2 Influence du nombre de données d'entrée

Cette analyse nous permet d'avoir une idée sur le nombre des données d'entrée qu'il faut utiliser pour arriver à mieux classifier les conditions atmosphériques. Pour cela, nous avons fait varier le nombre d'images dans la base d'apprentissage, en gardant la base de test fixe, pour étudier son influence. Notons Cerema-AWH-P la base contenant le moins d'images avec 15 000 images pour l'apprentissage et 5 000 images pour le test, et Cerema-AWH-G1 la base contenant plus d'images avec 125 000 images pour l'apprentissage et 5 000 images pour le test. Le Tableau 5.7 montre les résultats de classification pour ces deux bases de données.

|                      | Architectures |              |              |
|----------------------|---------------|--------------|--------------|
|                      | ResNet-152    | DenseNet-121 | Inception-v4 |
| <b>Cerema-AWH-P</b>  | 79,3          | 78,9         | 72,6         |
| <b>Cerema-AWH-G1</b> | 83,4          | 80,4         | 81,4         |

TABLEAU 5.7 – Résultats de classification des conditions météorologiques en 5 classes en changeant le nombre des données d'apprentissage

D'après le Tableau 5.7, le fait d'augmenter le nombre d'images de la base d'apprentissage permet d'améliorer les performances du système de classification. Cependant, cette amélioration dépend notamment de l'architecture du réseau de neurones.

Maintenant, nous faisons varier le nombre d'images de la base de test afin d'améliorer la précision. Pour cela, nous comparons les résultats de classification de la base Cerema-AWH-G1 avec ceux de la base Cerema-AWH-G2 ayant 125 000 images dans la base d'apprentissage et 75 000 images dans la base de test (Tableau 5.8).

|                      | Architectures |              |              |
|----------------------|---------------|--------------|--------------|
|                      | ResNet-152    | DenseNet-121 | Inception-v4 |
| <b>Cerema-AWH-G1</b> | 83,4          | 80,4         | 81,4         |
| <b>Cerema-AWH-G2</b> | 82,6          | 80,5         | 81,3         |

TABLEAU 5.8 – Résultats de classification des conditions météorologiques en 5 classes en changeant le nombre des données de test.

D'après le Tableau 5.8, nous remarquons que les scores n'ont pas eu une grande variation (hormis pour ResNet-152), mais ils sont beaucoup plus précis en augmentant le nombres d'images de test.

Dans les travaux de l'état de l'art, l'utilisation de bases de données à grande échelle est recommandée. Par contre, ces travaux n'indiquent pas un nombre de données fixe à faire rentrer dans le réseau, cela dépend de la nature de l'application, le réseau de neurones et la nature des données d'entrée. Dans la suite, nous utilisons la base Cerema-AWH-G2 en la notant Cerema-AWH, bien que Cerema-AWH-G1 aurait pu être suffisante, mise à part le réseau ResNet-152 qui montre un écart significatif.

### 5.3.3 Influence du nombre de classes

Dans cette partie nous nous intéressons à l'influence du changement du nombre de classes météorologiques sur le résultat de classification.

Commençons par une classification en 3 classes. Dans ce cas, la base d'apprentissage utilisée contient 75 000 images et la base de test contient 45 000 images sachant que le nombre d'images par classe est le même. Nous rappelons que les caractéristiques des classes météorologiques  $C_l = \{JCN, JB, JP\}$  sont :

- pour JCN : la distance de visibilité météorologique est fixée à 2 000m
- pour JB : la distance de visibilité météorologique est inférieure à 400m. C'est le brouillard routier.
- pour JP : l'intensité des précipitations est supérieure à  $2mm.h^{-1}$ .

Les scores de classification sont présentés dans le Tableau 5.9.

Comme notre objectif est de mesurer le temps par caméra, une classification plus fine a été proposée et donc une augmentation du nombre de classes a été mise en place. Ainsi, le deuxième cas consiste à catégoriser le temps selon cinq classes météorologiques. Dans ce cas, la base d'apprentissage utilisée contient 125 000 images et la base de test contient 75 000. Les caractéristiques des classes météorologiques  $C_l = \{JCN, JB1, JB2, JP1, JP2\}$  sont :

- pour JCN : la distance de visibilité météorologique est fixée à 2 000m
- pour JB1 : la distance de visibilité météorologique est inférieure à 400m.
- pour JB2 : la distance de visibilité météorologique est comprise entre 400m et 800m.

- pour JP1 : l'intensité des précipitations est supérieure à  $2\text{mm.h}^{-1}$ .
- pour JP2 : l'intensité des précipitations est inférieure à  $2\text{mm.h}^{-1}$ .

Les scores de classification en cinq classes sont présentés aussi dans le Tableau 5.9.

| Base de données | Nb de classes | Architectures |              |              |
|-----------------|---------------|---------------|--------------|--------------|
|                 |               | ResNet-152    | DenseNet-121 | Inception-v4 |
| Cerema-AWH      | 3             | 98,8          | 98,7         | 98           |
|                 | 5             | 82,6          | 80,5         | 81,3         |

TABLEAU 5.9 – Résultats de classification des conditions météorologiques en 3 et 5 classes avec différentes architectures sur la base Cerema-AWH.

D'après le Tableau 5.9, l'augmentation du nombre de classes météorologiques a un impact sur les résultats de classification. En effet, la tâche de classification devient de plus en plus compliquée, car le système doit faire la différence entre deux classes de brouillard et deux classes de pluie.

### 5.3.4 Utilisation des bases de données publiques

Les analyses précédentes ont été effectuées sur notre base de données construite au Cerema : Cerema-AWH (section 4.4). Il était intéressant de tester nos méthodes de classification sur d'autres bases météorologiques publiques, d'étudier leur apport et leur impact, et notamment comparer les résultats obtenus sur notre base avec celles obtenues sur les autres bases. L'utilisation de ces bases nous permettra de faire des apprentissages sur des scènes globales et introduire une variété de scènes vu que les images de la base Cerema-AWH appartiennent à des scènes acquises par une caméra fixe.

Les deux bases de données prises en considération sont : Image2Weather [43] (qui sera noté dans la suite I2W) et MWD (Multi-class Weather Dataset) [153]. Pour rappel, ces bases sont présentées en détail dans la section 4.3.

Pour comparer les bases Cerema-AWH, I2W et MWD, les contraintes suivantes ont été prises en considération :

- comme les images de I2W et MWD sont en couleur et celles de Cerema-AWH sont en niveaux de gris, les images des bases publiques ont été converties en niveaux de gris;
- la classe "neige" a été enlevée des deux bases I2W et MWD puisque Cerema-AWH ne contient pas cette classe dans le cadre de cette thèse;
- la classe "orage" a été enlevée de la base MWD car I2W et Cerema-AWH ne la contiennent pas;
- les deux classes "ensoleillé" et "nuageux" de I2W et MWD ont été fusionnées dans une seule classe qui présentera les conditions normales.

En prenant en considération toutes ces contraintes, nous ne pourrons plus faire la classification sur cinq classes météorologiques en utilisant les bases I2W et MWD, et donc la classification sera faite selon trois classes météorologiques  $Cl = \{JCN, JB, JP\}$ . Le Tableau 5.10 présente les résultats obtenus avec les réseaux de neurones pour les 3 bases de données.

|                   | Architectures |              |              |
|-------------------|---------------|--------------|--------------|
|                   | ResNet-152    | DenseNet-121 | Inception-v4 |
| <b>MWD</b>        | 75,9          | 76,7         | 74,5         |
| <b>I2W</b>        | 74,7          | 75,6         | 75           |
| <b>Cerema-AWH</b> | 98,8          | 98,7         | 98           |

TABLEAU 5.10 – Comparaison de la classification sur 3 bases différentes avec 3 classes météorologiques.

D’après le Tableau 5.10, nous remarquons que pour les bases publiques MWD et I2W, les scores de classification sont assez proches pour les 3 architectures. Par contre, les résultats sur la base Cerema-AWH sont bien meilleurs. Cela peut être expliqué par le fait que :

- les images des bases publiques présentent une diversité au niveau des scènes (images acquises depuis des webcams installées partout dans le monde) ;
- les images de Cerema-AWH appartiennent à des scènes acquises par une caméra routière fixe.

Cela confirme ce que nous avons vu précédemment : le changement de scènes diminue les performances de classification météorologique des réseaux de neurones.

## 5.4 Transfert de bases de données

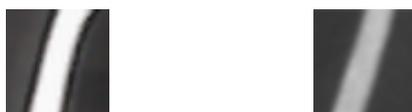
Dans les parties précédentes, des classifications météorologiques sur la base Cerema-AWH ont été effectuées avec succès en obtenant de bons résultats. Pour cette raison, il serait intéressant de tester le même algorithme sur une deuxième base de données, autre que celle de l’apprentissage, ayant d’autres caractéristiques. En effet, quel serait le résultat si une nouvelle base était utilisée pour la phase de test ?

Ainsi, nous nous intéressons dans cette section au transfert de bases. Il s’agit de faire l’apprentissage sur une base de données d’images et ensuite de faire le test sur une autre base. Cette méthode permet d’étudier la portabilité du système de détection des conditions météorologiques dégradées d’un site à un autre.

En premier lieu, nous effectuons une étude académique pour une première observation de l’influence du transfert de bases sur la classification météorologique. Pour cela, nous utilisons les mêmes éléments déjà utilisés dans la section 5.3 : des patchs contrastés de taille  $28 \times 28$  pixels avec LeNet comme réseau d’apprentissage. Nous étudions le transfert de bases entre la base Cerema-AWP comme base d’apprentissage et Cerema-AWH comme base de test.

Pour une première étude, nous nous limitons au transfert des conditions météorologiques. Pour cela, même si nous changeons de sites, nous choisissons des données d’entrée des deux bases de telle sorte qu’elles soient les plus similaires possible, cela permet de ne prendre en considération que le changement des conditions météorologiques dégradées. La Figure 5.9 montre les images d’entrée pour chaque base de données.

Après avoir entraîné le réseau LeNet sur la base Cerema-AWP et l’avoir testé sur la base Cerema-AWH pour une classification sur quatre classes  $Cl = \{JCN, JB, JP, NCN\}$ , un score



(a) Patch de la base AWP. (b) Patch de la base AWH.

FIGURE 5.9 – Exemple de deux patches similaires des bases Cerema-AWP et Cerema-AWH.

de classification de 65,7% a été obtenu. Il s’agit d’un score qui est inférieur à celui qui a été obtenu en faisant l’apprentissage et le test sur la même base de données Cerema-AWH qui était de 99,7%. Cela montre que, même si les données d’entrée se ressemblent, les performances du système de classification se dégradent. Cette pré-étude permet de préjuger que, si nous ajoutons la variabilité de scènes, les résultats de classification ne seront pas satisfaisants.

Afin d’analyser plus en détail ce résultat, une matrice de confusion des quatre classes a été construite pour déterminer les conditions météorologiques mal prédites en passant de la base Cerema-AWP à la base Cerema-AWH (Tableau 5.11).

|            |     | Vérité terrain |    |    |     |
|------------|-----|----------------|----|----|-----|
|            |     | JCN            | JB | JP | NCN |
| Prédiction | JCN | 38             | 0  | 0  | 2   |
|            | JB  | 0              | 68 | 40 | 0   |
|            | JP  | 61             | 32 | 60 | 0   |
|            | NCN | 0              | 0  | 0  | 98  |

TABLEAU 5.11 – Résultats détaillés du transfert de bases entre Cerema-AWP et Cerema-AWH.

Comme le montre le Tableau 5.11, il existe 32% d’images de JB et 61% d’images de JCN qui ont été prédites en tant que JP. De plus, il y a 40% d’images de JP qui ont été reconnues comme JB. Il est donc clair que la classification de la pluie de jour (JP) pose un problème pour le réseau LeNet. Pour cette raison, une vérification des images de pluie mal classées a été faite. Il s’est avéré que, pour la base de données Cerema-AWH, les gouttes de pluie se collent sur l’objectif de la caméra routière, contrairement aux images de la pluie acquises sur la plateforme PAVIN Brouillard & Pluie du Cerema (Figure 5.10).

À la suite de ce problème, une omission de la classe météorologique pluvieuse a été proposée afin de mieux comparer le reste des conditions. Un apprentissage prend lieu de nouveau sur la nouvelle base de données sans pluie. Dans ce cas, et avec 3 classes météorologiques  $C_l = \{JCN, JB, NCN\}$ , un score de classification de 96,8% a été obtenu.

Ce transfert de bases a été effectué entre la base Cerema-AWP qui a été acquise en site interne (la plateforme PAVIN Brouillard & Pluie du Cerema) et la base Cerema-AWH qui a été acquise en site externe (site autoroutier du Col de la Fageole). Cependant, nous visons une portabilité de notre système de classification météorologique sur des sites externes et qui présentent une diversité d’images et de différentes caractéristiques liées aux scènes. Pour cela, nous avons appliqué le transfert de bases sur les bases publiques I2W et MWD qui ont été utilisées précédemment. Les parties suivantes présentent ces expériences et



FIGURE 5.10 – Exemple de deux images de pluie des bases Cerema-AWH et Cerema-AWP.

leurs résultats de classification.

#### 5.4.1 Transfert entre Cerema-AWH et les bases publiques

Cette partie présente un transfert de bases entre les trois bases Cerema-AWH, I2W et MWD (ces bases sont décrites dans le chapitre 4) selon deux cas.

Le premier cas concerne le transfert de la base Cerema-AWH vers chacune des bases publiques. Le Tableau 5.12 présente les résultats de classification obtenus. Nous sommes assez proches du tirage aléatoire uniforme sur les trois classes. Le tableau montre que pour un algorithme entraîné sur la base Cerema-AWH et testé sur une autre base, le transfert n'est pas faisable vu les faibles scores obtenus. Ce résultat était plus ou moins prévisible, car le réseau apprend sur une base dont la scène est fixe, et donc les descripteurs d'images extraits ne concerneront que les changements des conditions météorologiques. Par contre, quand il va être testé sur une autre base, où les scènes et les phénomènes climatiques sont différents, il trouvera des difficultés dans la classification du temps.

| Base d'apprentissage | Base de test | Architectures |              |              |
|----------------------|--------------|---------------|--------------|--------------|
|                      |              | ResNet-152    | DenseNet-121 | Inception-v4 |
| Cerema-AWH           | MWD          | 43            | 37,6         | 38           |
|                      | I2W          | 37,6          | 37,3         | 37           |

TABLEAU 5.12 – Résultats de classification du transfert de bases entre Cerema-AWH et les bases publiques avec 3 classes météorologiques.

Le deuxième cas concerne le transfert de l'une des bases publiques vers la base Cerema-AWH. Contrairement au premier cas, le transfert de bases de MWD vers Cerema-AWH présente de meilleurs scores de classification (Tableau 5.13). Cette amélioration s'explique par le fait que le réseau a été appris sur différentes scènes et a été testé sur une scène fixe.

#### 5.4.2 Transfert entre des bases mélangées

Dans la section précédente, l'utilisation de la méthode de transfert de bases entre la base Cerema-AWH et les bases publiques n'a pas abouti à des résultats de classification satisfaisants. Pour cette raison, un mélange des différentes bases publiques avec la base

| Base d'apprentissage | Base de test | Architectures |              |              |
|----------------------|--------------|---------------|--------------|--------------|
|                      |              | ResNet-152    | DenseNet-121 | Inception-v4 |
| MWD                  | Cerema-AWH   | 63            | 59           | 54           |
| I2W                  |              | 33            | 36           | 42           |

TABLEAU 5.13 – Résultats de classification du transfert de bases entre les bases publiques et Cerema-AWH avec 3 classes météorologiques.

Cerema-AWH est proposé. Le fait de mélanger les bases permet d'avoir une nouvelle base plus riche et variée en images contenant différentes scènes et différentes intensités des conditions météorologiques. En effet, et comme il était indiqué dans la section 4.3, en parcourant les images de pluie de la base I2W, la plupart des images labellisées comme *pluie* contenaient de la pluie très faible reconnue par la chaussée mouillée, voire une pluie terminée. Aussi, dans la base MWD, les images qui expriment un temps pluvieux sont généralement des images d'inondations où la pluie est très forte. Dans ce cas, en mélangeant ces deux bases, nous obtenons une nouvelle base qui contient les deux types de pluie. De plus, l'ajout des images appartenant à I2W et MWD à la base Cerema-AWH apporte le changement de scènes, car les images de Cerema-AWH sont acquises par une caméra fixe sur un seul site, tandis que les images de I2W et MWD sont collectées à partir de différents sites/endroits.

Nous utilisons les mêmes images qui ont été sélectionnées dans la section précédente qui porte sur la comparaison des bases publiques avec Cerema-AWH afin de respecter les contraintes proposées. Dans ce cas, les bases mélangées ne contiendront que trois classes  $Cl = \{JCN, JB, JP\}$ . L'apprentissage se fait toujours avec les trois architectures ResNet-152, DenseNet-121 et Inceptions-v4. À noter que, dans une base donnée, le nombre des images par classe est toujours le même. Le Tableau 5.14 indique le nombre d'images dans chaque base mélangée. Ce choix de nombre d'images par base est imposé par le nombre de données appartenant à chaque base (par exemple, la base I2W ne contient que 720 images en base d'apprentissage et 351 images en base de test) afin de conserver une répartition uniforme entre les bases d'une même base mélangée.

|                    | Base d'apprentissage |     |      | Base de test |     |      |
|--------------------|----------------------|-----|------|--------------|-----|------|
|                    | AWH                  | I2W | MWD  | AWH          | I2W | MWD  |
| <b>AWH+I2W</b>     | 720                  | 720 | *    | 351          | 351 | *    |
| <b>AWH+MWD</b>     | 3000                 | *   | 3000 | 2100         | *   | 2100 |
| <b>MWD+I2W</b>     | *                    | 720 | 720  | *            | 351 | 351  |
| <b>AWH+I2W+MWD</b> | 720                  | 720 | 720  | 351          | 351 | 351  |

TABLEAU 5.14 – Nombre d'images pour chaque variante de base mélangée. Le nombre d'images indiqué concerne les trois classes météorologiques. Ce nombre est partagé équitablement sur ces classes.

### Influence du mélange des bases de données

Tout d'abord, et avant d'appliquer la méthode de transfert de bases, nous étudions l'influence de cette fusion de bases sur la classification avec un apprentissage et un test sur la même base mélangée. Le Tableau 5.15 présente les différentes variantes des bases mélangées et les scores de classification obtenus pour chacune des bases.

|                        | Architectures |              |              |
|------------------------|---------------|--------------|--------------|
|                        | ResNet-152    | DenseNet-121 | Inception-v4 |
| <b>AWH + MWD</b>       | 82            | 84           | 82           |
| <b>AWH + I2W</b>       | 82,5          | 83,8         | 78,9         |
| <b>MWD + I2W</b>       | 69,3          | 72,3         | 66,4         |
| <b>AWH + MWD + I2W</b> | 75            | 78,1         | 73,7         |

TABLEAU 5.15 – Comparaison de la classification sur différentes variantes de bases mélangées avec 3 classes météorologiques.

D'après le Tableau 5.15, et en comparant ces scores avec ceux obtenus sur la base Cerema-AWH et les bases publiques sans fusion (Tableau 5.10), nous remarquons qu'en ajoutant la base Cerema-AWH aux autres bases I2W et MWD, une amélioration de la classification météorologique apparaît, hormis pour l'ajout d'une base publique à Cerema-AWH.

L'avantage lié à la base Cerema-AWH est sa fiabilité au niveau des données météorologiques. L'augmentation du pourcentage des images de la base Cerema-AWH dans une base mélangée peut-il améliorer le résultat de classification? Pour répondre à cette question nous avons augmenté l'écart entre la base d'apprentissage de Cerema-AWH et celle de l'une des bases publiques selon le Tableau 5.16 et nous avons enregistré les scores de justesse présentés dans le Tableau 5.17.

|                        | Base d'apprentissage |     |      | Base de test |     |      |
|------------------------|----------------------|-----|------|--------------|-----|------|
|                        | AWH                  | I2W | MWD  | AWH          | I2W | MWD  |
| AWH + I2W              | 720                  | 720 | *    | 351          | 351 | *    |
| AWH + I2W <sub>2</sub> | 1440                 | 720 | *    | 351          | 351 | *    |
| AWH + I2W <sub>3</sub> | 2880                 | 720 | *    | 351          | 351 | *    |
| AWH + MWD              | 3000                 | *   | 3000 | 2100         | *   | 2100 |
| AWH + MWD <sub>2</sub> | 6000                 | *   | 3000 | 2100         | *   | 2100 |
| AWH + MWD <sub>3</sub> | 12000                | *   | 3000 | 2100         | *   | 2100 |

TABLEAU 5.16 – Nombre d'images pour chaque variante des bases AWH+I2W et AWH+MWD. Le nombre d'images indiqué concerne les trois classes météorologiques. Ce nombre est partagé équitablement sur ces classes.

D'après le Tableau 5.17, nous remarquons que la proportion des images appartenant à chaque base influe sur le résultat de la classification. Nous remarquons aussi que lorsque le nombre d'images appartenant à la base Cerema-AWH augmente, le score de justesse

|                        | Architectures |              |              |
|------------------------|---------------|--------------|--------------|
|                        | ResNet-152    | DenseNet-121 | Inception-v4 |
| AWH + I2W              | 82,5          | 83,8         | 78,9         |
| AWH + I2W <sub>2</sub> | 83,7          | 86,2         | 79           |
| AWH + I2W <sub>3</sub> | 84            | 85,5         | 79,4         |
| AWH + MWD              | 82            | 84           | 82           |
| AWH + MWD <sub>2</sub> | 82,7          | 84           | 83           |
| AWH + MWD <sub>3</sub> | 83,5          | 85           | 83,2         |

TABLEAU 5.17 – Résultats de classification obtenus avec les différentes variantes des bases AWH+I2W et AWH+MWD.

augmente. Cela dit que la base Cerema-AWH introduit une amélioration de la performance du système de classification météorologique.

Revenons aux résultats du Tableau 5.15. La combinaison qui présente les plus faibles scores est MWD+I2W. En effet, en regardant la classification des images de la base de test (Tableau 5.18), il s'avère que l'algorithme confond entre JCN et JP. Le problème ici est causé par les images de la pluie qui appartiennent à I2W. Vu la pluie très faible que ces images contiennent, le réseau se permet de les classer comme des conditions normales. Cette mauvaise prédiction est un peu améliorée en ajoutant des images de la base Cerema-AWH qui contient des images de pluie faible.

|            |     | Vérité terrain |    |    |
|------------|-----|----------------|----|----|
|            |     | JCN            | JB | JP |
| Prédiction | JCN | 58             | 16 | 26 |
|            | JB  | 3              | 85 | 12 |
|            | JP  | 16             | 10 | 74 |

TABLEAU 5.18 – Résultats détaillés sur la prédiction des classes météorologiques avec la base mélangée MWD+I2W. Le tableau exprime la précision appartenant à chaque classe. Il présente le résultat du test avec le réseau DenseNet-121 pour un score de justesse de 72,3%.

D'après tous ces résultats, nous remarquons un point important. L'ajout de notre base Cerema-AWH aux bases publiques améliore nettement les résultats puisque nous gagnons environ 10 points sur les scores de chacune des bases publiques. Cela est dû très certainement, d'une part, à la richesse de cette base qui contient deux types de brouillard et deux types de pluie, et d'autre part, à la fiabilité de la labellisation qui se base sur des capteurs météorologiques. Et donc, si nous mélangeons une base qui contient beaucoup de scènes différentes et qui n'est pas calibrée météorologiquement avec une base qui ne contient pas une diversité de scènes mais qui est riche et fiable en données météorologiques, cela améliore la classification par rapport aux bases de données publiques.

### Transfert entre les bases mélangées

Cette partie porte sur l'utilisation des 3 bases mélangées tout en faisant le transfert de bases. Plusieurs variantes ont été testées où la phase de l'apprentissage se fait sur un couple de deux bases et la phase de test se fait sur la troisième base qui est complètement inconnue par rapport à celles de l'apprentissage. Le Tableau 5.19 présente les différents résultats de transfert de bases.

| Nb variante | Base d'apprentissage | Base de test | Architectures |              |              |
|-------------|----------------------|--------------|---------------|--------------|--------------|
|             |                      |              | ResNet-152    | DenseNet-121 | Inception-v4 |
| 1           | AWH+MWD              | I2W          | 60            | 56           | 50           |
| <i>R1</i>   | <i>AWH</i>           | <i>I2W</i>   | <i>37,6</i>   | <i>37,3</i>  | <i>37</i>    |
| <i>R2</i>   | <i>MWD</i>           | <i>I2W</i>   | <i>37,2</i>   | <i>39,9</i>  | <i>39,9</i>  |
| 2           | AWH+I2W              | MWD          | 47            | 49,3         | 47           |
| <i>R3</i>   | <i>AWH</i>           | <i>MWD</i>   | <i>43</i>     | <i>37,6</i>  | <i>38</i>    |
| <i>R4</i>   | <i>I2W</i>           | <i>MWD</i>   | <i>45,4</i>   | <i>44,6</i>  | <i>48</i>    |
| 3           | MWD+I2W              | AWH          | 45            | 44,6         | 50           |
| <i>R5</i>   | <i>MWD</i>           | <i>AWH</i>   | <i>63</i>     | <i>59</i>    | <i>54</i>    |
| <i>R6</i>   | <i>I2W</i>           | <i>AWH</i>   | <i>33</i>     | <i>36</i>    | <i>42</i>    |

TABLEAU 5.19 – Résultats de classification du transfert entre les bases mélangées avec 3 classes météorologiques. Les cas en italique (Rx) sont des cas de référence permettant de faciliter la comparaison entre les résultats du transfert obtenus avec les bases mélangées et les résultats de transfert obtenus dans les analyses précédentes.

Pour une première observation du Tableau 5.19, nous remarquons que les scores de classification varient entre 44% et 60%. Mais en comparant ces résultats avec les résultats obtenus précédemment (les cas R1 - R6), nous constatons que le fait de mélanger deux bases acquises sur des sites différents et des conditions météorologiques différentes peut améliorer les résultats de transfert de bases (hormis le cas de la variante 3 en comparaison avec la référence R5).

Nous remarquons aussi que l'ajout de la base Cerema-AWH permet d'améliorer la classification de temps par rapport aux autres bases publiques. Cela est dû à la scène fixe qu'elle présente dans toutes ses images ainsi qu'aux données météorologiques fiables qu'elle présente.

Les résultats de la classification météorologique obtenus avec la méthode de transfert de bases dans les différentes parties de cette section n'ont pas dépassé 64%. Cela veut dire que cette tâche de transfert est un véritable challenge qui devrait être étudié plus profondément dans de futurs travaux afin d'améliorer ces résultats.

## 5.5 Mesure de la distance de visibilité météorologique

L'objectif dans cette section est d'obtenir une mesure continue des conditions météorologiques et plus précisément de la distance de visibilité météorologique qui caractérise la condition du brouillard. En effet, notre méthode consiste à classer les conditions

météorologiques dégradées. Une fois la classe de brouillard prédite, notre système mesure sa distance de visibilité météorologique. Pour cela, un réseau de neurones qui génère des valeurs continues basées sur des images d'entrée a été mis en place. Ce réseau permet d'estimer la mesure de la condition météorologique rencontrée.

Le réseau de neurones convolutif utilisé est nommé "ClariNet-M" (Figure 5.11) ayant une architecture simple.

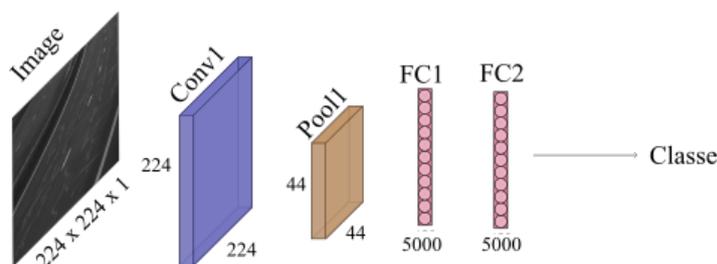


FIGURE 5.11 – Architecture du réseau de neurones de mesure ClariNet-M.

La particularité d'un réseau de neurones de mesure est sa sortie. En effet, au lieu d'utiliser la couche de type Softmax (utilisée à la sortie d'un réseau de classification), nous utilisons une distance euclidienne entre le modèle et le label selon l'équation (5.2). Dans ce cas, le réseau de neurones sera optimisé à l'aide d'une fonction de perte qui exprime l'erreur quadratique moyenne entre les valeurs de sortie du réseau de neurones  $\hat{y}_n$  et le label fourni  $y_n$  en tant que valeurs continues.

$$E = \frac{1}{2N} \sum_{n=1}^N \|\hat{y}_n - y_n\|_2^2, \quad (5.2)$$

où N est le nombre total des images.

La phase de mesure peut passer par deux étapes. La première étape consiste à cerner la gamme du brouillard (fort ou faible). S'il s'agit d'un brouillard fort, le système passe à la deuxième étape qui permet de mesurer d'une manière plus précise la distance de visibilité météorologique. Dans notre cas, nous nous intéressons au brouillard routier, c'est le brouillard fort ayant une visibilité inférieure à 400 m.

L'évaluation du résultat de mesure se fait à l'aide d'un coefficient de corrélation  $R^2$  entre les valeurs réelles et les valeurs prédites. Ce coefficient est comparé avec le coefficient de corrélation  $R^2_{\text{capteurs}}$  entre les deux capteurs utilisés pour mesurer la visibilité météorologique sur site. Nous rappelons que la base de données utilisée dans cette partie est la base Cerema-AWH-M (voir description dans la section 4.4). Le Tableau 5.20 montre les résultats de la prédiction de la distance de visibilité météorologique.

| Classes météo | $R^2$ | Eq de régression         | $R^2_{\text{capteurs}}$ |
|---------------|-------|--------------------------|-------------------------|
| JB1 + JB2     | 0,67  | $V_m = 0,68V_l + 236,73$ | 0,88                    |
| JB1           | 0,74  | $V_m = 0,88V_l + 18,19$  | 0,76                    |

TABLEAU 5.20 – Résultats de la mesure de la distance de visibilité météorologique sur la base Cerema-AWH-M.

D'après le Tableau 5.20, nous remarquons que pour la première variante (JB1 + JB2), l'algorithme fournit une régression avec  $R^2$  égal à 0,67 et qui est loin de la valeur atteinte par  $R^2_{\text{capteurs}}$  (0,88). La Figure 5.12 montre la distance de visibilité météorologique prédite en fonction de la visibilité terrain fournie par les capteurs. Le problème ici, c'est que l'algorithme n'arrive pas à extraire les caractéristiques de chaque gamme de brouillard. Il y a un grand écart entre la valeur réelle et la valeur prédite de la visibilité météorologique.

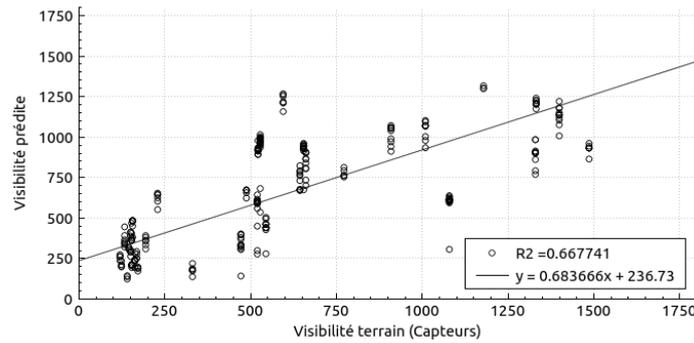


FIGURE 5.12 – Courbe de régression linéaire entre la visibilité prédite et la visibilité terrain pour JB1+JB2.

Comme la fusion des deux types de brouillard n'a pas abouti à une bonne régression, nous avons proposé de les séparer. Nous nous sommes concentrés sur l'analyse de la prédiction des visibilités du brouillard routier (le brouillard fort JB1). La Figure 5.13 présente la distance de visibilité météorologique prédite en fonction de la visibilité terrain des images de JB1. À partir de cette figure, et en la comparant avec les résultats du brouillard routier (visibilités < 400m) de la Figure 5.12, nous remarquons que la prédiction s'est améliorée et elle est devenue plus précise. Cette variante fournit une corrélation de 0,74 qui est pratiquement égale à celle des capteurs physiques (0,76 présentée dans la Figure 5.14).

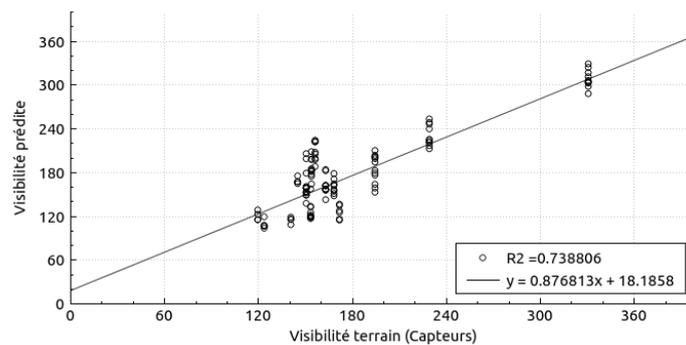


FIGURE 5.13 – Courbe de régression linéaire entre la visibilité prédite et la visibilité terrain pour JB1.

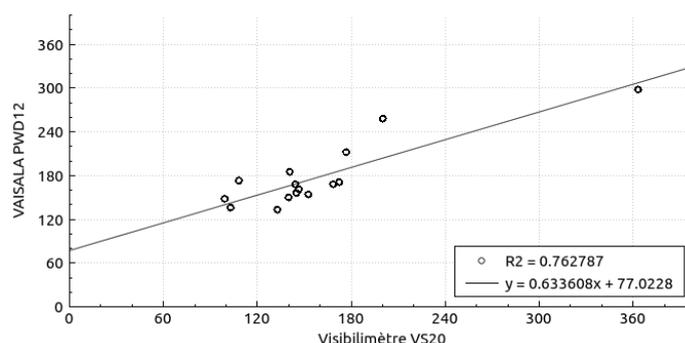


FIGURE 5.14 – Courbe de régression linéaire des deux capteurs de visibilité pour JB1.

## 5.6 Synthèse

Dans ce chapitre, une méthode de classification des conditions météorologiques dégradées par des réseaux de neurones convolutifs a été présentée. Cette classification se base essentiellement sur des caméras routières installées au bord des routes pour la surveillance du trafic. Différents niveaux d'intensité des conditions météorologiques peuvent être différenciés (brouillard fort, brouillard faible, pluie forte, pluie faible). Les données météorologiques sont collectées à partir des capteurs météorologiques dédiés à la mesure physique, d'où leur fiabilité par rapport aux bases de données déjà existantes.

Les réseaux de neurones convolutifs profonds réussissent à classifier les conditions météorologiques dégradées avec des scores de justesse qui atteignent 98,8% pour une classification ne prenant en compte que le type de la condition météorologique (trois classes) et 82,6% en considérant ses intensités pour une classification plus fine (cinq classes) sur la base de données Cerema-AWH. Ces scores chutent fortement en utilisant la méthode de transfert de bases où la base de test utilisée contient des images acquises sur un site totalement différent de celui de la base d'apprentissage.

Après l'analyse des résultats de transfert de bases, il s'est avéré que la condition de la pluie est la condition la plus difficile à transférer. Cela est dû à la particularité de cette condition dans chaque base de données. La question qui se pose ici : qu'est ce qu'on entend par la pluie? L'existence des gouttes de pluie qui se collent sur l'objectif de la caméra, la présence de la chaussée mouillée dans l'image, ou la présence d'images d'inondations?

La méthode de transfert de bases a été appliquée sur des bases simples et des bases mélangées. Le fait de fusionner les bases de données a apporté une amélioration sur certains cas faibles où les scores de justesse passent de 30% à 50% et 60%. Mais le meilleur score a été obtenu sans mélanger les bases avec une valeur égale à 63%. Pour mieux analyser l'effet du transfert de bases sur des bases mélangées ou non, il faut avoir une collection de bases de données plus importante.

Après avoir fait la classification météorologique, et dans le cas où la condition du brouillard est prédite, notre système passe à la phase de mesure pour prédire la distance de visibilité météorologique. Cette prédiction est plus précise dans le cas où il s'agit d'un brouillard routier. Notre système arrive à mesurer la visibilité du brouillard avec un coefficient de corrélation égal à 0,74 qui est proche du coefficient de corrélation des capteurs physiques (0,76).

# Chapitre 6

## Conclusion

### Sommaire

---

|                                    |            |
|------------------------------------|------------|
| <b>6.1 Bilan général</b> . . . . . | <b>108</b> |
| <b>6.2 Perspectives</b> . . . . .  | <b>109</b> |

---

## 6.1 Bilan général

Comme nous avons pu déjà l'indiquer, les systèmes de transport intelligents sont en mesure d'apporter des solutions pour faciliter la mobilité routière et assurer la sécurité associée, notamment pour la gestion dynamique du trafic à l'aide de données issues des véhicules ou de l'infrastructure elle-même. Les systèmes de vision artificielle permettent de délivrer une partie de ces données, pour la surveillance du trafic et la détection d'incidents par exemple. Embarqués dans les véhicules, ces systèmes apportent une aide à la conduite ou sont au cœur même de la chaîne d'acquisition pour la conduite automatisée. Sous des conditions météorologiques dégradées, leur fonctionnement est fortement affecté en raison de la perte de visibilité induite par la présence de brouillard ou de pluie forte par exemple. Les algorithmes automatisés de détection d'obstacles ou de piétons, ou encore de reconnaissance de signalisation, voient leur performance diminuée pouvant avoir un impact en termes de sécurité. Cette problématique constitue à la fois un enjeu pour les autorités publiques ou privées en charge de l'exploitation des réseaux routiers, mais également pour la filière automobile qui s'investit dans la conduite autonome.

Pour ces raisons, il est important de pouvoir qualifier la météorologie locale sur un réseau routier. Outre des stations météorologiques à même de pouvoir délivrer l'information, il serait opportun d'exploiter le gisement important de caméras de surveillance présentes en bord de voie. L'objectif est donc d'adjoindre aux caméras des algorithmes de détermination automatisée des conditions météorologiques. Les caméras de surveillances pourraient donc devenir des capteurs météorologiques une fois cette fonctionnalité en déployant une couche logicielle. Les travaux de cette thèse poursuivent cet objectif.

Utiliser les caméras pour détecter des conditions météorologiques dégradées présente de nombreux intérêts. D'une part, les caméras peuvent s'auto-diagnostiquer, savoir par exemple si leur algorithme de détection de piéton, qui est dégradé par les conditions météorologiques, est fiable ou non. D'autre part, les caméras pourraient servir de capteurs météorologiques bas coût, ce qui permettrait de bénéficier d'un réseau dense de capteurs météorologiques, car de nombreuses caméras de surveillance sont installées sur le territoire.

Dans cette thèse, nous avons utilisé une méthode basée sur les réseaux de neurones convolutifs pour classifier les conditions météorologiques dégradées par les caméras routières. La revue bibliographique du chapitre 2 a montré les performances des méthodes basées sur l'apprentissage profond par rapport aux méthodes plus traditionnelles de l'apprentissage automatique.

Dans le chapitre 5, nous avons évalué la capacité de classification météorologique de divers modèles de réseaux de neurones profonds, en nous fondant sur un benchmark international de 2018. Trois réseaux de la littérature ont été retenus : DenseNet-121, Inception-v4 et ResNet-152. Les modèles examinés dans cette thèse diffèrent par leurs architectures et par leurs techniques d'apprentissage qui ont été présentées dans le chapitre 3. Afin de pouvoir évaluer et étudier les différents réseaux, nous avons constitué des bases de données couplant images et variables météorologiques (intensité de pluie, visibilité météorologique), à la fois par des acquisitions au sein de la plateforme PAVIN "Brouillard-pluie" du Cerema et au col de la Fageole dans le contexte routier de l'autoroute A75. Les caméras utilisées ont les caractéristiques des caméras de surveillance routière. En développant un réseau de neurones instrumental et paramétrable, nommé ClariNet,

nous avons montré que les réseaux de neurones profonds sont plus performants que les réseaux de neurones classiques avec des scores qui peuvent atteindre près de 99% pour classifier le temps météorologique entre conditions normales, conditions de pluie et conditions de brouillard. En exploitant des bases de données publiques images-météo, nous avons également montré qu'un transfert de système de classification des conditions météorologiques d'un site à un autre est possible (apprentissage sur un site et application du modèle sur un autre) mais avec des scores qui chutent fortement, ne dépassant pas 65 %. Ces résultats ont été discutés et la discussion a mis en exergue la très forte sensibilité des réseaux de classification aux données d'entrée, à savoir les bases de données. D'une part, l'apprentissage sur un site qui présente une variabilité assez faible de la scène conduit à de très bons scores lorsqu'il est évalué sur des images provenant du même site. Cela requiert pour un emploi opérationnel d'apprendre sur un hiver avec des images labelisées via des capteurs météorologiques classiques placés momentanément sur le site, ce qui peut être un frein à un déploiement en masse. D'autre part, pour s'affranchir d'un tel usage temporaire de capteurs météorologiques sur chaque site à exploiter, il nous semble que nos travaux doivent être poursuivis en utilisant un nombre plus important de bases de données offrant des variabilités de scènes plus larges, avec une attention particulière sur la fiabilité de la labelisation météorologiques des images. Notons enfin qu'au delà de la classification des conditions météorologiques, un réseau a été développé en fin de chapitre 5 pour effectuer une mesure météorologique de la distance de visibilité météorologique présente en cas de brouillard routier.

## 6.2 Perspectives

Les travaux de recherche réalisés dans le cadre de cette thèse mettent en exergue de nombreuses perspectives à explorer.

Il serait intéressant d'étendre nos travaux à d'autres types de conditions météorologiques dégradées comme l'ajout de la neige. Cela est possible vu la disponibilité des images de la neige dans notre base de données. D'autre part, une classification plus fine des phénomènes atmosphériques pourra être mise en place en distinguant plusieurs intensités de brouillard et de pluie (par exemple pluie forte, pluie moyenne et pluie faible).

D'autres perspectives liées à la méthode de transfert de bases peuvent être discutées. Premièrement, dans l'analyse de transfert de bases, deux types de bases ont été testées (bases simples et bases mélangées). Cependant, en mélangeant les bases, dans un cas les scores de classification ont été améliorés, dans d'autres cas les scores ont chuté. Pour savoir si le fait de mélanger des bases a un intérêt dans la méthode de transfert de bases, il faut le tester sur une collection de bases de données plus importante.

Deuxièmement, la méthode de transfert a été effectuée sur une classification de trois classes, les résultats n'étaient pas satisfaisants. Cela étant dit, pour une classification plus fine, les scores de justesse seront-ils plus faibles? Pour répondre à cette question, il faut trouver des bases de données qui contiennent un nombre de classes plus important avec des conditions météorologiques de différents niveaux (par exemple faible, moyen, fort). Nous rejoignons ici la discussion en fin de section précédente sur la fiabilité de la labelisation météorologiques des images.

Troisièmement, parmi les limitations que la base Cerema-AWH présente, la principale

est l'unique scène capturée par la caméra routière fixe. Cela n'introduit pas la variabilité de scènes que nous rencontrons dans la plupart des bases de données publiques, mais qui sont dépourvues de fiabilité sur la labélisation des images. Pour cela, nous pouvons prévoir de construire une autre base de données à grande échelle ayant un nombre plus important de scènes tout en gardant la fiabilité des données météorologiques qui doivent être enregistrées par des capteurs météorologiques dédiés à la mesure physique et qui sont installées sur le même site d'acquisition d'images. Cela peut être réalisé à travers des stations météorologiques portatives qui ne coûtent pas très chers pour une durée bien déterminée.

Toutes ces perspectives dépendent de la nature de l'application que nous visons. En effet, si nous cherchons la précision, il se peut que la méthode de transfert de bases ne fournisse pas les résultats de classification et de mesure que nous attendons et dans ce cas un apprentissage et un test sur le même site seront exigés.

# Bibliographie

- [1] National and Local Weather Radar, Daily Forecast, Hurricane and information from The Weather Channel. <https://weather.com/>. Consulté le 2019-08-26. 58
- [2] Systèmes de Transports Intelligents, vers plus de sécurité et d'intégration aux territoires durables. <http://www.cerema.fr/fr/innovation-recherche/recherche/equipements/sti-systemes-transports-intelligents-plus-securite>. Consulté le 2019-09-16. 2
- [3] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu. Convolutional Neural Networks for Speech Recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(10) :1533–1545, Oct. 2014. 23
- [4] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, and G. Penn. Applying Convolutional Neural Networks concepts to hybrid NN-HMM model for speech recognition. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4277–4280, Kyoto, Japan, Mar. 2012. IEEE. 23
- [5] AFNOR. Météorologie routière – recueil des données météorologiques et routières. Terminologie. Norme NF P 99-320., Apr. 1998. xiii, 59, 60, 61, 71, 74
- [6] J.-P. Andreu, S. Mayer, K. Gutjahr, and H. Ganster. Measuring Visibility using Atmospheric Transmission and Digital Surface Model. *The 39th Annual Workshop of the Austrian Association for Pattern Recognition (OAGM)*, May 2015. 30
- [7] J. Andrey, B. Mills, M. Leahy, and J. Suggett. Weather as a Chronic Hazard for Road Transportation in Canadian Cities. *Natural Hazards*, 28(2) :319–343, Mar. 2003. 5
- [8] A. Angelova, A. Krizhevsky, V. Vanhoucke, A. Ogale, and D. Ferguson. Real-Time Pedestrian Detection with Deep Network Cascades. In *Proceedings of the British Machine Vision Conference 2015*, pages 32.1–32.12, Swansea, 2015. British Machine Vision Association. 22, 61
- [9] A. Armitage. Neural Networks in Measurement and Control. *Measurement and Control*, 28(7) :208–215, Sept. 1995. 28
- [10] M. Attari, M. Heniche, and F. Boudjema. A two dimensional intelligent calibration of an ion sensor. In *IEEE Instrumentation and Measurement Technology Conference and IMEKO Tec*, volume 2, pages 788–791. IEEE, 1996. 28

- 
- [11] N. Audebert. *Classification de données massives de télédétection*. PhD thesis, Oct. 2018. [xi](#), [49](#), [50](#), [51](#), [54](#), [55](#)
- [12] Y. Avrithis, Y. Kalantidis, G. Toliás, and E. Spyrou. Retrieving landmark and non-landmark images from community photo collections. In *Proceedings of the international conference on Multimedia - MM '10*, page 153, Firenze, Italy, 2010. ACM Press. [65](#)
- [13] R. Babari. *Estimation des conditions de visibilité météorologique par caméras routières*. phdthesis, Université Paris-Est, Apr. 2012. [60](#)
- [14] R. Babari, N. Hautière, E. Dumont, J. Papelard, and N. Paparoditis. Computer vision for the remote sensing of atmospheric visibility. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 219–226, Nov. 2011. [30](#)
- [15] R. Babari, N. Hautière, r. Dumont, R. Brémond, and N. Paparoditis. A model-driven approach to estimate atmospheric visibility with ordinary cameras. *Atmospheric Environment*, 45(30) :5316–5324, Sept. 2011. [30](#)
- [16] R. Babari, N. Hautière, r. Dumont, N. Paparoditis, and J. Misener. Visibility monitoring using conventional roadside cameras – Emerging applications. *Transportation Research Part C : Emerging Technologies*, 22 :17–28, June 2012. [30](#)
- [17] R. Belaroussi and D. Gruyer. Convergence of a Traffic Signs-Based Fog Density Model. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 783–787, Sept. 2015. [6](#), [25](#)
- [18] Y. Bengio. Practical recommendations for gradient-based training of deep architectures. *Neural networks : Tricks of the trade.*, pages 437–478, June 2012. [43](#)
- [19] A. Bernieri, G. Betta, C. Liguori, and A. Losi. Neural networks and pseudo-measurements for real-time monitoring of distribution systems. *IEEE transactions on instrumentation and measurement*, 45(2) :645–650, 1996. [28](#)
- [20] A. Bernieri, G. Betta, A. Pietrosanto, and C. Sansone. A neural network approach to instrument fault detection and isolation. In *IEEE Instrumentation and Measurement Technology Conference*, pages 139–144. IEEE, 1994. [28](#)
- [21] M. J. A. Berry and G. S. Linoff. *Data Mining Techniques : For Marketing, Sales, and Customer Relationship Management*. John Wiley & Sons, Apr. 2004. [14](#)
- [22] M. Bianchini and F. Scarselli. On the complexity of shallow and deep neural network classifiers. *Computational Intelligence*, page 6, 2014. [40](#)
- [23] S. Bianco, R. Cadene, L. Celona, and P. Napoletano. Benchmark Analysis of Representative Deep Neural Network Architectures. *IEEE Access*, 6 :64270–64277, 2018. [83](#), [91](#)

- [24] S. Biondo, E. Ramos, M. Deiros, J. M. Ragué, J. De Oca, P. Moreno, L. Farran, and E. Jaurrieta. Prognostic factors for mortality in left colonic peritonitis : a new scoring system. *Journal of the American College of Surgeons*, 191(6) :635–642, Dec. 2000. [14](#)
- [25] C. Blake and C. Merz. UCI repository of machine learning databases, 1998. [10](#)
- [26] J. Bossu, N. Hautiere, and J. P. Tarel. Rain or Snow Detection in Image Sequences through use of a Histogram of Orientation of Streaks. *International Journal of Computer Vision*, 93(3) :pp. 348–367, Jan. 2011. [25](#)
- [27] L. Bottou. Stochastic Gradient Descent Tricks. *Neural networks : Tricks of the trade*, 7700 :421–436, Jan. 2012. [43](#)
- [28] C. R. Boyd, M. A. Tolson, and W. S. Copes. Evaluating Trauma Care : The TRISS Method. *Journal of Trauma and Acute Care Surgery*, 27(4) :370, Apr. 1987. [14](#)
- [29] M. J. Brady. Biometric recognition using a classification neural network. *U.S. Patent No 5,892,838*, 1999. [23](#)
- [30] A. Broggi, K. Ikeuchi, and C. Thorpe. Special issue on vision applications and technology for intelligent vehicles : Part II—Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 1(3) :133, 2000. [5](#)
- [31] L. Caltagirone, S. Scheidegger, L. Svensson, and M. Wahde. Fast LIDAR-based Road Detection Using Fully Convolutional Neural Networks. *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1019–1024, Mar. 2017. [22](#)
- [32] F. Carretas, F. Wagner, and F. M. Janeiro. Atmospheric visibility and Angström exponent measurements through digital photography. *Measurement*, 64 :147–156, Mar. 2015. [30](#)
- [33] A. Carullo, F. Ferraris, S. Graziani, U. Grimaldi, and M. Parvis. Ultrasonic distance sensor improvement using a two-level neural-network. *IEEE Transactions on Instrumentation and Measurement*, 45(2) :677–682, 1996. [28](#)
- [34] C.-C. Chang and C.-J. Lin. LIBSVM a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3) :27, 2011. [19](#)
- [35] S. A. Chatzichristofis and Y. S. Boutalis. CEDD Color and Edge Directivity Descriptor : A Compact Descriptor for Image Indexing and Retrieval. In A. Gasteratos, M. Vincze, and J. K. Tsotsos, editors, *Computer Vision Systems*, Lecture Notes in Computer Science, pages 312–322. Springer Berlin Heidelberg, 2008. [26](#)
- [36] S. A. Chatzichristofis and Y. S. Boutalis. FCTH Fuzzy Color and Texture Histogram - A Low Level Feature for Accurate Image Retrieval. In *2008 Ninth International Workshop on Image Analysis for Multimedia Interactive Services*, pages 191–196, May 2008. [26](#)
- [37] a. A. Chekima and a. J. A. Dargham. Iris recognition using self-organizing neural network. In *Student Conference on Research and Development*, pages 169–172, July 2002. [24](#)

- [38] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2147–2156, 2016. [21](#), [61](#)
- [39] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia. Multi-View 3D Object Detection Network for Autonomous Driving. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1907–1915, Nov. 2016. [21](#)
- [40] Z. Chen, F. Yang, A. Lindner, G. Barrenetxea, and M. Vetterli. How is the weather : Automatic inference from images. In *2012 19th IEEE International Conference on Image Processing*, pages 1853–1856, Sept. 2012. [26](#)
- [41] A. K. Choudhary, J. A. Harding, and M. K. Tiwari. Data mining in manufacturing : a review based on the kind of knowledge. *Journal of Intelligent Manufacturing*, 20(5) :501, July 2008. [18](#)
- [42] W. I. Christensen and R. A. Bryson. An investigation of the potential of component analysis for weather classification. *Monthly Weather Review*, 94(12) :697–709, Dec. 1966. [5](#)
- [43] W. Chu, X. Zheng, and D. Ding. Image2Weather A Large-Scale Image Dataset for Weather Property Estimation. In *2016 IEEE Second International Conference on Multimedia Big Data (BigMM)*, pages 137–144, Apr. 2016. [96](#)
- [44] W.-T. Chu, K.-C. Ho, and A. Borji. Visual Weather Temperature Prediction. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 234–241, 2018. [25](#), [62](#), [63](#)
- [45] W.-T. Chu, X.-Y. Zheng, and D.-S. Ding. Camera as weather sensor : Estimating weather information from single images. *Journal of Visual Communication and Image Representation*, 46 :233–249, 2017. [xi](#), [26](#), [61](#), [62](#), [65](#), [66](#), [67](#)
- [46] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber. A committee of neural networks for traffic sign classification. In *The 2011 International Joint Conference on Neural Networks*, pages 1918–1921, July 2011. [25](#)
- [47] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber. Multi-column deep neural network for traffic sign classification. *Neural Networks*, 32 :333–338, Aug. 2012. [25](#)
- [48] D. C. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Mitosis detection in breast cancer histology images with deep neural networks. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 411–418. Springer, 2013. [20](#)
- [49] D. C. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhuber. Deep, Big, Simple Neural Nets for Handwritten Digit Recognition. *Neural Computation*, 22(12) :3207–3220, Sept. 2010. [23](#)
- [50] D. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *Advances in neural information processing systems*, pages 2843–2851, 2012. [20](#)

- [51] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber. Convolutional Neural Network Committees for Handwritten Character Classification. In *2011 International Conference on Document Analysis and Recognition*, pages 1135–1139, Sept. 2011. [23](#)
- [52] M. Colomb, K. Hirech, P. André, J. Boreux, P. Lacôte, and J. Dufour. An innovative artificial fog production device improved in the European project “FOG”. *Atmospheric Research*, 87(3-4) :242–251, Mar. 2008. [61](#), [64](#)
- [53] A. Cord and N. Gimonet. Detecting Unfocused Raindrops : In-Vehicle Multipurpose Cameras. *Robotics & Automation Magazine, IEEE*, 21 :49–56, Mar. 2014. [6](#), [25](#)
- [54] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3) :273–297, Sept. 1995. [35](#)
- [55] T. M. Cover and P. E. Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1) :21–27, 1967. [18](#)
- [56] S. P. Curram and J. Mingers. Neural Networks, Decision Tree Induction and Discriminant Analysis : an Empirical Comparison. *Journal of the Operational Research Society*, 45(4) :440–450, Apr. 1994. [22](#)
- [57] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4) :303–314, Dec. 1989. [40](#)
- [58] G. E. Dahl, D. Yu, L. Deng, and A. Acero. Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(1) :30–42, Jan. 2012. [23](#)
- [59] K. Dahmane, N. E. B. Amara, P. Duthon, F. Bernardin, M. Colomb, and F. Chausse. The Cerema pedestrian database : A specific database in adverse weather conditions to evaluate computer vision pedestrian detectors. In *2016 7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, pages 472–477, Dec. 2016. [61](#), [62](#), [64](#), [67](#)
- [60] K. Dahmane, P. Duthon, F. Bernardin, M. Colomb, C. Blanc, and F. Chausse. Weather classification with traffic surveillance cameras. In *25th ITS World Congress 2018*, 2018. [82](#)
- [61] P. Daponte, D. Grimaldi, A. Piccolo, and D. Villacci. A neural diagnostic system for the monitoring of transformer heating. *Measurement*, 18(1) :35–46, 1996. [28](#)
- [62] J. G. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *JOSA A*, 2(7) :1160–1169, July 1985. [26](#)
- [63] A. de la Escalera, J. M. Armingol, and M. Mata. Traffic sign recognition and analysis for intelligent vehicles. *Image and Vision Computing*, 21(3) :247–258, Mar. 2003. [25](#)

- [64] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F. F. Li. ImageNet a Large-Scale Hierarchical Image Database. *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255, June 2009. [36](#), [91](#)
- [65] L. Deng. The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]. *IEEE Signal Processing Magazine*, 29(6) :141–142, Nov. 2012. [84](#)
- [66] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection : A benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009*, pages 304–311, June 2009. [61](#)
- [67] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian Detection : An Evaluation of the State of the Art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4) :743–761, Apr. 2012. [61](#)
- [68] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014. [27](#)
- [69] B. Dong and X. Wang. Comparison deep learning method to traditional methods using for network intrusion detection. In *2016 8th IEEE International Conference on Communication Software and Networks (ICCSN)*, pages 581–585. IEEE, 2016. [15](#), [18](#)
- [70] X. Du, M. El-Khamy, J. Lee, and L. S. Davis. Fused DNN : A deep neural network fusion approach to fast and robust pedestrian detection. *2017 IEEE winter conference on applications of computer vision (WACV)*, pages 953–961, 2017. [22](#)
- [71] E. Dumont. *Caractérisation, modélisation et simulation des effets visuels du brouillard pour l'utilisateur de la route*. phdthesis, Université René Descartes - Paris V, Nov. 2002. [60](#)
- [72] V. Dumoulin and F. Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv :1603.07285*, Mar. 2016. [xi](#), [45](#)
- [73] A. Durg, W. V. Stoecker, J. P. Cookson, S. E. Umbaugh, and R. H. Moss. Identification of variegated coloring in skin tumors : neural network vs. rule-based induction methods. *IEEE Engineering in Medicine and Biology Magazine*, 12(3) :71–74, Sept. 1993. [24](#)
- [74] P. Duthon. *Descripteurs d'images pour les systèmes de vision routiers en situations atmosphériques dégradées et caractérisation des hydrométéores*. phdthesis, Université Clermont Auvergne, Dec. 2017. [xi](#), [6](#), [60](#), [69](#), [70](#), [72](#), [73](#), [87](#), [90](#)
- [75] P. Duthon, F. Bernardin, F. Chausse, and M. Colomb. Benchmark for the robustness of image features in rainy conditions. *Machine Vision and Applications*, 29 :915–927, 2018. [62](#)
- [76] M. Elhoseiny, S. Huang, and A. Elgammal. Weather classification with deep convolutional neural networks. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 3349–3353, Sept. 2015. [27](#), [82](#)

- [77] A. Elkamel, S. Abdul-Wahab, W. Bouhamra, and E. Alper. Measurement and prediction of ozone levels around a heavily industrialized area : a neural network approach. *Advances in environmental research*, 5(1) :47–59, 2001. [29](#)
- [78] H. Eren, C. C. Fung, and K. W. Wong. An application of artificial neural network for prediction of densities and particle size distributions in mineral processing industry. In *IEEE Instrumentation and Measurement Technology Conference Sensing, Processing, Networking, IMTC Proceedings*, volume 2, pages 1118–1121. IEEE, 1997. [28](#)
- [79] A. Farhat and S. Mestiri. Détection de la défaillance des entreprises tunisiennes par la régression logistique semi paramétrique et les réseaux de neurones. 2013. [23](#)
- [80] D. A. Freedman. *Statistical models : theory and practice*. cambridge university press, 2009. [14](#)
- [81] J. Fritsch, T. Kuhn, and A. Geiger. A new performance measure and evaluation benchmark for road detection algorithms. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 1693–1700, The Hague, Netherlands, Oct. 2013. IEEE. [61](#)
- [82] K. Fukushima. Neocognitron A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4) :193–202, Apr. 1980. [34](#)
- [83] K. Fukushima. Neocognitron for handwritten digit recognition. *Neurocomputing*, 51 :161–180, Apr. 2003. [23](#)
- [84] J. W. Gardner, E. L. Hines, and C. Pang. Detection of vapours and odours from a multisensor array using pattern recognition : self-organising adaptive resonance techniques. *Measurement and control*, 29(6) :172–178, 1996. [28](#)
- [85] K. Garg and S. K. Nayar. Detection and removal of rain from videos. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–528–I–535 Vol.1, June 2004. [25](#)
- [86] K. Garg and S. K. Nayar. When does a camera see rain? In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1067–1074 Vol. 2, Oct. 2005. [25](#)
- [87] K. Garg and S. K. Nayar. Vision and rain. *International Journal of Computer Vision*, 75(1) :3, 2007. [25](#)
- [88] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, June 2012. [61](#)
- [89] M. Gheisari, G. Wang, and M. Z. A. Bhuiyan. A survey on deep learning in big data. In *2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, volume 2, pages 173–180. IEEE, 2017. [19](#)

- 
- [90] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 27
- [91] D. Glasner, P. Fua, T. Zickler, and L. Zelnik-Manor. Hot or Not : Exploring Correlations Between Appearance and Temperature. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3997–4005, 2015. 63
- [92] X. Glorot, A. Bordes, and Y. Bengio. Deep Sparse Rectifier Neural Networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 315–323, June 2011. 39, 40
- [93] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood Components Analysis. *Advances in Neural Information Processing Systems*, 17 :513–520, 2005. 17
- [94] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. 43
- [95] M. Grafmüller, J. Beyerer, and K. Kroschel. Decision Tree Classifier for Character Recognition Combining Support Vector Machines and Artificial Neural Networks. *Proceedings of SPIE - The International Society for Optical Engineering*, 7799, Aug. 2010. 15
- [96] A. Graves, A. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649, May 2013. 23
- [97] J. C. V. Guerra, Z. Khanam, S. Ehsan, R. Stolkin, and K. McDonald-Maier. Weather Classification : A new multi-class dataset, data augmentation approach and comprehensive evaluations of Convolutional Neural Networks. In *2018 NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, pages 305–310, Aug. 2018. 27
- [98] T. Hagiwara, S. Fujita, and K. Kizaka. Assessment of visibility on Roads under Snow Conditions using Digital Images. *Journal of the Illuminating Engineering Institute of Japan*, 86 :302–302, Jan. 2002. 30
- [99] E.-H. S. Han, G. Karypis, and V. Kumar. Text categorization using weight adjusted k-nearest neighbor classification. In *Pacific-asia conference on knowledge discovery and data mining*, pages 53–65. Springer, 2001. 18
- [100] F. E. Harrell. Regression modeling strategies. *BIOS*, 330 :2018, 2017. 14
- [101] N. Hautiere, D. Aubert, . Dumont, and J. Tarel. Experimental Validation of Dedicated Methods to In-Vehicle Estimation of Atmospheric Visibility Distance. *IEEE Transactions on Instrumentation and Measurement*, 57(10) :2218–2225, Oct. 2008. 30
- [102] N. Hautiere and A. Boubezoul. Combination of Roadside and In-vehicle Sensors for Extensive Visibility Range Monitoring. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 388–393, Sept. 2009. 30

- [103] N. Hautière, J. Bossu, E. Bigorgne, and D. Aubert. Vers un système de détection et caractérisation par caméra de conditions météorologiques critiques pour la sécurité routière. *Traitement du signal*, page 29, 2011. [5](#)
- [104] N. Hautière, J.-P. Tarel, J. Lavenant, and D. Aubert. Automatic fog detection and estimation of visibility distance through use of an onboard camera. *Machine Vision and Applications*, 17(1) :8–20, 2006. [30](#), [60](#)
- [105] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, Dec. 2015. [xiii](#), [49](#), [50](#), [51](#), [86](#), [91](#)
- [106] M. Helu, D. Libes, J. Lubell, K. Lyons, and K. C. Morris. Enabling smart manufacturing technologies for decision-making support. In *ASME 2016 international design engineering technical conferences and computers and information in engineering conference*, pages V01BT02A035–V01BT02A035. American Society of Mechanical Engineers, 2016. [18](#)
- [107] G. Hinton, L. Deng, D. Yu, G. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, B. Kingsbury, and T. Sainath. Deep Neural Networks for Acoustic Modeling in Speech Recognition. *IEEE Signal Processing Magazine*, 29, Nov. 2012. [16](#), [23](#)
- [108] G. E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7) :1527–1554, 2006. [18](#)
- [109] S. Hochreiter and J. Schmidhuber. Long Short-term Memory. *Neural computation*, 9 :1735–80, Dec. 1997. [35](#)
- [110] W. Homenda and W. Lesinski. Decision Trees and Their Families in Imbalanced Pattern Recognition : Recognition with and without Rejection. In K. Saeed and V. Snášel, editors, *Computer Information Systems and Industrial Management*, volume 8838, pages 219–230. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. [15](#)
- [111] K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2) :251–257, Jan. 1991. [40](#)
- [112] G. Horvath. Neural networks from the perspective of measurement systems. In *Proceedings of the 20th IEEE Instrumentation Technology Conference (Cat. No. 03CH37412)*, volume 2, pages 1102–1107. IEEE, 2003. [28](#)
- [113] J. Hosang, M. Omran, R. Benenson, and B. Schiele. Taking a Deeper Look at Pedestrians. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4073–4082, Jan. 2015. [22](#), [61](#)
- [114] C. Hu, G. Jain, P. Zhang, C. Schmidt, P. Gomadam, and T. Gorka. Data-driven method based on particle swarm optimization and k-nearest neighbor regression for estimating capacity of lithium-ion battery. *Applied Energy*, 129 :49–55, 2014. [18](#)
- [115] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely Connected Convolutional Networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, Aug. 2016. [xiii](#), [54](#), [55](#), [86](#), [91](#)

- [116] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. Cheng-Yue, F. Mujica, A. Coates, and A. Y. Ng. An Empirical Evaluation of Deep Learning on Highway Driving. *arXiv preprint arXiv :1504.01716*, Apr. 2015. [21](#), [22](#)
- [117] S. Hwang, J. Park, N. Kim, Y. Choi, and I. So Kweon. Multispectral Pedestrian Detection : Benchmark Dataset and Baseline. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1037–1045, 2015. [61](#)
- [118] N. P. Hyslop. Impaired visibility : the air pollution people see. *Atmospheric Environment*, 43(1) :182–195, Jan. 2009. [5](#)
- [119] S. B. Imandoust and M. Bolandraftar. Application of k-nearest neighbor (knn) approach for predicting economic events : Theoretical background. *International Journal of Engineering Research and Applications*, 3(5) :605–610, 2013. [18](#)
- [120] M. Islam, N. Jacobs, H. Wu, and R. Souvenir. Images+ weather : Collection, validation, and refinement. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop on Ground Truth*, volume 6, page 2, 2013. [62](#)
- [121] A. G. Ivakhnenko, V. G. Lapa, United States, and Joint Publications Research Service. *Cybernetic predicting devices*. CCM Information Corp., New York, 1965. OCLC : 23815433. [34](#)
- [122] N. Jacobs, W. Burgin, R. Speyer, D. Ross, and R. Pless. Adventures in archiving and using three years of webcam images. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 39–46. IEEE, 2009. [26](#), [62](#)
- [123] N. Jacobs, N. Roman, and R. Pless. Consistent temporal variations in many outdoor scenes. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–6. IEEE, 2007. [26](#), [27](#), [62](#)
- [124] J. Jin, K. Fu, and C. Zhang. Traffic Sign Recognition With Hinge Loss Trained Convolutional Neural Networks. *IEEE Transactions on Intelligent Transportation Systems*, 15(5) :1991–2000, Oct. 2014. [25](#)
- [125] G. H. John and P. Langley. Estimating continuous distributions in Bayesian classifiers. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 338–345. Morgan Kaufmann Publishers Inc., 1995. [19](#)
- [126] M. Johnson. Neural networks in advanced instrument design. *Measurement and Control*, 29(4) :101–105, 1996. [28](#)
- [127] I. Karouche and S. Bouhenni. *Classification algorithms for plant disease forecasting*. PhD thesis, May 2017. [18](#)
- [128] J. Khan, J. S. Wei, M. Ringnér, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 7(6) :673–679, June 2001. [24](#)

- [129] K.-J. Kim and S.-B. Cho. Prediction of colon cancer using an evolutionary neural network. *Neurocomputing*, 61 :361–379, Oct. 2004. [24](#)
- [130] K. W. Kim. Estimation of visibility using a visual image. *Environmental Monitoring and Assessment*, 187(3) :66, Mar. 2015. [30](#)
- [131] J. Kivinen, C. Williams, and N. Heess. Visual boundary prediction : A deep neural prediction network and quality dissection. In *Artificial Intelligence and Statistics*, pages 512–521, 2014. [20](#)
- [132] M. Kologlu, D. Elker, H. Altun, and I. Sayek. Validation of MPI and PIA II in two different groups of patients with secondary peritonitis. *Hepato-Gastroenterology*, 48(37) :147–151, Feb. 2001. [14](#)
- [133] S. Kombrink, T. Mikolov, M. Karafiát, and L. Burget. Recurrent neural network based language modeling in meeting recognition. In *Twelfth annual conference of the international speech communication association*, 2011. [23](#)
- [134] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. [20](#), [27](#), [36](#)
- [135] L. Kuang, L. Yang, and Y. Liao. An integration framework on cloud for cyber physical social systems big data. *IEEE Transactions on Cloud Computing*, 2015. [19](#)
- [136] A. Kuehnle. Method and system for weather condition detection with image-based road characterization. *U.S. Patent No 8,436,902*, May 2013. [25](#)
- [137] H. Kurihata, T. Takahashi, I. Ide, Y. Mekada, H. Murase, Y. Tamatsu, and T. Miyahara. Rainy weather recognition from in-vehicle camera images for driver assistance. In *IEEE Proceedings. Intelligent Vehicles Symposium, 2005.*, pages 205–210, June 2005. [6](#), [25](#)
- [138] P. Lade, R. Ghosh, and S. Srinivasan. Manufacturing Analytics and Industrial Internet of Things. *IEEE Intelligent Systems*, 32(3) :74–79, May 2017. [18](#)
- [139] P.-Y. Laffont, Z. Ren, X. Tao, C. Qian, and J. Hays. Transient attributes for high-level understanding and editing of outdoor scenes. *ACM Transactions on Graphics (TOG)*, 33(4) :149, 2014. [62](#), [63](#)
- [140] G. H. Landeweerd, T. Timmers, E. S. Gelsema, M. Bins, and M. R. Halie. Binary tree versus single level tree classification of white blood cells. *Pattern Recognition*, 16(6) :571–577, Jan. 1983. [15](#)
- [141] Y. Le Cun. Learning Process in an Asymmetric Threshold Network. In E. Bienenstock, F. F. Soulié, and G. Weisbuch, editors, *Disordered Systems and Biological Organization*, NATO ASI Series, pages 233–240. Springer Berlin Heidelberg, 1986. [41](#)

- [142] J. R. Le Gall, S. Lemeshow, and F. Saulnier. A new Simplified Acute Physiology Score (SAPS II) based on a European/North American multicenter study. *JAMA*, 270(24) :2957–2963, Dec. 1993. [14](#)
- [143] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4) :541–551, Dec. 1989. [35](#)
- [144] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324, Nov. 1998. [35](#), [43](#), [46](#), [49](#)
- [145] J. Lee. Measurement of machine performance degradation using a neural network model. *Computers in Industry*, 30(3) :193–209, 1996. [28](#)
- [146] M. Leshno and Y. Spector. Neural network prediction analysis : The bankruptcy case. *Neurocomputing*, 10(2) :125–147, Mar. 1996. [23](#)
- [147] J. Lettvin, H. Maturana, W. McCulloch, and W. Pitts. What the Frog’s Eye Tells the Frog’s Brain. *Proceedings of the IRE*, 47(11) :1940–1951, Nov. 1959. [36](#)
- [148] C. Li and P. Peng. Visibility measurement using multi-angle forward scattering by liquid droplets. *Measurement Science and Technology*, 23(10) :105802, Aug. 2012. [60](#)
- [149] P. Li, D. Wang, L. Wang, and H. Lu. Deep visual tracking : Review and experimental comparison. *Pattern Recognition*, 76 :323–338, 2018. [16](#), [18](#)
- [150] P. Li, Y. Zang, C. Wang, J. Li, M. Cheng, L. Luo, and Y. Yu. Road network extraction via deep learning and line integral convolution. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1599–1602, July 2016. [22](#)
- [151] S. Li, H. Fu, and W.-L. Lo. Meteorological Visibility Evaluation on Webcam Weather Image Using Deep Learning Features. *International Journal of Computer Theory and Engineering*, 9 :455–461, Jan. 2017. [30](#)
- [152] W. Li, R. Zhao, T. Xiao, and X. Wang. DeepReID Deep Filter Pairing Neural Network for Person Re-identification. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 152–159, June 2014. [22](#), [61](#)
- [153] D. Lin, C. Lu, H. Huang, and J. Jia. RSCM Region selection and concurrency model for multi-class weather recognition. *IEEE Transactions on Image Processing*, 26(9) :4154–4167, 2017. [xi](#), [27](#), [61](#), [62](#), [64](#), [65](#), [67](#), [96](#)
- [154] H. W. Lin, M. Tegmark, and D. Rolnick. Why does deep and cheap learning work so well? *Journal of Statistical Physics*, 168(6) :1223–1247, Sept. 2017. [41](#)
- [155] S. Linnainmaa. *The representation of the cumulative rounding error of an algorithm as a taylor expansion of the local rounding errors*. Master’s thesis, 1970. [35](#)
- [156] J. Liu, S. Zhang, S. Wang, and D. N. Metaxas. Multispectral Deep Neural Networks for Pedestrian Detection. *arXiv preprint arXiv :1611.02644*, Nov. 2016. [22](#), [61](#)

- [157] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi. A survey of deep neural network architectures and their applications. *Neurocomputing*, 234 :11–26, Apr. 2017. [16](#)
- [158] Y. Liu. Calibrating an industrial microwave six-port instrument using the artificial neural network technique. *IEEE transactions on instrumentation and measurement*, 45(2) :651–656, 1996. [28](#)
- [159] C. Lu, D. Lin, J. Jia, and C. Tang. Two-Class Weather Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12) :2510–2524, Dec. 2017. [27](#)
- [160] C. Lu, D. Lin, J. Jia, and C.-K. Tang. Two-Class Weather Classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3718–3725, 2014. [25](#), [27](#), [62](#)
- [161] Y. Lu, X. Xu, and J. Xu. Development of a Hybrid Manufacturing Cloud. *Journal of Manufacturing Systems*, 33(4) :551–566, Oct. 2014. [18](#)
- [162] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang. The Expressive Power of Neural Networks : A View from the Width. *Advances in neural information processing systems*, pages 6231–6239, Sept. 2017. [41](#)
- [163] P. Luo, Y. Tian, X. Wang, and X. Tang. Switchable Deep Network for Pedestrian Detection. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 899–906, June 2014. [22](#), [61](#)
- [164] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang. Learning traffic as images : a deep convolutional neural network for large-scale transportation network speed prediction. *Sensors*, 17(4) :818, 2017. [21](#)
- [165] C. J. Maestre. Processus essai-erreur et pratique de la responsabilité politique. *Économie rurale*, 127(1) :15–19, 1978. [41](#)
- [166] E. Manitsas, R. Singh, B. C. Pal, and G. Strbac. Distribution system state estimation using an artificial neural network approach for pseudo measurement modeling. *IEEE Transactions on Power Systems*, 27(4) :1888–1896, 2012. [29](#)
- [167] J. C. Marshall, D. J. Cook, N. V. Christou, G. R. Bernard, C. L. Sprung, and W. J. Sibbald. Multiple organ dysfunction score : a reliable descriptor of a complex clinical outcome. *Critical Care Medicine*, 23(10) :1638–1652, Oct. 1995. [14](#)
- [168] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4) :115–133, Dec. 1943. [34](#)
- [169] C. C. T. Mendes, V. Frémont, and D. F. Wolf. Exploiting fully convolutional neural networks for fast road detection. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3174–3179, May 2016. [22](#), [61](#)
- [170] Meng Joo Er, Shiqian Wu, Juwei Lu, and Hock Lye Toh. Face recognition with radial basis function (RBF) neural networks. *IEEE Transactions on Neural Networks*, 13(3) :697–710, May 2002. [23](#)

- [171] J. Mercer. Functions of Positive and Negative Type, and their Connection with the Theory of Integral Equations. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 209 :415–446, 1909. [16](#)
- [172] H. Mhaskar, Q. Liao, and T. A. Poggio. When and Why Are Deep Networks Better Than Shallow Ones? In *AAAI*, 2017. [40](#)
- [173] D. Michie, D. J. Spiegelhalter, C. C. Taylor, and J. Campbell, editors. *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, Upper Saddle River, NJ, USA, 1994. [22](#)
- [174] W. E. K. Middleton. Vision through the Atmosphere. In J. Bartels, editor, *Geophysics II*, number 10 / 48 in *Encyclopedia of Physics*, pages 254–287. Springer Berlin Heidelberg, 1957. [60](#)
- [175] R. P. Mihail, S. Workman, Z. Bessinger, and N. Jacobs. Sky segmentation in the wild : An empirical study. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–6. IEEE, 2016. [63](#)
- [176] L. Monostori, A. Markus, H. Van Brussel, and E. Westkämpfer. Machine Learning Approaches to Manufacturing. *CIRP Annals*, 45(2) :675–712, Jan. 1996. [18](#)
- [177] D. T. Morris and D. Kalles. Decision trees and domain knowledge in pattern recognition. In E. S. Gelsema and L. S. Kanal, editors, *Machine Intelligence and Pattern Recognition*, volume 16 of *Pattern Recognition in Practice IV*, pages 25–36. North-Holland, Jan. 1994. [15](#)
- [178] J. K. Mui and K. Fu. Automated classification of nucleated blood cells using a binary tree classifier. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-2(5) :429–443, Sept. 1980. [15](#)
- [179] G. P. Naidu and K. Govinda. Bankruptcy prediction using neural networks. In *2018 2nd International Conference on Inventive Systems and Control (ICISC)*, pages 248–251, Jan. 2018. [23](#)
- [180] S. G. Narasimhan and S. K. Nayar. Vision and the Atmosphere. *International Journal of Computer Vision*, 48(3) :233–254, July 2002. [25](#)
- [181] M. Negru and S. Nedevschi. Image based fog detection and visibility estimation for driving assistance systems. In *2013 IEEE 9th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 163–168, Sept. 2013. [30](#)
- [182] J. A. Nystuen and H. D. Selsor. Weather Classification Using Passive Acoustic Drifters. *Journal of Atmospheric and Oceanic Technology*, 14(3) :656–666, June 1997. [5](#)
- [183] G. L. Oliveira, W. Burgard, and T. Brox. Efficient deep models for monocular road segmentation. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4885–4891, Oct. 2016. [22](#), [61](#)

- [184] ONISR. Grands themes de la securite routiere en France, 2008. [5](#)
- [185] S. K. Palei and S. K. Das. Logistic regression model for prediction of roof fall risks in bord and pillar workings in coal mines : An approach. *Safety Science*, 47(1) :88–96, 2009. [14](#)
- [186] S. Park, D. Park, and C. Won. Core experiments on mpeg-7 edge histogram descriptor, 2000. [26](#)
- [187] M. Pavlic. Method for fog detection. *U.S. Patent No 9,384,401*, July 2016. [6](#), [25](#)
- [188] M. Pavlic, G. Rigoll, and S. Ilic. Classification of images in fog and fog-free scenes for use in vehicles. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 481–486, June 2013. [6](#), [25](#)
- [189] K. R. Popper. *Evolutionary Epistemology, Rationality, and the Sociology of Knowledge*. Open Court Publishing, 1987. [41](#)
- [190] J. R. Quinlan. *C4. 5 : programs for machine learning*. Elsevier, 2014. [19](#)
- [191] E. Rabiei, U. Haberlandt, M. Sester, and D. Fitzner. Rainfall estimation using moving cars as rain gauges. *Hydrology and Earth System Sciences*, 17(11) :4701–4712, Nov. 2013. [82](#)
- [192] D. Ribli, L. Dobos, and I. Csabai. Galaxy shape measurement with convolutional neural networks. *Monthly Notices of the Royal Astronomical Society*, 489(4) :4847–4859, 2019. [29](#)
- [193] P. Rämä and A. Schirokoff. Effects of weather-controlled variable speed limits on injury accidents. *European Congress on Intelligent Transportation Systems and Services, 4th*, 2004. [5](#)
- [194] C. Robert, C. Guilpin, and A. Limoge. Comparison between conventional and neural network classifiers for rat sleep-wake stage discrimination. *Neuropsychobiology*, 35(4) :221–225, 1997. [22](#)
- [195] F. Rosenblatt. *The Perceptron, a Perceiving and Recognizing Automaton Project Para*. Cornell Aeronautical Laboratory, 1957. [34](#)
- [196] M. Roser and F. Moosmann. Classification of weather situations on single color images. In *2008 IEEE Intelligent Vehicles Symposium*, pages 798–803, June 2008. [6](#), [25](#), [26](#), [61](#), [82](#)
- [197] H. R. Roth, L. Lu, J. Liu, J. Yao, A. Seff, K. Cherry, L. Kim, and R. M. Summers. Improving computer-aided detection using convolutional neural networks and random view aggregation. *IEEE transactions on medical imaging*, 35(5) :1170–1181, 2016. [20](#)
- [198] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088) :533–536, Oct. 1986. [35](#)

- 
- [199] D. E. Rumelhart and J. L. McClelland. Learning Internal Representations by Error Propagation. In *Parallel Distributed Processing : Explorations in the Microstructure of Cognition : Foundations*. MITP, 1987. [41](#)
- [200] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe a database and web-based tool for image annotation. *International journal of computer vision*, 77(1-3) :157–173, 2008. [62](#)
- [201] A.-M. Saeed. Intelligent handwritten digit recognition using artificial neural network. *Int. Journal of Engineering Research and Applications*, ISSN, pages 2248–9622, 2015. [23](#)
- [202] A. E. Sallab, M. Abdou, E. Perot, and S. Yogamani. Deep reinforcement learning framework for autonomous driving. *Electronic Imaging*, 2017(19) :70–76, 2017. [21](#)
- [203] R. Schnur and D. P. Lettenmaier. A case study of statistical downscaling in Australia using weather classification by recursive partitioning. *Journal of Hydrology*, 212-213 :362–379, Dec. 1998. [5](#)
- [204] B. Scholz-Reiter, D. Weimer, and H. Thamer. Automated surface inspection of cold-formed micro-parts. *CIRP annals*, 61(1) :531–534, 2012. [19](#)
- [205] I. Seck, K. Dahmane, P. Duthon, and G. Loosli. Baselines and a datasheet for the Cerema AWP dataset. June 2018. [67](#)
- [206] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv :1312.6229*, 2013. [27](#)
- [207] P. Sermanet and Y. LeCun. Traffic sign recognition with multi-scale Convolutional Networks. In *The 2011 International Joint Conference on Neural Networks*, pages 2809–2813, San Jose, CA, USA, July 2011. IEEE. [25](#)
- [208] S. Shalev-Shwartz, S. Shammah, and A. Shashua. Safe, multi-agent, reinforcement learning for autonomous driving. *arXiv preprint arXiv :1610.03295*, 2016. [21](#)
- [209] M. Shehata, J. Cai, W. Badawy, T. Burr, M. S. Pervez, R. Johannesson, and A. Radmanesh. Video-Based Automatic Incident Detection for Smart Roads : The Outdoor Environmental Challenges Regarding False Alarms. *Intelligent Transportation Systems, IEEE Transactions on*, 9 :349–360, July 2008. [82](#)
- [210] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv :1409.1556*, 2014. [27](#), [90](#)
- [211] C. Sinclair, L. Pierce, and S. Matzner. An application of machine learning to network intrusion detection. In *Proceedings 15th Annual Computer Security Applications Conference (ACSAC'99)*, pages 371–377, Dec. 1999. [19](#)
- [212] B. T. Smith and B. Eng. Lagrange Multipliers Tutorial in the Context of Support Vector Machines. *Memorial University of Newfoundland St. John's, Newfoundland, Canada*, page 17, 2004. [16](#)

- [213] M. Sommer, S. Tomforde, and J. Hähner. Using a Neural Network for Forecasting in an Organic Traffic Control Management System. In *Presented as part of the 2013 Workshop on Embedded Self-Organizing Systems*, 2013. [21](#)
- [214] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15 :1929–1958, 2014. [43](#), [49](#)
- [215] M. Strano and B. M. Colosimo. Logistic regression analysis for experimental determination of forming limit diagrams. *International Journal of Machine Tools and Manufacture*, 46(6) :673–682, 2006. [14](#)
- [216] W. Sun, B. Zheng, and W. Qian. Computer aided lung cancer diagnosis with deep learning algorithms. In *Medical imaging 2016 : computer-aided diagnosis*, volume 9785, page 97850Z. International Society for Optics and Photonics, 2016. [18](#)
- [217] Y. Sun, D. Liang, X. Wang, and X. Tang. DeepID3 Face Recognition with Very Deep Neural Networks. *arXiv preprint arXiv :1502.00873*, Feb. 2015. [24](#)
- [218] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *Thirty-first AAAI conference on artificial intelligence*, 2017. [xi](#), [52](#), [53](#), [86](#), [91](#)
- [219] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, Sept. 2014. [27](#), [52](#)
- [220] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, Dec. 2015. [52](#)
- [221] K. Y. Tam and M. Y. Kiang. Managerial Applications of Neural Networks : The Case of Bank Failure Predictions. *Management Science*, 38(7) :926–947, 1992. [23](#)
- [222] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun. Multinet Real-time joint semantic reasoning for autonomous driving. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1013–1020. IEEE, 2018. [21](#), [61](#)
- [223] R. Teti, K. Jemielniak, G. O’Donnell, and D. Dornfeld. Advanced monitoring of machining operations. *CIRP annals*, 59(2) :717–739, 2010. [18](#)
- [224] T.-Q. Thach, C.-M. Wong, K.-P. Chan, Y.-K. Chau, Y.-N. Chung, C.-Q. Ou, L. Yang, and A. J. Hedley. Daily visibility and mortality : assessment of health benefits from improved visibility in Hong Kong. *Environmental Research*, 110(6) :617–623, Aug. 2010. [5](#)
- [225] D. Tomè, F. Monti, L. Baroffio, L. Bondi, M. Tagliasacchi, and S. Tubaro. Deep Convolutional Neural Networks for pedestrian detection. *Signal Processing : Image Communication*, 47 :482–489, Sept. 2016. [22](#)

- [226] J. Truett, J. Cornfield, and W. Kannel. A multivariate analysis of the risk of coronary heart disease in Framingham. *Journal of Clinical Epidemiology*, 20(7) :511–524, 1967. [14](#)
- [227] A. Uçar, Y. Demir, and C. Güzeliş. Object recognition and detection with deep learning for autonomous driving applications. *Simulation*, 93(9) :759–769, 2017. [21](#), [61](#)
- [228] S. Urolagin, P. K. V., and N. Reddy. Design of a Decision Tree to Classify Similar Looking Characters Using Subimages for Kannada Script. *Journal of Pattern Recognition Research*, 7(1) :42–55, 2012. [15](#)
- [229] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluation of color descriptors for object and scene recognition. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, AK, USA, June 2008. IEEE. [26](#)
- [230] B.-J. van der Zwaag. Handwritten digit recognition : A neural network demo. In *International Conference on Computational Intelligence*, pages 762–771. Springer, 2001. [23](#)
- [231] V. Vapnik. *Estimation of Dependences Based on Empirical Data : Springer Series in Statistics (Springer Series in Statistics)*. Springer-Verlag, Berlin, Heidelberg, 1982. [16](#)
- [232] A. Volokitin, R. Timofte, and L. V. Gool. Deep Features or Not : Temperature and Time Prediction in Outdoor Scenes. In *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1136–1144, June 2016. [27](#)
- [233] J. Wang, X. Liu, X. Yang, M. Lei, S. Ruan, K. Nie, Y. Miao, and J. Liu. Development and evaluation of a new digital photography visiometer system for automated visibility observation. *Atmospheric Environment*, 87 :19–25, Apr. 2014. [30](#)
- [234] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu. Deep learning for smart manufacturing : Methods and applications. *Journal of Manufacturing Systems*, 48 :144–156, 2018. [18](#), [19](#)
- [235] J. Wang, J. Song, M. Chen, and Z. Yang. Road network extraction : a neural-dynamic framework based on deep learning and a finite state machine. *International Journal of Remote Sensing*, 36(12) :3144–3169, June 2015. [21](#)
- [236] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(Feb) :207–244, 2009. [18](#)
- [237] P. Werbos and P. John. Beyond regression : new tools for prediction and analysis in the behavioral sciences. *Ph. D. dissertation, Harvard University*, Jan. 1974. [41](#)
- [238] B. Widrow, D. E. Rumelhart, and M. A. Lehr. Neural Networks : Applications in Industry, Business and Science. *Commun. ACM*, 37(3) :93–105, Mar. 1994. [22](#)
- [239] World Health Organization. *World health statistics 2018 : monitoring health for the SDGs*. 2018. [4](#)

- [240] D. Wu, D. W. Rosen, and D. Schaefer. Cloud-Based Design and Manufacturing : Status and Promise. In D. Schaefer, editor, *Cloud-Based Design and Manufacturing (CBDM) : A Service-Oriented Product Development Paradigm for the 21st Century*, pages 1–24. Springer International Publishing, Cham, 2014. [18](#)
- [241] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database : Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492. IEEE, 2010. [62](#)
- [242] L. Xie, A. Chiu, and S. Newsam. Estimating Atmospheric Visibility Using General-Purpose Cameras. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, F. Porikli, J. Peters, J. Klosowski, L. Arns, Y. K. Chun, T.-M. Rhyne, and L. Monroe, editors, *Advances in Visual Computing*, Lecture Notes in Computer Science, pages 356–367. Springer Berlin Heidelberg, 2008. [30](#)
- [243] H. Xu, P. Jia, and X. Zhang. A neural network based method for shape measurement in steel plate forming robot. In *International Symposium on Neural Networks*, pages 291–296. Springer, 2005. [29](#)
- [244] X. Yan, Y. Luo, and X. Zheng. Weather Recognition Based on Images Captured by Vision System in Vehicle. In *Advances in Neural Networks – ISNN 2009*, Lecture Notes in Computer Science, pages 390–398. Springer, Berlin, Heidelberg, May 2009. [26](#), [82](#)
- [245] J. Yang, J. Deng, S. Li, and Y. Hao. Improved traffic detection with support vector machine based on restricted Boltzmann machine. *Soft Computing*, 21(11) :3101–3112, June 2017. [17](#), [19](#)
- [246] F. Yu and V. Koltun. Multi-Scale Context Aggregation by Dilated Convolutions. *arXiv preprint arXiv :1511.07122*, Nov. 2015. [45](#)
- [247] D. Zhang, T. Sullivan, N. E. O’Connor, R. Gillespie, and F. Regan. Coastal fog detection using visual sensing. In *OCEANS 2015 - Genova*, pages 1–5, May 2015. [26](#), [62](#), [66](#)
- [248] G. Zhang, M. Y. Hu, B. Eddy Patuwo, and D. C. Indro. Artificial neural networks in bankruptcy prediction : General framework and cross-validation analysis. *European Journal of Operational Research*, 116(1) :16–32, July 1999. [23](#)
- [249] G. P. Zhang. Neural networks for classification : a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 30(4) :451–462, 2000. [22](#)
- [250] J. Zhang and M. Zulkernine. A hybrid network intrusion detection technique using random forests. In *First International Conference on Availability, Reliability and Security (ARES’06)*, pages 8 pp.–269, Apr. 2006. [14](#)
- [251] K. Zhang, C. Butler, Q. Yang, and Y. Lu. A fibre optic sensor for the measurement of surface roughness and displacement using artificial neural networks. In *Quality Measurement : The Indispensable Bridge between Theory and Reality (No Measurements? No Science! Joint Conference-1996 : IEEE Instrumentation and Measurement Technology Conference and IMEKO Tec*, volume 2, pages 917–920. IEEE, 1996. [28](#)

- [252] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu. Road crack detection using deep convolutional neural network. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3708–3712, Sept. 2016. [22](#), [61](#)
- [253] M.-L. Zhang and Z.-H. Zhou. A k-nearest neighbor based algorithm for multi-label classification. *GrC*, 5 :718–721, 2005. [18](#)
- [254] Y. Zhang, K. Sohn, R. Villegas, G. Pan, and H. Lee. Improving object detection with deep convolutional networks via bayesian optimization and structured prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 249–258, 2015. [20](#)
- [255] Z. Zhang, Q. Liu, and Y. Wang. Road Extraction by Deep Residual U-Net. *IEEE Geoscience and Remote Sensing Letters*, 15(5) :749–753, May 2018. [22](#)
- [256] Z. Zhang and H. Ma. Multi-class weather classification on single images. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 4396–4400. IEEE, 2015. [26](#), [62](#), [66](#)
- [257] Z. Zhang, H. Ma, H. Fu, and C. Zhang. Scene-free multi-class weather classification on single images. *Neurocomputing*, 207 :365–373, Sept. 2016. [26](#), [62](#), [66](#)
- [258] J. Zhao, M. Mathieu, R. Goroshin, and Y. LeCun. Stacked What-Where Auto-encoders. *arXiv preprint arXiv :1506.02351*, June 2015. [46](#)
- [259] C. Zhao-zheng, L. Jia, and C. Qi-mei. Real-Time Video Detection of Road Visibility Conditions. In *2009 WRI World Congress on Computer Science and Information Engineering*, volume 5, pages 472–476, Mar. 2009. [30](#)
- [260] Z. Zhu, L. Zhuo, P. Qu, K. Zhou, and J. Zhang. Extreme Weather Recognition Using Convolutional Neural Networks. In *2016 IEEE International Symposium on Multimedia (ISM)*, pages 621–625, Dec. 2016. [27](#)

*BIBLIOGRAPHIE*

---