Restauration de la visibilité dans des images et des vidéos acquises en conditions météorologiques défavorables

Alexandra Duminil

Restauration de la visibilité dans des images et des vidéos acquises en conditions météorologiques défavorables

Thèse de doctorat de l'Université Gustave Eiffel

École doctorale : Mathématiques et Sciences et Techniques de l'Information et de la Communication, MSTIC 532

Spécialité de doctorat : Signal, Image, Automatique

Unité de recherche : COSYS PICS-L

Thèse présentée et soutenue à l'Université Gustave Eiffel, le 30 novembre 2022.

ALEXANDRA DUMINIL

Composition du Jury

Samia BOUCHAFA Professeur, Université d'Evry	Présidente du jury
Catherine ACHARD Professeur, Sorbonne Université	Rapportrice
Samia Ainouz ZEMOUCHE Professeur, INSA Rouen	Rapportrice
Frédéric BERNARDIN Ingénieur chercheur, CEREMA	Examinateur
Pierre CHARBONNNIER Directeur de recherche, CEREMA	Examinateur
Roland BREMOND Directeur de recherche, Université Gustave Eiffel	Directeur de thèse
Jean-Philippe TAREL Chargé de recherche, Université Gustave Eiffel	Co-Encadrant

Remerciements

Je tiens tout d'abord à remercier les membres du jury de thèse, Mme Samia Ainouz Zemouche, Mme Catherine Achard, M Pierre Charbonnier, M Frédéric Bernardin et Mme Samia Bouchafa, pour avoir accepté d'évaluer mes travaux.

Je voudrais adresser toute ma reconnaissance à mon directeur de thèse Roland Brémond ainsi qu'à mon encadrant Jean-Philippe Tarel pour m'avoir aidé et conseillé tout au long de cette thèse et grâce à qui j'ai beaucoup appris.

Un grand merci à tous mes collègues du laboratoire PICS-L, à commencer par Éric Dumont, le directeur, pour ses conseils et discussions intéressants. Merci à Fabrice Vienne et Céline Villa pour leur soutien moral, et Sio Song Ieng pour ses conseils avisés. Merci à Thong Dang de m'avoir prêté du matériel et pour la visite guidée du campus de Satory, ainsi qu'à Pachak Boungnalith pour m'avoir aidé à construire le dispositif expérimental pour la création de la base de données. Je remercie également les collègues du laboratoire que j'ai pu côtoyer durant ces trois années, Stéphane Caro, Enoch Saint-Jacques, Régis Lobjois, ainsi que tous les autres, pour leur sympathie. Mention spéciale à Amélie Cadot pour sa joie et sa bonne humeur quotidienne.

Je remercie également mes proches pour m'avoir toujours soutenu, et en surtout pendant ces trois années. En particulier, un grand merci à mon père et ma sœur Laura pour leurs relectures enrichissantes et leur conseils rédactionnels.

Enfin, je tiens à remercier Kévin Morvan pour son soutien quotidien et pour m'avoir supporté dans les moments difficiles.

Résumé

Certaines conditions météorologiques telles que la pluie et le brouillard contribuent à réduire la visibilité, pouvant causer des accidents de la circulation. Lorsqu'on utilise des systèmes d'aide à la conduite (ADAS), les risques d'accidents sont moindres en raison d'un grand nombre de capteurs embarqués dont les données sont traitées par des processeurs et logiciels spécifiques. Cependant, les technologies embarquées aujourd'hui ne permettent pas encore de disposer de systèmes intégrés et bien accordés ensemble pour faire face de manière efficace à des conditions météorologiques dégradées. Il devient alors indispensable d'intégrer des capteurs et des moyens de traitements afin d'atténuer les artefacts provoqués par ce genre de conditions météorologiques. L'utilisation d'algorithmes de restauration d'images et de vidéos permettrait d'atténuer leurs effets sur l'image restituée par le système pour l'aide à la décision.

L'objectif de cette thèse est de concevoir des algorithmes s'exécutant au plus proche du temps réel pour améliorer la visibilité de la scène en restaurant les images acquises en conditions météorologiques dégradées qui appartiennent à la catégorie des particules fines (brume, brouillard et poussière). Deux approches vont être confrontées : une approche fondée sur des techniques traditionnelles impliquant un modèle physique du brouillard, et une approche fondée sur des techniques de *deep learning*.

Un ensemble de techniques et de méthodes ont été proposés pour répondre à ces problématiques dont un algorithme d'atténuation du brouillard dans des images basé sur un modèle physique ainsi qu'un algorithme d'atténuation du brouillard dans des vidéos basé sur une technique de *deep learning*. Le peu de bases de données existantes de vidéos contenant du brouillard pour évaluer ces algorithmes nous ont confortés dans la décision d'en créer une.

Des évaluations ont été menées pour chacune des méthodes afin d'évaluer leur efficacité.

Abstract

Some weather conditions such as rain and fog contribute to reduced visibility, which can cause traffic accidents. Using advanced driver assistance systems (ADAS), the risk of accidents is reduced due to a large number of on-board sensors equipped with specific processors and software. However, these systems are not always efficient enough to deal effectively with degraded atmospheric conditions. Then, it becomes necessary to integrate sensors and image processing in order to attenuate the visual effects caused by degraded conditions. The use of image and video restoration algorithms makes it possible to attenuate these effects.

The aim of this thesis is to design algorithms running as close as possible to real time to improve the visibility of the scene by restoring the images acquired in degraded weather conditions. We worked on fine particles including mist, fog and dust. Two approaches are compared : a physical-based approach involving a physical model of fog, and a learning-based approach.

Several algorithms have been proposed to address these issues including a single image fog removal algorithm based on physical prior, and a video fog removal algorithm with a deep learning technique. Moreover, the lack of video datasets containing fog encouraged us to create one.

Evaluations were conducted for each of the algorithms in order to assess their effectiveness.

Table des matières

Table des figures

Liste des tableaux

Chapitre 1

Introduction

Dans ce chapitre introductif, le périmètre de la thèse incluant le contexte et les objectifs va être défini.

1.1 Contexte

Le bilan d'accidentalité établie par l'**Observatoire National Interministériel de la Sécurité Routière** (ONISR) en 2020 a montré qu'un accident sur cinq a lieu dans des conditions météorologiques dégradées.

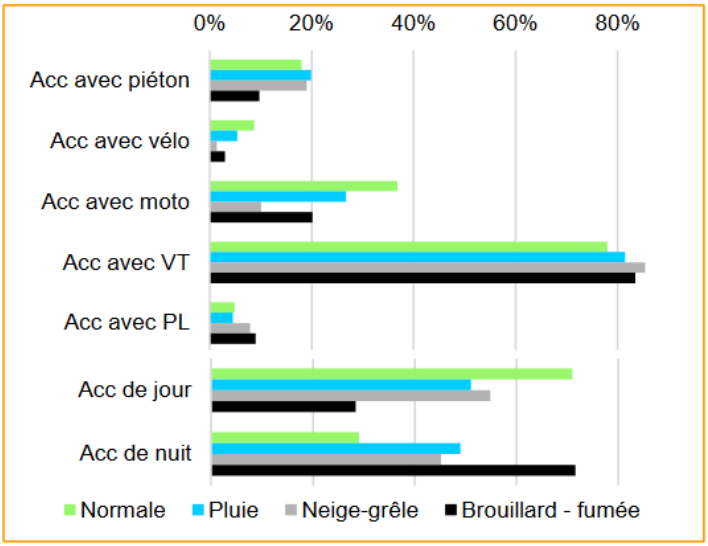


FIGURE 1.1 – Types d'accident selon certaines conditions météorologiques (2015-2019). VT correspond à voiture et PL à poids lourd.

La figure 1.1 montre que les accidents en conditions météorologiques dégradés sont moins fréquents que ceux qui se produisent sous des conditions météorologiques normales, qui constituent 80 % des accidents. Il faut néanmoins considérer que le brouillard est un phénomène plutôt rare en France. D'après les données observées par Météo France, le nombre moyen de jour de brouillard par an dans les villes les plus exposées n'excéderait pas 85. Dans certaines régions,

le brouillard se manifeste proche de différentes zones (forêt, plaines, côtes,...) où l'air est saturé en vapeur d'eau sur un sol froid.

Les accidents en conditions météorologiques dégradées ne constituent pas la majorité des accidents mais sont en général plus graves, dû à une réduction de la distance de visibilité. Une étude menée par Cavallo et al. (2000) sur les effets visuels du brouillard a montré que les conducteurs avaient tendance à surestimer les distances dans le brouillard, contribuant à la réduction des intervalles de sécurité entre les véhicules et donc à l'augmentation du risque d'accident.

L'essor des systèmes d'assistance à la conduite de véhicules (en anglais, *Advanced Driver Assistance Systems*, ADAS) a permis une avancée significative dans la prévention des accidents de la circulation. Dotés de systèmes multi-capteurs, les ADAS assistent le conducteur dans sa perception de l'environnement. Des dispositifs d'aide à la conduite de véhicule parmi lesquels : l'aide au freinage d'urgence, l'alerte de franchissement de ligne, le régulateur de vitesse et bien d'autres (voir figure 1.3) existent depuis plusieurs décennies et les constructeurs automobile ne cessent d'en améliorer l'efficacité. Les progrès de la technologie ont permis l'essor de nouveaux capteurs permettant de percevoir l'environnement. De plus en plus équipés de capteurs caméras, LIDAR, SONAR, RADAR et GPS, il devient possible de gérer des tâches complexes comme la détection de panneaux de la circulation ou de piétons par la construction d'un modèle tridimensionnel de l'environnement. La figure 1.2 présente une capture du système de caméras Mobileye, une entreprise qui développe des technologies de conduite autonome et des ADAS.

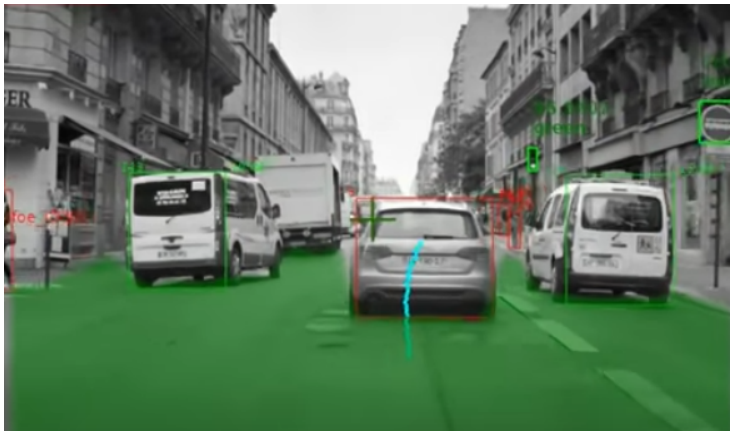


FIGURE 1.2 – Capture d'écran d'un système de caméras Mobileye. Cette capture provient du site `https://dewesoft.com/fr/daq/types-capteurs-adas`.

Le système de la figure 1.2 est par exemple capable de détecter les véhicules, les panneaux et les piétons. La plupart de ces informations provenant des capteurs n'est pas directement visible par le conducteur. En général, ils sont alertés par des voyants lumineux et des signaux sonores.

Cependant, les technologies embarquées aujourd'hui ne permettent pas encore de disposer de systèmes intégrés et bien accordés pour faire face de manière efficace à des conditions météorologiques dégradées. Bien qu'il existe des capteurs de détection de pluie, permettant d'activer les essuies-glasses de manière automatique, ces capteurs ne permettent pas d'améliorer la visibilité pour les conducteurs. Ainsi, par forte pluie ou brouillard dense, les systèmes de vision peuvent soit ne plus envoyer de données, soit prendre une décision inadaptée au contexte, comme indiqué dans la figure 1.4. Dans le premier cas, en général, le conducteur est informé et celui-ci peut reprendre le contrôle du véhicule. Dans l'autre cas, si une décision est inadaptée (par exemple si le système fournit des données erronées, ou un manque de fiabilité entraînant

non détection d'un piéton ou d'un panneau), les conséquences peuvent être dramatiques. L'intégration des systèmes permettant aux conducteurs d'avoir un retour visuel, comme à partir d'un écran ou un pare-brise affichant en réalité augmentée (head up display), issues de traitements temps réel de données capteurs, doit être envisagé (des projets commencent à voir le jour). L'intégration de ces systèmes dans les ADAS permet au conducteur d'avoir une meilleure visibilité et donc un meilleur contrôle du véhicule (voir figure 1.5).

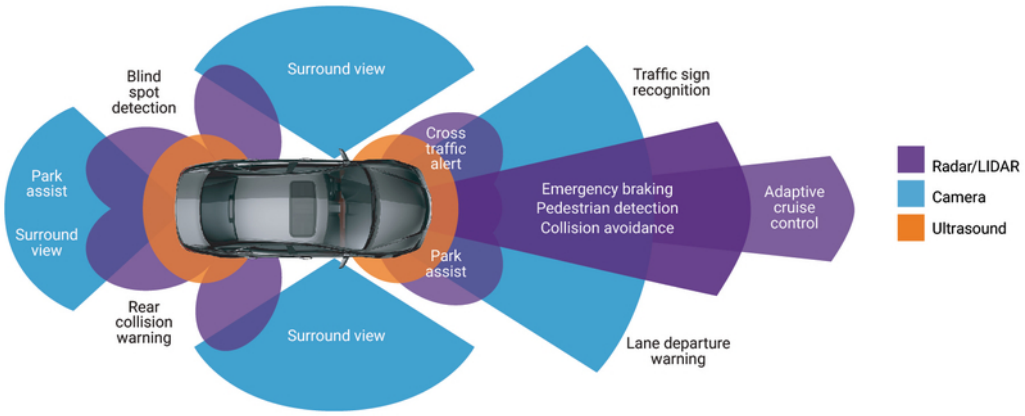


FIGURE 1.3 – Champs de perception de différents capteurs. Image provenant du site https://www.synopsys.com/automotive/what-is-adas.html.

Dans le processus d'amélioration continue d'aide à la conduite dans le but d'améliorer la sécurité, il devient nécessaire d'intégrer des capteurs et des moyens de traitements pour détecter les conditions météorologiques dégradées et de restaurer les flux vidéo en atténuant les effets visuels des conditions dégradées. L'utilisation d'algorithmes de restauration d'images permet d'atténuer le brouillard sur l'image restituée par le capteur. Ces algorithmes peuvent également être appliqué au traitement des autres dégradations visuelles liées aux conditions météorologiques. L'emploi de ces algorithmes de restauration d'images vise deux applications : améliorer la perception des capteurs, et améliorer la visibilité pour le conducteur. L'utilisation d'algorithmes de restauration de la visibilité doit contribuer à réduire les risques d'accident de la circulation.

Dans le cadre de cette thèse, nous partons de l'hypothèse que les images et les flux vidéos traités proviennent des capteurs caméras RGB des véhicules.

1.2 Objectifs

L'un des objectifs de la thèse est la réalisation d'algorithmes se rapprochant du temps réel, permettant la restauration de la visibilité des images et vidéos dégradées quelles que soient les conditions météorologiques, de nuit comme de jour. Parmi les différents types de conditions

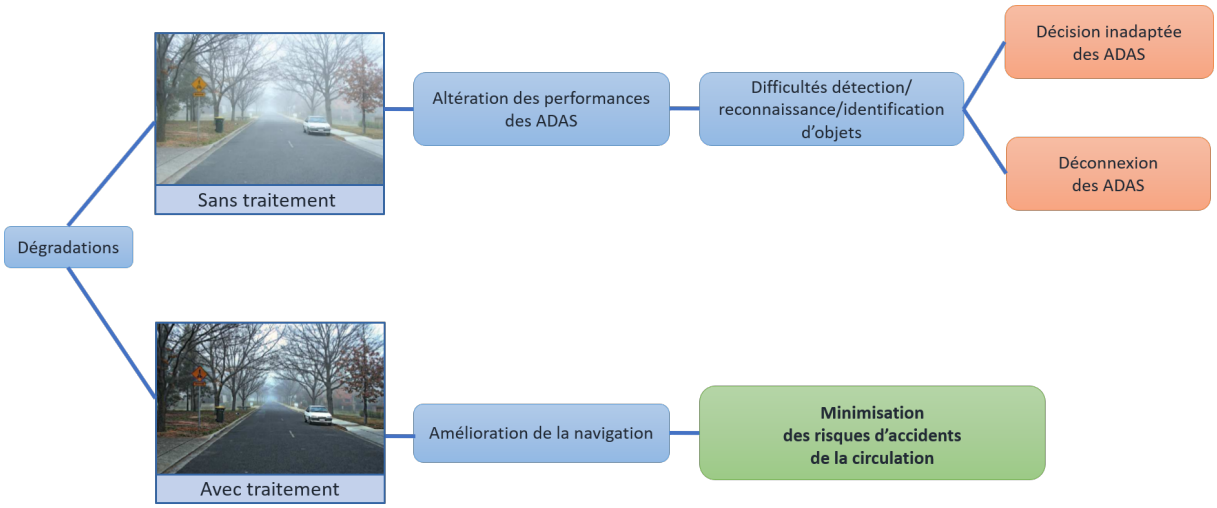


FIGURE 1.4 – Impact de l'utilisation d'un algorithme de restauration pour les ADAS.

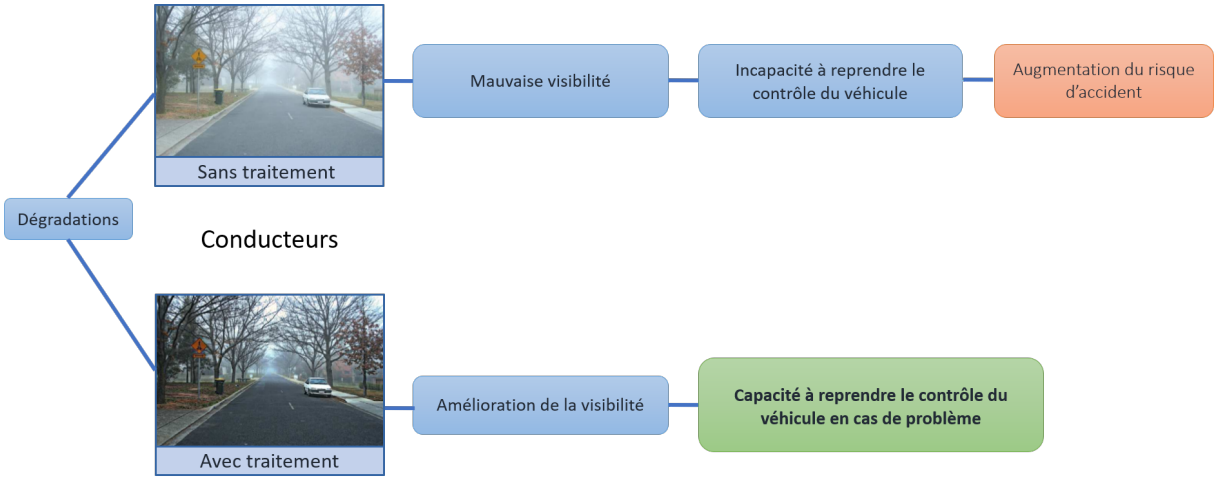


FIGURE 1.5 – Impact de l'utilisation d'un algorithme de restauration pour les conducteurs.

dégradées, nous pouvons distinguer les particules par leur rayon en micromètre. Le tableau 1.1 répertorie quelques conditions météorologiques comme le brouillard et la pluie.

TABLE 1.1 – Type de particules en fonction des conditions météorologiques (adapté de l'article de Nayar and Narasimhan (1999)).

Conditions	**Type de particules**	**Rayon en μm**	**Concentration en cm^{-3}**
Air	Molécule	10^{-4}	10^{19}
Brume	Aérosol	$10^{-2} - 1$	$10^{3} - 10$
Brouillard	Goutte d'eau	$1 - 10$	$100 - 10$
Poussière	Aérosol	$1 - 10^{2}$	-
Nuage	Goutte d'eau	$1 - 10$	$300 - 10$
Pluie	Goutte d'eau	$10^{2} - 10^{4}$	$10^{-2} - 10^{-5}$

Les types de conditions météorologiques peuvent être divisés en deux catégories. La catégorie des particules fines où figurent la brume, le brouillard et la poussière, et la catégorie des grosses

particules comme les particules de pluie. Durant cette étude, nous nous concentrons essentiellement sur les conditions météorologiques de la catégorie des particules fines, incluant, brume, brouillard et poussière.

L'algorithme doit être le plus générique possible et être capable de restaurer la visibilité dans des images avec différentes densité de brouillard, homogène ou non. Des évaluations seront conduites avec les algorithmes réalisés pour analyser leurs adéquations à d'autres type de conditions et statuer sur le niveau de généralité de ces algorithmes. Ces évaluations permettront ainsi de conclure si ces algorithmes sont en mesure de restaurer les images dégradées par la pluie ou s'il est nécessaire de les adapter.

Deux types d'approches vont être confrontés, une approche fondée sur des techniques de *deep learning* et une autre fondée sur des techniques plus traditionnelles impliquant le modèle physique du brouillard. L'objectif est de choisir la méthode la plus adaptée en réalisant un état de l'art des deux approches et des bases de données existantes en image par image et en vidéo.

Chapitre 2

État de l'Art

Restauration d'images et de vidéos dans le brouillard

La dégradation de la visibilité due à de mauvaises conditions météorologiques a un impact négatif sur la compréhension de l'environnement. L'objectif poursuivi est donc d'atténuer un maximum leurs effets. Dans ce chapitre, nous allons détailler différentes méthodes de restauration d'images et de vidéos impactées par le brouillard.

2.1 Les images dégradées par le brouillard

2.1.1 La formation du brouillard

Le brouillard est constitué d'un ensemble de particules d'eau suspendues dans l'air. Leur formation nécessite une grande quantité d'humidité dans l'air, ainsi qu'une température basse au niveau du sol. Un processus de refroidissement provoque la condensation de la vapeur d'eau et permet la formation de fines gouttelettes en suspension, qui constituent le brouillard (voir figure 2.1).

Il existe plusieurs types de brouillard, dont la formation et la densité varient en fonction des conditions de température, d'humidité et des zones géographiques :

— **Le brouillard de rayonnement** : il se forme la nuit par refroidissement du sol.

— **Le brouillard d'advection** : il se forme par contact d'une masse d'air chaude et humide avec une surface froide. Les gouttelettes d'eau en suspension dans l'atmosphère vont alors se condenser.

— **Le brouillard d'évaporation** : il se forme sur les surfaces aquatiques, lorsqu'une masse d'air froide provenant de la surface terrestre entre en contact avec l'air chaud et humide de la surface d'eau.

Pour rappel, les ADAS ont pour objectif d'améliorer la sécurité en assistant le conducteur dans sa perception de l'environnement (comme les détecteurs de dépassement, de risque de gel, de piétons, etc). Grâce aux capteurs, le système va pouvoir prévenir les risques et anticiper sur la réaction du conducteur. Cependant, de mauvaises conditions météorologiques peuvent altérer les performances des ADAS et engendrer des difficultés de détection, de reconnaissance et d'identification des objets. Par exemple, un brouillard dense qui diminue la visibilité à moins

FIGURE 2.1 – Images avec du brouillard.

de 200 mètres, conduit à de difficiles conditions de circulation. Dans ce cas, une décision inadaptée du système pourrait augmenter le risque d'accident. L'utilisation de pré-traitements de restauration de la visibilité des images avec de mauvaises conditions pourrait alors minimiser ces risques.

2.1.2 Le modèle physique du brouillard

La diffusion de la lumière par les particules atmosphériques peut être décrite par un modèle simple, connu sous le nom de la loi de Koschmieder (1924), également détaillé par Nayar and Narasimhan (1999) et Halmaoui (2012). Le modèle fait intervenir deux aspects : la transmission directe et le voile atmosphérique.

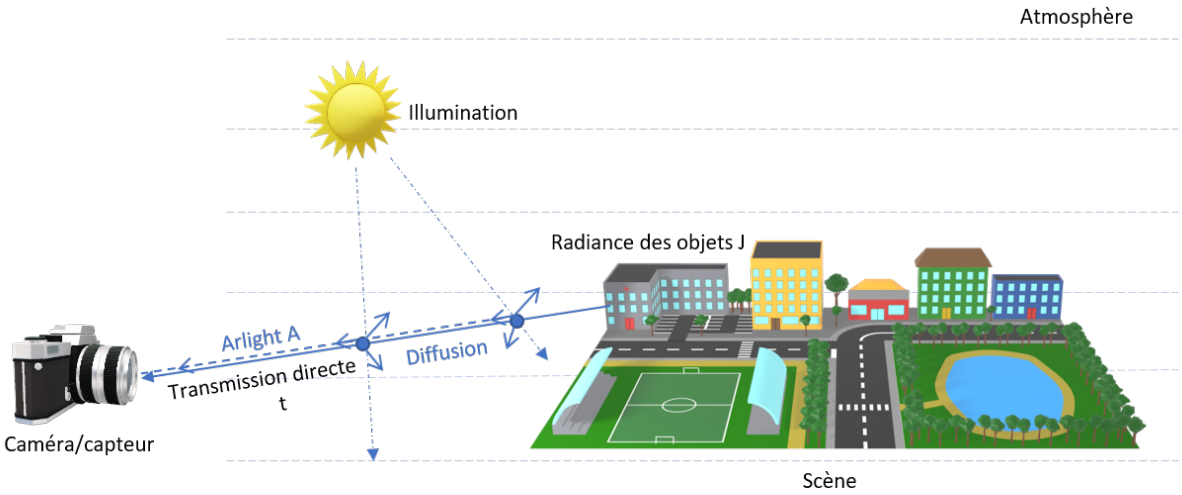


FIGURE 2.2 – Description des phénomènes de transmission et du voile atmosphérique.

2.1.2.1 La transmission directe

Le modèle de transmission directe décrit la manière dont la lumière est atténuée lorsqu'elle traverse l'atmosphère. Du fait de la diffusion atmosphérique, une partie du flux lumineux diffusée est soustraite du faisceau incident. Le flux non diffusé, appelé transmission directe, est dirigé vers l'avant (voir la flèche bleue continue entre la caméra et la scène sur la figure 2.2).

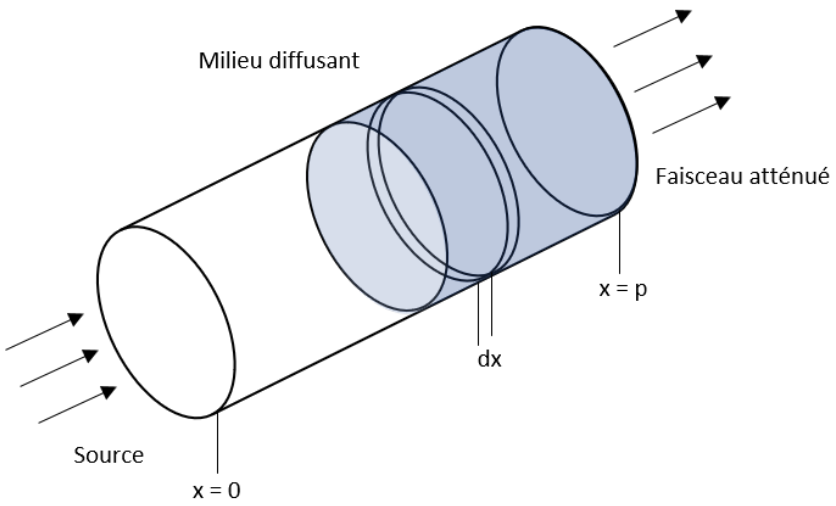


FIGURE 2.3 – Atténuation directe d'un faisceau lumineux par des particules en suspension.

Considérons un faisceau de lumière incident au milieu atmosphérique, illustré à la figure 2.3, traversant une tranche entre x et $x+dx$ du faisceau cylindrique. La variation du flux lumineux E en x s'écrit :

$$\frac{dE(x)}{E(x)} = -\beta dx \tag{2.1}$$

β correspond au coefficient d'extinction qui est supposé constant dans l'ensemble du cylindre ; le brouillard est supposé homogène. En intégrant entre $x = 0$ et $x = p$, l'équation obtenue est la suivante :

$$E(p) = E_0 e^{-\beta p} \tag{2.2}$$

où E_0 est l'irradiance à la source. Cette équation est la loi d'atténuation exponentielle de Bouguer (Bouguer, 1729). Elle montre que la luminance des objets diminue exponentiellement lorsque la densité du brouillard et la distance des objets dans la scène augmentent.

2.1.2.2 Le voile atmosphérique

La diffusion atmosphérique produit un voile atmosphérique, en anglais *airlight*. Il correspond à la façon dont une portion de l'atmosphère réfléchit la lumière environnante dans toutes les directions. Les sources sont diverses : la lumière directe du soleil, celle diffusée par le ciel et la lumière réfléchie par le sol.

Comme visible sur la figure 2.3, chaque tranche entre x et $x+dx$ du faisceau cylindrique diffuse une partie de la lumière environnante dans la direction de l'observateur. A nouveau, on suppose le brouillard homogène et l'éclairage uniforme de valeur E_s. Chaque tranche diffuse $\beta E_s dx$ dans la direction de l'observateur. Ce flux est atténué comme dans l'équation 2.2. La variation du voile atmosphérique due à la tranche entre x et $x+dx$ s'écrit donc :

$$dE(x) = e^{-\beta x} \beta E_s dx \tag{2.3}$$

En intégrant l'équation 2.3 entre $x = 0$ et $x = p$, on obtient :

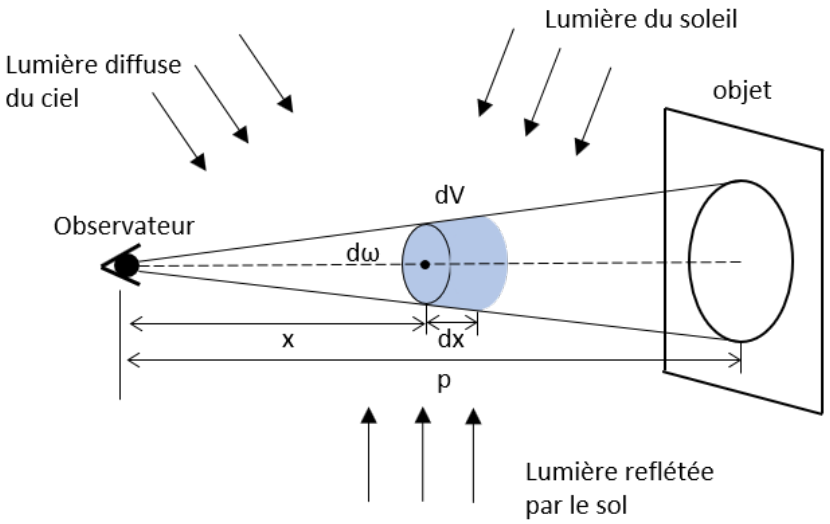


FIGURE 2.4 – Illustration d'un cône d'atmosphère entre un observateur et un objet.

$$E(p) = \int_0^p e^{-\beta x} \beta E_s dx \tag{2.4}$$

Par intégration, on obtient la solution suivante :

$$E(p) = E_s(1 - e^{-\beta p}) \tag{2.5}$$

La formule du voile atmosphérique peut également être obtenue en considérant un cône d'atmosphère, comme illustré à la figure 2.4, au lieu d'un cylindre. Avec un cône ou un cylindre, des formules identiques sont obtenues pour le voile atmosphérique comme pour la loi de Bouguer.

2.1.2.3 La loi de Koschmieder

La somme des équations de la transmission $E_T(d)$ (équation 2.2) et du voile atmosphérique $E_A(d)$ (équation 2.5) correspond au modèle de Koschmieder (1924) en termes de luminance :

$$E(d) = E_A(d) + E_T(d) \tag{2.6}$$

La loi de Koschmieder est un modèle mathématique simple utilisé pour décrire les effets visuels d'un milieu diffusant, comme le brouillard. Le brouillard et l'éclairage sont supposés homogènes le long d'un faisceau passant par x. L'équation 2.6 peut également s'écrire de la manière suivante, en termes d'intensité lorsque la réponse de la caméra est supposée linéaire :

$$I(x) = J(x)t(x) + A(1 - t(x)) \tag{2.7}$$

où $I(x)$ est l'image avec brouillard, $J(x)$ l'image sans brouillard, A l'intensité du ciel et $x = (u, v)$ désigne les coordonnées des pixels dans chaque image. La transmittance $t(x)$ décrit le pourcentage de lumière qui n'est pas diffusé :

$$t(x) = e^{-\beta d(x)} \tag{2.8}$$

où β est le coefficient d'extinction lié à la densité du brouillard de jour et $d(x)$ la distance entre la caméra et les objets de la scène. Le voile atmosphérique correspond au second terme de l'équation (2.7).

2.1.2.4 La distance de visibilité

Comme il a été mentionné dans l'introduction de la thèse, le brouillard et autres conditions atmosphériques dégradés, sont responsables de la réduction de la visibilité sur les routes. Mais à quoi correspond exactement la distance de visibilité ? Il existe plusieurs définitions dans la littérature. D'après le vocabulaire international de la météorologie (WMO (1992)), et de l'éclairage (CIE (1987)), la visibilité est définie comme « la plus grande distance à laquelle un objet noir de dimensions convenables peut être reconnu de jour sur le ciel à l'horizon ». Mais en pratique, la visibilité correspond à la plus grande distance à laquelle il est possible d'identifier à l'oeil nu un objet étendu sur le ciel à l'horizon de jour, et une source lumineuse diffuse de nuit. La CIE (1987) distingue trois types de distance de visibilité :

1. La portée visuelle : elle correspond à la distance de visibilité de jour
2. La portée lumineuse : elle correspond à la distance de visibilité de nuit
3. La portée optique météorologique : elle regroupe les définitions précédentes et est utilisée dans la terminologie de la météorologie routière, AFNOR (1998).

La portée visuelle est caractérisée par l'atténuation atmosphérique du contraste C, entre l'objet de luminance L et l'arrière plan sur lequel il se détache, de luminance L_f. Duntley (Middleton (1952)) établit une loi d'atténuation du contraste C, définie par :

$$C = \frac{L - L_f}{L_f} = C_0 e^{-kd} \tag{2.9}$$

où C est le contraste d'un objet observé à une distance donnée d, C_0 le contraste observé à courte distance et k le coefficient d'extinction atmosphérique, supposé uniforme.

La portée optique météorologique V_{met} correspond à la longueur d'un trajet dans l'atmosphère sur lequel le flux lumineux d'un faisceau quasi parallèle de rayonnement, émanant d'une source de lumière de température de couleur 2700 K, est réduit de 95 % (CIE (1987)).
La valeur de l'atténuation a été choisie de telle sorte que ce terme conduise à une mesure approximative de la visibilité V_{met}, pour un seuil de contraste C_s fixé arbitrairement à 5%. V_{met} est relié au coefficient atmosphérique par la formule suivante :

$$e^{-kV_{met}} = C_s \tag{2.10}$$

Pour un seuil de contraste à 5 % la formule de V_{met} est défini par :

$$V_{met} = -\frac{1}{k} ln(0.05) \simeq \frac{3}{k} \tag{2.11}$$

Dans la suite de ce chapitre, des méthodes de restauration d'images et de vidéos contenant du brouillard vont être présentés.

2.2 Méthodes de restauration d'images contenant du brouillard

Les méthodes classiques de restauration d'images s'appuient sur une analyse descriptive du problème en définissant un modèle physique ou mathématique compréhensible (voir le chapitre 2.1). L'analyse consiste à collecter différentes données sur un phénomène en émettant des hypothèses (appelées *a priori*) et en les validant à l'aide d'un processus d'évaluation avec des données réelles. On différencie les *a priori* non physiques des *a priori* physiques, car ces derniers représentent un modèle mathématique de dégradation de l'image qui permet d'atténuer

certains effets dans l'image comme le flou, le bruit, ainsi que d'autres types de dégradations. Les *a priori* non physiques sont utilisés, par exemple, pour l'amélioration d'images (égalisation d'histogrammes par exemple). Il s'agit de contraintes ajoutées, souvent subjectivement, qui permettent de traiter les défauts sans ambiguïté, et qui sont souvent inspirées du système visuel humain. Les méthodes d'apprentissage par *deep learning*, quant à elles, n'utilisent pas d'analyse descriptive. Il s'agit de systèmes alimentés par un très grand nombre de données d'apprentissage. Ils sont obtenus par minimisation de la fonction de coût entre les données observées et les données prédites.

De nombreux travaux ont prouvé l'efficacité des algorithmes basés sur les *a priori*, comme ceux basés sur l'apprentissage pour la restauration de la visibilité d'une image contenant du brouillard. Très récemment, des chercheurs ont commencé à explorer la piste de la restauration de vidéos en exploitant les informations supplémentaires disponibles entre les trames adjacentes qui n'existent pas dans le cas d'images seules. Il semblerait judicieux, dans un premier temps, d'étendre les techniques existantes de traitement d'images aux vidéos en appliquant les traitements trame par trame. Cependant, comme ces algorithmes de restauration d'images contenant du brouillard traitent uniquement les images statiques, les appliquer à des vidéos introduirait des artefacts et des effets de scintillements en raison d'un manque de cohérence temporelle. En effet, il peut exister des variations de luminosité et de couleurs entre trames adjacentes qui seraient amplifiées après traitement, rendant les résultats de la restauration des trames voisines différentes. Le challenge consiste donc à obtenir un algorithme de restauration de vidéos capable d'assurer une cohérence spatio-temporelle.

Dans les sous-sections suivantes, les principales méthodes de restauration d'images et de vidéos vont être présentées.

2.2.1 Les méthodes basées sur des modèles physiques

Les algorithmes d'atténuation du brouillard dans les images peuvent être divisés en deux catégories. Les algorithmes d'amélioration d'images constituent la première catégorie. Ils utilisent des techniques *ad hoc* pour améliorer le contraste de l'image, telles que l'égalisation d'histogramme, la transformation en ondelettes ou le retinex. Les algorithmes de restauration d'images, qui constituent la seconde catégorie, reposent sur un modèle physique, en général la loi de Koschmieder.

Le problème est vu comme un problème inverse mal posé qui doit être résolu en ajoutant de l'information (un *a priori*) afin de devenir un problème bien posé et bien conditionné.

Nous verrons dans la suite de cette sous-section, différentes approches basées sur des a priori physiques.

2.2.1.1 Le *Dark Channel*

He et al. (2009) ont introduit la méthode du *Dark Channel Prior* (*DCP*) dédiée aux images en couleur. Le principe est de supposer qu'une image extérieure sans brouillard contient des pixels de très faible intensité sur au moins l'un des trois canaux de couleur, dans n'importe quelle zone de l'image. L'équation du *Dark Channel* est définie de la façon suivante :

$$I_{dark}(x) = \min_{c \in r,g,b} [\min_{y \in \Omega(x)} (I^c(y)] \tag{2.12}$$

où I^c correspond au canal de couleur c de l'image I, et $\Omega(x)$ est une fenêtre locale centrée en x. Les auteurs s'appuient sur l'observation suivante : exceptée la zone du ciel, $I_{dark}(x)$ est proche de zéro si l'image I est l'image sans brouillard (voir figure 2.5).

Dans l'équation 2.12, le *DCP* suppose que le brouillard est homogène sur chaque fenêtre locale $\Omega(x)$. La méthode du *Dark Channel* permet d'obtenir une approximation de la carte de transmittance d'équation :

$$t(x) = 1 - \omega \min_c (\min_{y \in \Omega(x)} (\frac{I^c(y)}{A^c})) \tag{2.13}$$

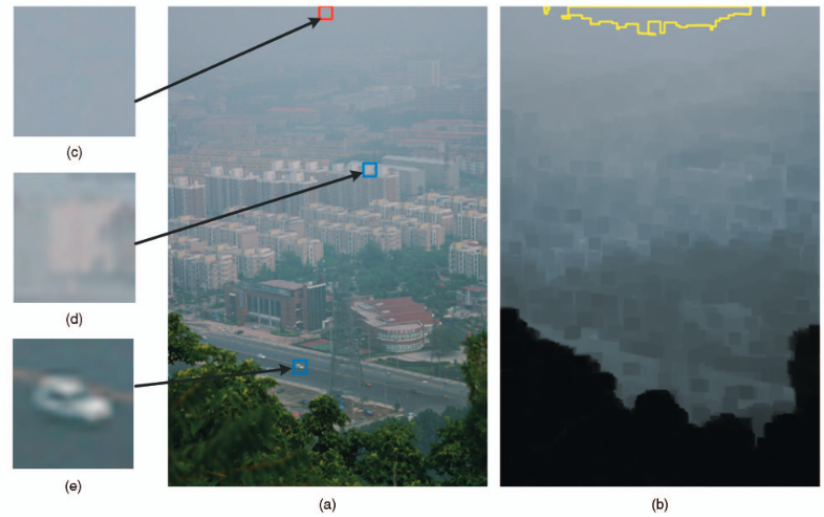


FIGURE 2.5 – Estimation de l'intensité du ciel (paramètre A) : a) l'image d'entrée, b) le *dark channel*, c) le patch à partir duquel est obtenu l'intensité A, d) et e) les patches contenant des pixels dont l'intensité est plus élevée que la valeur de A. L'image provient de l'article de He et al. (2011).

où $\Omega(x)$ est une fenêtre locale centrée sur x, c est le canal de couleur, A l'intensité du ciel et I l'intensité de l'image. D'après l'équation 2.7, le voile atmosphérique est égal à $A(1-t(x))$. $I_{dark}(x)$ peut être interprété comme étant le voile atmosphérique. Un paramètre largement utilisé, appelé ω, a été introduit dans le calcul de la carte de transmittance et du voile atmosphérique. Ce paramètre, généralement fixé à $\omega = 0.95$, est constant sur toute l'image et a été introduit pour la première fois pour atténuer le phénomène de sur-restauration d'images. Selon He et al. (2009), ω permet de garder une petite quantité de voile devant des objets distants, afin d'obtenir des résultats plus naturels et réalistes. Ce paramètre peut être interprété comme étant un *a priori* et il semble nécessaire qu'il soit analysé.

2.2.1.2 *A priori* sur le voile atmosphérique

Tan et al. (2007) sont parmi les premiers à avoir proposé un article sur la restauration de la visibilité dans le brouillard, et ils ont introduit deux *a priori* :

1. Les images sans brouillard sont plus contrastées que les images brumeuses, ce qui conduit à une diminution du contraste.

2. Le voile atmosphérique est constant sur chaque région locale comme dans la méthode *DCP*.

Cette méthode peut s'appliquer à la fois sur les images en couleur et sur celles en niveaux de gris.

Tarel (2009) a proposé deux *a priori* : le brouillard est supposé blanc et localement lisse. Comme indiqué dans Tarel et al. (2012), le premier *a priori* conduit à utiliser le canal d'intensité minimum parmi les trois canaux de couleur (équation 2.12). Le second *a priori* consiste à utiliser un filtre local, de préférence apte à conserver les bords et les coins de l'image. Cet algorithme s'exécute rapidement et permet la restauration d'images en couleur et en niveaux de gris. Dans l'approche de Tarel et al. (2012), une hypothèse de monde plan a été introduite afin d'éviter une sur-restauration dans la partie inférieure des images restaurées, dans le cadre d'images routières. Les résultats sont satisfaisants sur des objets proches de la caméra, mais la hauteur de la ligne d'horizon doit être approximativement connue. Pour faire face à cette difficulté de sur-restauration éventuelle, une fonction de modulation du voile atmosphérique a été introduite par Negru et al. (2015) pour traiter les scènes de trafic routier. Cette fonction de modulation dépend de la position du pixel et ces paramètres doivent donc être réglés en fonction de la scène observée.

2.2.1.3 La cohérence temporelle dans les vidéos

Les algorithmes de restauration d'images produisent d'excellents résultats, mais leur applicabilité reste limitée. Pour rappel, appliquer un algorithme de restauration d'image sur une vidéo entraîne l'apparition d'artefacts due à un manque d'information de la dimension temporelle. Pour remédier à ce problème, Zhang et al. (2011) proposent une méthode de restauration de brouillard par l'estimation trame par trame de la carte de transmission à l'aide de la méthode du *DCP* (voir 2.2.1.1). Afin de préserver la cohérence temporelle, une estimation du flot optique est réalisée dans le but d'estimer le mouvement inter-trame de chaque pixel des images de la séquence vidéo. L'estimation du flot optique permet d'obtenir des cartes d'erreur qui vont servir au modèle MRF (champs aléatoire de Markov) pour procéder à un lissage de la carte de transmission brute le long de la dimension temporelle. L'hypothèse la plus souvent prise dans le calcul du flux optique est la constance de la luminosité dans les images. Dans une paire d'images avec et sans brouillard de la même scène et en conditions de luminosité identique, les luminosités des pixels correspondants ne sont pas identiques en raison du changement de profondeur. Cependant, le changement reste minime entre deux images adjacentes dans une vidéo. Ce procédé permet d'obtenir des cartes de précision utilisées ensuite pour construire le modèle MRF avec pour objectif d'améliorer la cohérence spatiale et temporelle des cartes de transmission estimées.
Cependant, leur algorithme nécessite un paramétrage complexe et n'est pas applicable à la restauration de vidéo en temps réel. La même année, Kim et al. (2012) ont également exploité les informations des trames voisines d'une vidéo pour l'estimation de la carte de transmission. En supposant que les valeurs associées à la transmission d'un objet dans la scène sont identiques entre les trames, ils ont développé une fonction de coût de cohérence temporelle. Ajouté à une fonction de coût globale, l'objectif est de minimiser cette fonction de coût afin d'estimer une carte de transmission optimale. Les résultats expérimentaux démontrent que l'algorithme s'exécute rapidement et supprime efficacement les artefacts. Cependant, le traitement est réalisé au niveau du pixel, ce qui ne convient pas aux traitements de vidéos. De plus, les performances de l'algorithme diminuent lorsque le brouillard est dense.

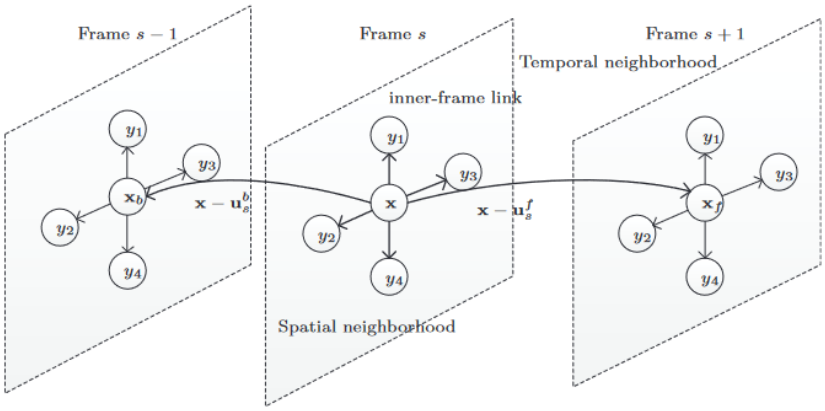


FIGURE 2.6 – Schéma du modèle MRF utilisé pour améliorer la cohérence spatio-temporelle de la séquence de cartes de transmission d'après Zhang et al. (2011).

2.2.2 Les méthodes basées sur les données

2.2.2.1 Généralités sur le *deep learning*

Le *deep learning*, ou apprentissage profond, est un sous-domaine du *machine learning* et de l'intelligence artificielle. Il permet de modéliser et prédire des données de manière automatique, en s'appuyant sur des réseaux de neurones artificiels multi-couches. En alimentant ces réseaux de neurones d'une grande quantité de données, le système va pouvoir analyser les données (apprentissage des données) et réaliser des prédictions. Ces prédictions sont ensuite comparées à un ensemble de données indépendant appelé label ou vérité terrain afin d'en vérifier l'exactitude. Les applications du *deep learning* sont multiples : le traitement d'images (par exemple la classification ou reconnaissance d'objets), le traitement des sons, le traitement du langage naturel, et bien d'autres.

Prenons un exemple de classification d'images afin de mieux appréhender les mécanismes de l'apprentissage automatique. La première étape de l'algorithme consiste à détecter et décrire les caractéristiques visuelles des images (en anglais, *features*) du jeu de données, puis à entraîner un classifieur (réseaux de neurones qui classifient les images par classes) sur ces *features*. Cette étape peut être réalisée, par exemple, par un réseau de neurones convolutifs (CNN) qui permet d'extraire et de hiérarchiser automatiquement les *features* à différents niveaux de complexité. Ensuite, le réseau prédit des labels sur un lot d'exemples et les compare à des labels de référence. Les écarts entre les prédictions et les références font intervenir le processus de rétro-propagation du gradient, qui consiste à mettre à jour les poids des neurones de la dernière couche en direction de la première couche (voir figure 2.7). De cette manière la rétro-propagation permet de corriger les erreurs de prédictions. Une fonction de coût est appliquée pour quantifier l'erreur entre les labels prédits et les labels de référence. L'apprentissage se poursuit en répétant ces étapes et s'arrête en fonction des critères d'arrêts (avec un nombre d'étapes ou suivant des critères de stabilité, par exemple).
La figure 2.7 illustre la méthode de mise à jour des poids par rétro-propagation. La propagation fournit une valeur de la fonction de coût. La rétro-propagation permet ensuite d'obtenir les gradients nécessaires à la mise à jour des poids du réseau.

Il faut distinguer les techniques d'apprentissage supervisées des techniques d'apprentissage non supervisées. Dans le premier cas, le réseau est guidé par des données labellisées. Le but est alors d'obtenir des valeurs de sortie aussi proches que possible de la référence. La tech-

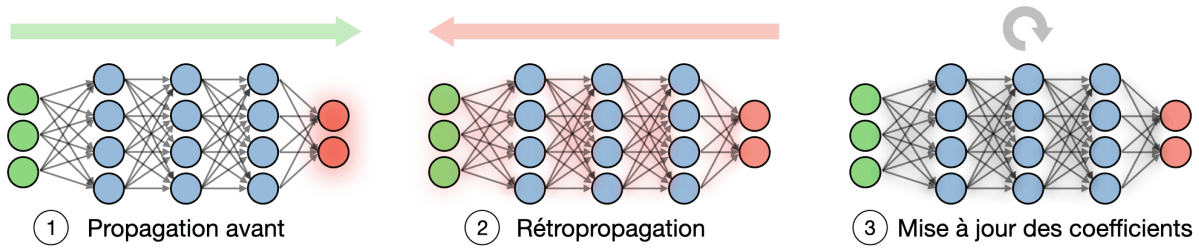


FIGURE 2.7 – Processus de mise à jour des poids d'un réseau (source : stanford.edu).

nique d'apprentissage non supervisées, quant à elle, consiste à apprendre sur un jeu de données, sans données labellisées. Au terme de l'étape d'apprentissage, le réseau code un facteur sur les données qui peut être appliqué sur n'importe quelle image. La plupart des algorithmes de classification sont supervisés.

En résumé, les techniques d'apprentissage supervisé de classification et reconnaissance d'objets dans des images prennent des jeux de données d'images en entrée, ainsi que des labels de référence. Ces labels sont en général des nombres ou des mots. Dans le cadre des techniques de restauration d'images, les données d'entrées sont des images à restaurer, et la vérité terrain des images de référence.

Dans la suite de la thèse, les données d'entrée sont des images à restaurer, et la vérité terrain des images de références.

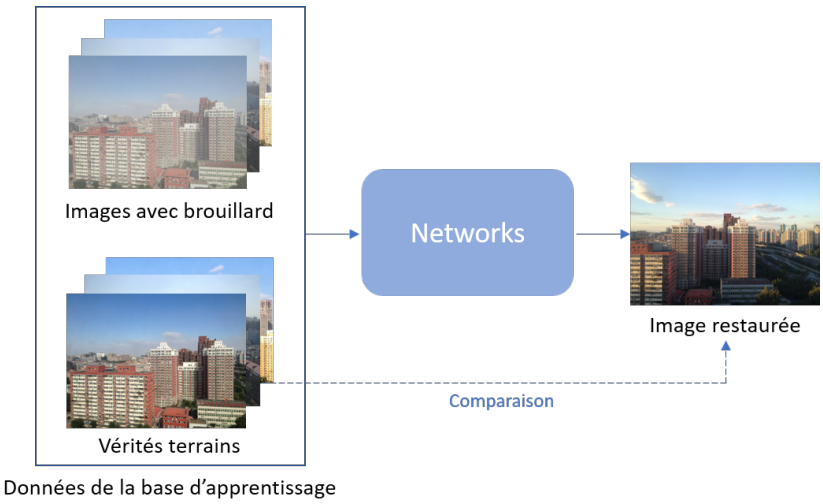


FIGURE 2.8 – Apprentissage supervisé d'un réseau pour la restauration d'images.

Dans le cadre des techniques de restauration de la visibilité dans le brouillard, il est nécessaire de fournir pour l'apprentissage du réseau de neurones des jeux de données d'images avec brouillard ainsi que des images vérités terrain sans brouillard. La figure 2.8 illustre le principe d'apprentissage. L'objectif est de construire une architecture neuronale adaptée à notre application. Il en existe plusieurs sortes qui peuvent être composées de couches de convolution différentes. L'état de l'art ne cesse de croître avec de nouveaux mécanismes et types d'architectures différentes. Le challenge est de déterminer ce qui va composer l'architecture de manière à optimiser les résultats.

2.2.2.2 L'atténuation du brouillard basée sur des techniques de deep learning

Avec le développement de l'apprentissage profond, les réseaux de neurones convolutifs (CNN) ont été utilisés dans de nombreux domaines et ont obtenu des résultats fort intéressants, notamment en détection et reconnaissance d'objets ou en segmentation d'images. L'atténuation de brouillard est un domaine qui a également attiré l'attention. Certains travaux ont été réalisés dans un premier temps avec des CNNs dans le but de produire directement les cartes de transmission à partir de jeux de données d'entrées pour l'apprentissage d'*a priori* sur le voile. Les images sans brouillard sont ensuite calculées à partir du modèle physique en inversant l'équation de Koschmieder (2.7). Cai et al. (2016) ont proposé une technique, appelée DehazeNet, qui prend en entrée des images avec du brouillard et qui permet de créer des cartes de transmission en sortie. Une technique d'extraction de caractéristiques (le *Dark Channel*, la disparité de teinte et l'atténuation de couleurs) et de cartographie multi-échelles ont notamment été utilisées. Cependant, les cartes de transmission ne sont pas raffinées dans la méthode proposée par Cai et al. (2016). Pour y remédier, des méthodes multi-échelles ont plus tard été proposées pour l'amélioration des cartes de transmission (Ren et al. (2016, 2020)). Cependant, cette méthode introduit une étape intermédiaire dans le processus de restauration d'image et nécessite des cartes de transmission très précises. Le manque de vérité terrain des cartes de transmission induit inévitablement un manque de précision qui pourrait se répercuter sur les résultats de la restauration.

L'apprentissage partiel des paramètres intermédiaire, comme celui de la carte de transmission, peut ne pas prendre en compte les *a priori*. Ainsi, beaucoup de travaux fondés sur des techniques de *deep learning* ont été rendu plus flexibles afin d'éviter ces étapes intermédiaires de calcul. De nouvelles méthodes de restauration « de bout en bout » ont ensuite été proposées, estimant directement des images sans brouillard à partir des images d'entrées. Li et al. (2017) ont été l'un des précurseurs dans l'emploi de ces méthodes avec AOD-Net. Composé d'un module d'estimation, qui comprend les paramètres de transmission et d'atténuation, et d'un module de génération d'images sans brouillard, le réseau est appris par minimisation de la fonction de coût.

Depuis l'apparition de ces premiers algorithmes, l'émergence de nouvelles architectures et de nouveaux mécanismes conduisent à des méthodes de plus en plus complexes. Dans la suite de l'état de l'art, des mécanismes essentiels pour la compréhension de la suite de la thèse sont introduits.

2.2.2.3 Les LSTM convolutionnels

Lorsque nous lisons une phrase, notre compréhension du mot courant se fait en fonction des mots que nous avons lu précédemment. On peut dire qu'il existe une interdépendance entre ces mots. Les réseaux de neurones traditionnels ne permettent pas de prendre en compte l'interdépendance au sein d'une série chronologique (données collectées sur une période de temps successive). Ce type de donnée est généralement traité par un modèle fondé sur une architecture de réseau de neurones récurrents (RNN). Les RNN utilisent des boucles permettant de conserver les informations pendant un certain temps.

La figure 2.9 présente le schéma d'un RNN déplié pour la compréhension de son fonctionnement. Une portion de réseau de neurone A prend une entrée x_t et produit une valeur h_t. La boucle permet de transmettre l'information d'une couche du réseau à une autre.

Les RNNs peuvent s'appliquer à une grande diversité d'applications. Cependant, l'inconvénient majeur est l'impossibilité, en pratique, de modéliser des dépendances à long terme.

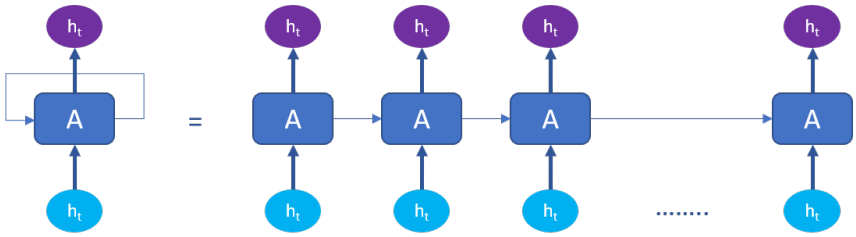


FIGURE 2.9 – Schéma d'un RNN inspiré des explications de https://colah.github.io/posts/2015-08-Understanding-LSTMs/.

Le *LSTM* (en anglais, Long Short Term Memory), présenté à la figure 2.10, composé d'une architecture basée sur les RNNs, a été introduit par Hochreiter and Schmidhuber (1997) pour résoudre ce problème. Alors qu'un RNN classique consiste à appliquer une fonction d'activation entre l'entrée et la sortie de la cellule précédente, la conception d'un LSTM est plus complexe en raison de l'introduction d'un système de portes qui régule l'information qui entre et sort de la cellule *Cell* (voir figure 2.10).

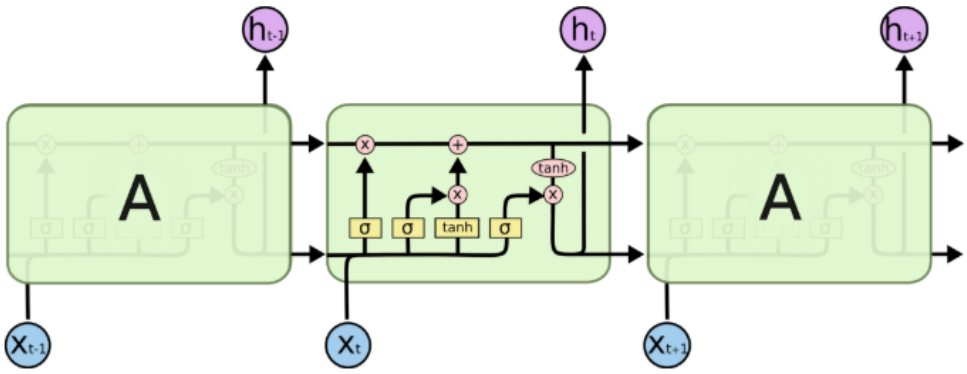


FIGURE 2.10 – Schéma d'une structure LSTM.

Cette cellule est illustrée par la ligne horizontale qui traverse le haut du bloc central de la figure 2.10, appelée *Cell state*, et indique l'état de la cellule dans le temps. Les informations peuvent être ajoutées ou supprimées en fonction de leur pertinence.

Les LSTMs prennent en entrée des séquences de caractéristiques à une seule dimension. Or, nous travaillons avec des séquences vidéos. Pour répondre à cette problématique, Shi et al. (2015) ont mis en œuvre l'architecture ConvLSTM, qui possède une structure interne similaire à celle d'un LSTM, et prend en entrée des matrices 2D. Certains travaux en restauration de vidéos reposent sur des architectures de type RNN pour capturer l'information temporelle des séquences vidéo. Zhong et al. (2021) ont utilisé des RNNs pour l'atténuation du flou dans des vidéos. Ils ont proposé d'ajouter des blocs denses résiduels dans des cellules RNN, de manière à extraire efficacement les caractéristiques spatiales des trames. De plus, un module d'attention spatio-temporel global est proposé pour fusionner les caractéristiques hiérarchiques des images passées et futures, afin d'améliorer le processus d'atténuation du flou.

Zhang et al. (2020b) ont proposé une méthode de super résolution dans des vidéos en introduisant un réseau de fusion d'entités spatio-temporelles multi-étages. Le réseau est organisé en plusieurs étapes, de sorte que la corrélation temporelle des caractéristiques à différentes étapes du réseau puisse être pleinement exploitée. De plus, des couches de convLSTM sont utilisées pour capturer l'information temporelle à la fin de chaque étape.

2.2.2.4 Le mécanisme d'attention

Le mécanisme d'attention en *deep learning* est inspiré du fonctionnement du cerveau humain. Il est pourvu d'une capacité d'apprentissage qui lui permet de se focaliser sur des zones qui recèlent les informations les plus pertinente dans un jeu de données. Cela peut être une région spécifique d'une image ou un mot dans une phrase. Il a été introduit par Bahdanau et al. (2016) dans le but d'améliorer les performances d'un modèle de type auto-encodeur pour la traduction automatique du langage.

Le but du mécanisme d'attention est de permettre au décodeur d'utiliser les parties les plus pertinentes de la séquence d'entrée, en utilisant un système de pondération. Ainsi, plus les poids d'attention sont élevés, plus la région correspondante est représentative pour l'application visée.

De nombreux travaux récents en classification et en restauration d'images et de vidéos ont repris ce mécanisme dans leurs architectures afin que les réseaux apprennent à se focaliser sur des caractéristiques spécifiques les plus informatives des images. Sur la figure 2.11, l'attention est représentée par des halos blancs. Ces images montrent que, dans un contexte de reconnaissance d'objet, l'algorithme a su se focaliser sur les parties pertinentes des images.

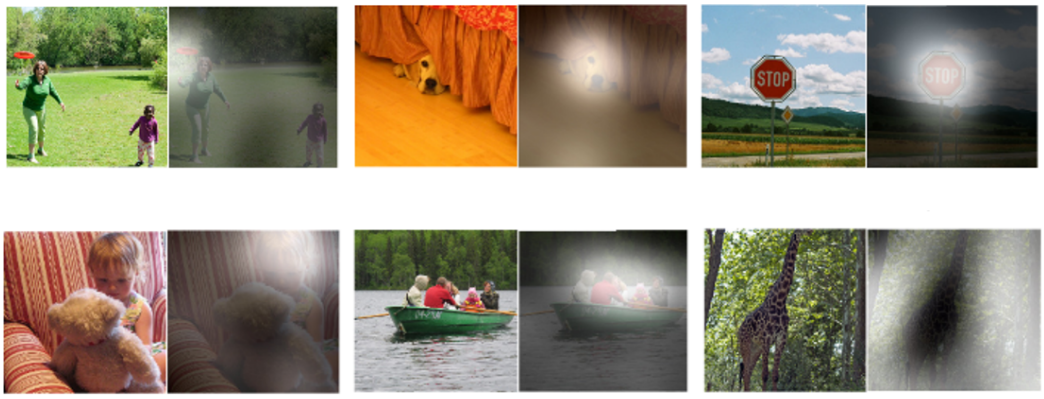


FIGURE 2.11 – Focalisation de l'attention dans certaines zones de l'image. Les images proviennent du site : https://medium.com/heuritech/attention-mechanism-5aba9a2d4727.

En plus d'améliorer les performances des modèles, ce mécanisme permet de faciliter l'interprétation des résultats obtenus à l'aide d'algorithmes basés sur des réseaux de neurones. Le succès grandissant du mécanisme d'attention est à l'origine de l'émergence de nouvelles techniques de restauration d'images.

Qin et al. (2019) ont proposé une méthode d'atténuation du brouillard utilisant ce mécanisme d'attention. Il s'agit d'un réseau de fusion de caractéristiques de bout en bout, appelé *FFA-Net*. Il est composé d'une structure de fusion à différents niveaux. Les poids associés aux caractéristiques des images sont appris de manière adaptative à partir du module d'attention, en espérant donner plus de poids aux caractéristiques importantes du brouillard. Ce module combine les mécanismes d'attention des canaux et des pixels, permettant de traiter différents types d'informations. Les auteurs s'appuient notamment sur l'hypothèse que le brouillard n'est pas uniformément réparti dans l'image ni au travers des canaux de couleurs. Cette méthode est particulièrement efficace pour un certain type de données, mais les performances diminuent sur une base de tests avec des images aux caractéristiques de brouillard différentes de la base

d'apprentissage. En effet, une évaluation quantitative a été réalisée avec l'algorithme présenté dans le chapitre 3, et la méthode de Qin et al. (2019) a montré qu'elle permettait parfaitement d'atténuer le brouillard sur des images au brouillard léger et uniforme. Cependant, les performances diminuent sur un jeu de données composé d'images avec un brouillard plus dense et inhomogène.

2.2.2.5 Les *Transformers*

Le *Transformer* est une architecture neuronale récente utilisant exclusivement le mécanisme d'attention dans le but d'améliorer les performances des modèles. Il a été introduit dans l'article « *Attention is all you need* » par Vaswani et al. (2017) et s'est imposé comme une architecture incontournable dans la plupart des applications de données textuelles. L'architecture *Transformer* a été proposée comme un moyen de pallier les limitations des architectures de modélisation de séquences (images, mots). En effet, les réseaux de neurones récurrents (RNN), comme le *LSTM* (Hochreiter and Schmidhuber (1997)) ou le *GRU* (Cho et al. (2014)), utilisés dans des applications comme la traduction des langues, le traitement du langage naturel et la restauration d'images et de vidéos (Zhong et al. (2021), Zhang et al. (2020b)), présentent certaines contraintes et limites : la perte de mémoire et l'impossibilité de paralléliser des tâches. En effet, un évènement lointain dans la séquence a peu d'influence sur la sortie. Contrairement aux RNNs, le *Transformer* permet de traiter toute la séquence en une fois, et peut paralléliser certaines opérations. L'attention permet notamment de conserver l'interdépendance au sein d'une séquence.

Cette architecture a, par la suite, été utilisée dans diverses applications de vision par ordinateur. Dosovitskiy et al. (2021) ont introduit le *Vision Transformer* (ViT), une architecture de classification d'images basée sur les *Transformers*. Cette méthode sert de point de départ à la plupart des travaux pour les applications d'images fondées sur cette architecture.

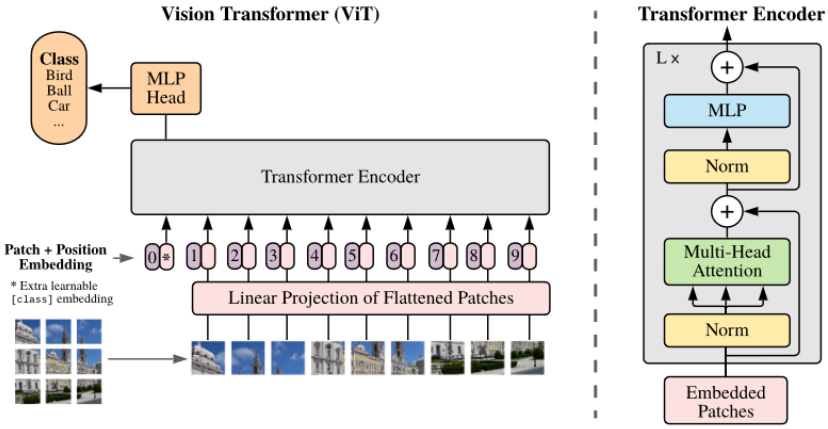


FIGURE 2.12 – Aperçu du modèle ViT, d'après Vaswani et al. (2017).

La figure 2.12 présente un schéma du modèle ViT pour une tâche de classification d'images. Des images subdivisées en patches de taille fixe sont fournies en entrée du *Transformer* et intégrées avec leur positions respectives (de plus amples précisions seront fournies dans le chapitre 4). Le mécanisme central de l'architecture *Transformer* repose sur la couche *Multi-Head-Attention* (MHA) correspondant au bloc vert à droite (*Transformer Encoder*) dans la figure 2.12. Le schéma du bloc est présenté à la figure 2.14b et va être détaillé dans la suite de ce paragraphe.

L'Attention mise à l'échelle du produit scalaire (en anglais, *Scaled Dot-Product Attention*) proposée par Vaswani et al. (2017) dans l'article « *Attention is all you need* », a été un point de départ à la construction du mécanisme de *Multi-Head-Attention*, et correspond au bloc violet de la figure 2.14. Pour faciliter la compréhension de ce mécanisme, je prends un exemple d'application en traduction du langage.

L'Attention peut être divisée en trois composants appelés : requête, clé et valeur (en anglais, *Query, Key, Value*). Ces vecteurs q, k et v sont attribués à chaque mot d'une séquence. Le modèle se focalise sur le vecteur q d'un mot M de la séquence d'entrée. Ce processus peut être assimilé à la requête suivante : avec quel mot de la séquence appartenant au vecteur k ce mot M est le plus corrélé ? Un score v est ainsi établit entre le mot M de la requête et les clés k, correspondants aux mots de la séquence. Les scores sont mis à jour à chaque nouvelle requête, de valeur v. Pour chaque mot considéré, une valeur d'attention est calculée. Un exemple est présenté à la figure 2.13.

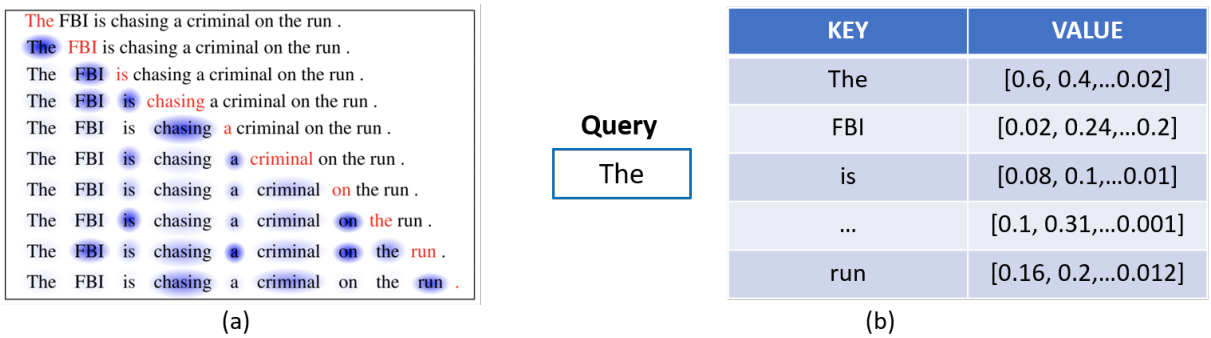


FIGURE 2.13 – Illustration du mécanisme d'attention : (a) le mot courant M est en rouge et les mots avec lesquels les scores d'attention sont élevés sont en bleus. L'image provient de l'article de Cheng et al. (2016), (b) Illustration des rôles des matrices Q, K et V associée à l'exemple (a).

Dans la figure 2.14b, le bloc *Scaled Dot-Product* calcule les produits scalaires entre les requêtes q et l'ensemble des clés k. Il divise ensuite chaque résultat par $\sqrt{d_k}$, correspondant à la dimension du vecteur k, et procède à l'application d'une fonction softmax pour générer des poids. Ces poids sont mis à jour dans le vecteur v. Ce calcul est réalisé sur l'ensemble des vecteurs q simultanément. Soient les matrices Q, K, V (dont la taille dépend de la taille de la séquence d'entrée), la formule de l'Attention est définie par :

$$Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2.14}$$

où K^T est la transposée de la matrice K. Cette équation indique que plus la valeur du produit entre Q et K est élevée, plus la similarité est importante. Plus la valeur de softmax est élevée plus les scores d'attention seront élevés.

Multi-Head Attention signifie Attention Multi-tête. Elle permet de réaliser des calculs d'attention dans différents sous-espaces de représentation, appelées têtes (illustrées par la lettre h pour *head* dans la figure 2.14). En effet, Vaswani et al. (2017) ont trouvé opportun de projeter linéairement les matrices Q,K, V h fois en parallèle avec différentes projections linéaires apprises aux dimensions d_q, d_k et d_v, respectivement, au lieu d'effectuer un simple calcul d'attention. La fonction *MHA* est définie comme suit :

$$\begin{aligned} MultiHead(Q,K,V) &= Concat(head_1,...,head_h)W^O \\ head_i &= Attention(QW_i^Q, KW_i^K, VW_i^V) \end{aligned} \tag{2.15}$$

où W^Q, W^K, W^V désignent les matrices de projection utilisées pour générer les représentations des sous-espaces des matrices Q, K, V. W_O correspond à la matrice de projection de la sortie de la fonction MHA. i correspond au numéro de la tête d'attention.

La figure 4.22 illustre un exemple de découpage de la séquence d'images en imagettes. Pour chaque imagette encodée de la séquence, les matrices Q, K, V sont générées. Chaque vecteur de requête Q est comparé à toutes les clés K. De cette manière, chaque imagette de chaque image de la séquence est comparée avec les imagettes de l'ensemble des images de la séquence. Les vecteurs Q et K sont considérés comme similaires s'ils entretiennent un lien ou une connexion mutuelle. Cette similarité est calculée avec le produit scalaire des lignes et des colonnes de la matrices QK^T, illustrée à la figure 2.14a.

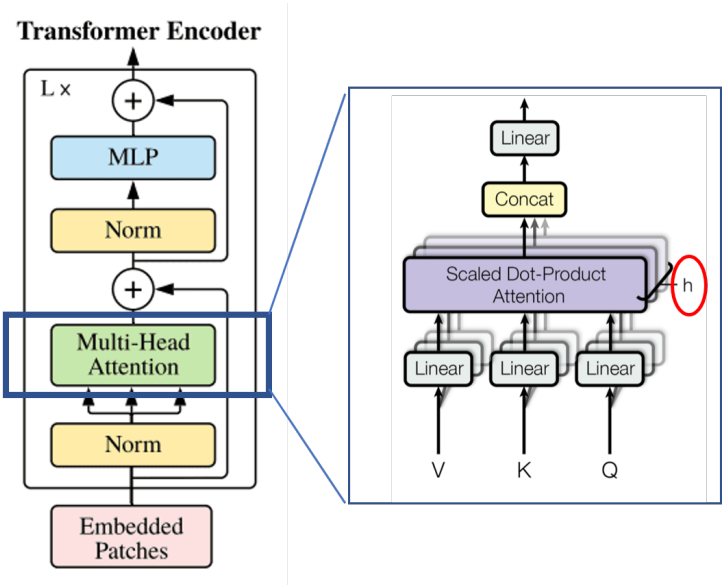


FIGURE 2.14 – Schéma de la couche MHA, d'après Vaswani et al. (2017).

En résumé, le calcul de l'attention est réalisé par différents blocs h en parallèle, appelés « tête ». Chaque tête va traiter différentes parties de la séquence d'entrée. L'ensemble de ces calculs forment ensuite un score d'attention final.

Cette approche basée sur des images a ensuite été étendue à des tâches de segmentation, de restauration, ou encore de classification d'objets. Encore peu de chercheurs utilisent les *Transformers* dans leurs travaux pour l'atténuation du brouillard dans des images et des vidéos.

Zhao et al. (2022) ont été les premiers avec leur méthode hybride *HyLoG-ViT*. Cette méthode comporte deux branches principales permettant de capturer les dépendances locales et globales des images avec du brouillard. La figure 2.15 présente le diagramme du bloc hybride local-global présent dans l'encodeur. L'architecture globale se compose d'un encodeur et de trois décodeurs. Deux décodeurs permettent d'obtenir une composante de réflectance (qui contient les informations de couleur de la scène) et une composante d'ombre (qui contient les informations de structure et de texture de la scène). Les données de deux décodeurs sont ensuite

fusionnées avec les données du troisième décodeur, dans le but d'obtenir l'image restaurée. Zhao et al. (2022) ont mis l'accent sur le fait que le *ViT* standard ne possède pas de localité et ont tenté de remédier à ce problème avec leur méthode hybride. Li et al. (2022) ont proposé un réseau construit avec des *Swin-Tranformers* en deux étapes. Le *Swin-Tranformers*, introduit par Liu et al. (2022b), est une variante du ViT qui possède une architecture hiérarchique ayant la capacité de modéliser à différentes échelles. Ce réseau à une complexité de calcul linéaire par rapport à la taille de l'image. La première étape consiste en une extraction de caractéristiques de base des images avec brouillard, et la deuxième correspond à une extraction de caractéristiques locales. Song et al. (2022) ont proposé l'architecture *DehazeFormer*, fondée sur des *Swin-Tranformers*. Plusieurs améliorations ont été intégrées afin de compenser les lacunes des *Swin-Tranformers* d'origine appliqués à la restauration d'images sans brouillard. Valanarasu et al. (2021) ont proposé *Transweather*, un modèle basé sur le *Transformer* composé d'un encodeur et d'un décodeur capable de restaurer une image dégradée par n'importe quelle condition météorologique.

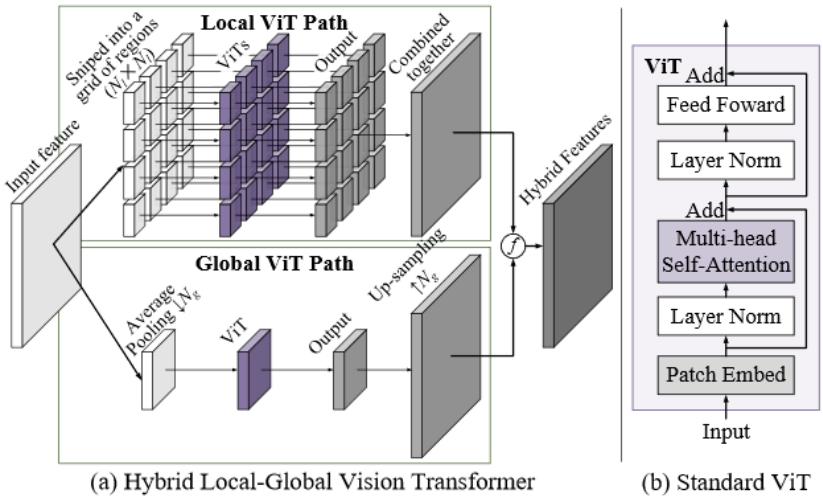


FIGURE 2.15 – Comparaison des schémas de la méthode *HyLoG-ViT* (a) et de la méthode classique *ViT* (b). L'image est extraite de l'article de Zhao et al. (2022).

En s'appuyant sur le succès récent des modèles de classification d'images, Arnab et al. (2021) ont présenté des modèles basés sur des *Transformers* purs pour la classification de vidéos. Ils proposent notamment des variantes de leur modèle pour la gestion des aspects spatio-temporels des séquences vidéo. De plus amples précisions seront apportées dans le chapitre 4. Liang et al. (2022) ont proposé une méthode de restauration de vidéos basée sur l'architecture *Transformers* (VRT) où chaque échelle est composée de deux modules : l'attention mutuelle temporelle (TMSA) et la déformation parallèle. L'attention mutuelle permet l'alignement entre trames voisines, tandis que l'attention est utilisée pour extraire les caractéristiques des images. À chaque échelle, le module de déformation parallèle est utilisé pour la fusion des informations de trames voisines avec la trame courante. Cette méthode présente de très bonnes performances pour les applications de défloutage, de débruitage ou encore de super-résolution.

2.2.2.6 Les convolutions déformables

Les CNN ont une capacité limitée à prendre en compte les variations géométriques telles que l'échelle, la pose ou la déformation des objets. Ceci provient de l'utilisation d'un noyau

fixe lors de l'échantillonnage des cartes caractéristiques. Les convolutions déformables ont été introduites par Dai et al. (2017) afin de prendre en compte ces variations géométriques.

Un système de décalage 2D a été ajoutés au niveau du noyau d'échantillonnage pour permettre aux champs récepteurs de se déformer et de s'adapter dynamiquement aux différentes formes et variations des objets de l'image d'entrée. La figure 2.16 illustre le principe de déformation des noyaux d'échantillonnage. Comparé aux convolutions standards, les convolutions déformables s'adaptent à la forme des objets.

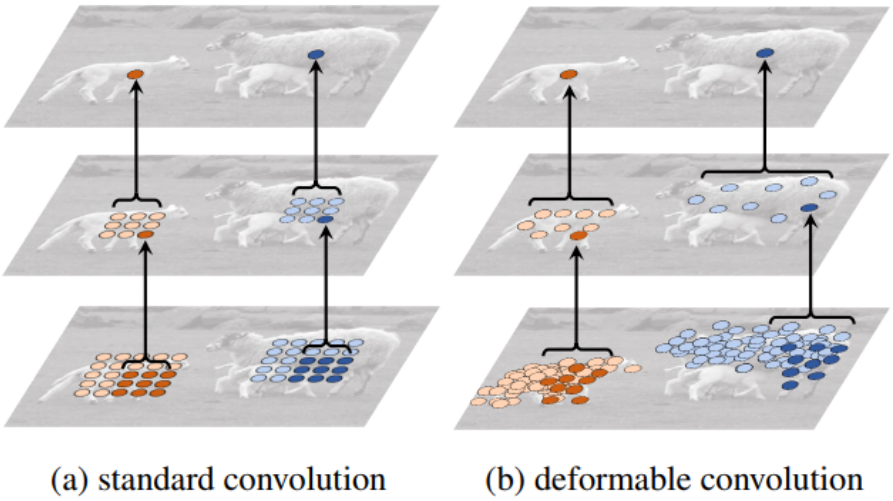


FIGURE 2.16 – Exemple d'application des convolutions déformables. L'illustration provient du site : https://towardsdatascience.com/deformable-convolutions-demystified-2a77498699e8.

Wang et al. (2019b) ont proposé une méthode de restauration de vidéos appelée EDVR. Elle comporte un module d'alignement pyramidal, en cascade et déformable (*PCD Align Fusion module* dans la figure 2.17). L'utilisation de convolutions déformables permet de réaliser l'alignement des trames successives au niveau des caractéristiques des images. Le deuxième module principal est le module de fusion spatio-temporel (*TSA Fusion module* dans la figure 2.17), basé sur une technique d'Attention. Les relations temporelles et spatiales inter-images sont essentielles dans le processus de fusion. Les images adjacentes ne fournissent pas forcément les mêmes informations (occlusions). De plus, un mauvais alignement peut affecter négativement les performances de restauration. Par conséquent, l'agrégation dynamique des images voisines au niveau du pixel est indispensable pour une fusion efficace. Cette étape est réalisée par le module TSA.

Comme il est précisé dans l'article de Wang et al. (2019b), cette structure peut s'adapter à diverses tâches de restauration vidéo comme la super-résolution ou le défloutage. Le module PCD souffre cependant d'instabilité à l'entraînement et échoue avec des images de haute résolution, d'après Zhang et al. (2022). Ces derniers ont alors proposé une version du module PCA pour corriger ces problèmes. Cette méthode a été proposée pour la restauration de vidéos dans le brouillard et comporte trois modules principaux. Le premier module, *Confidence guided pre-dehazing*, a été créé d'après l'hypothèse suivante : la densité du brouillard peut varier entre les images adjacentes. Ce module sert alors de pré-traitement des entrées, dans le but d'améliorer

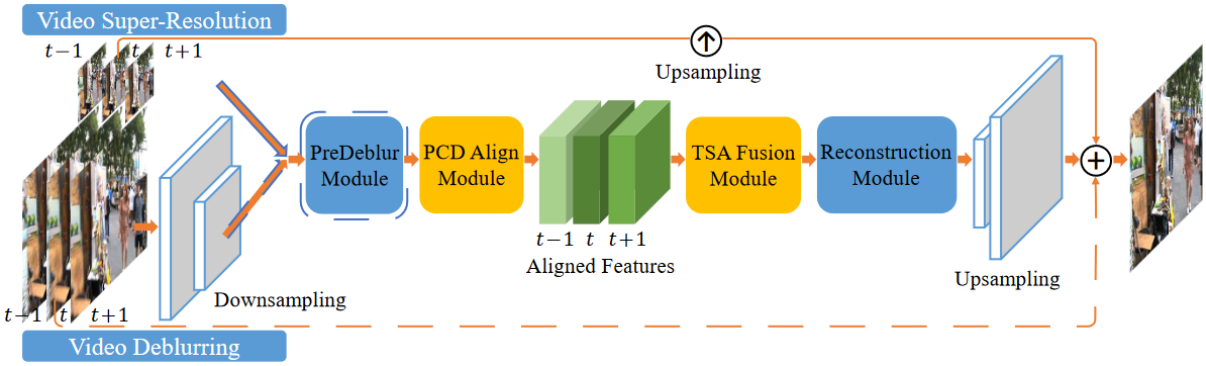


FIGURE 2.17 – Schéma de l'algorithme EDVR

les performances du module d'alignement (l'équivalent du PCA amélioré). Pour exploiter pleinement la redondance temporelle, un module de *Multi-Feature Fusion* (MFF) est proposé pour fusionner les images adjacentes alignées avant l'étape de restauration.

2.2.2.7 Les *GANs*

Le *GAN (Generative Adversarial Network)* est un algorithme d'apprentissage non-supervisé introduit par Goodfellow et al. (2014). Le *GAN* est composé de deux réseaux de neurones qui réalisent un apprentissage simultané, l'un contre l'autre, dans le but de générer des nouvelles données. Ce type de réseau a été utilisé dans un premier temps en génération d'images, puis, dans un second temps, il a été généralisé à d'autres types d'application tels que la restauration d'images.

Ces deux réseaux de neurones sont appelés générateur et discriminateur. Le générateur génère des exemples nouveaux et le discriminateur classifie les exemples comme vrai ou faux par rapport aux données de la base d'apprentissage. Pendant la phase d'entraînement, les deux modèles sont en compétition. Le but du générateur est de produire des exemples les plus conformes à la base de départ possibles, tandis que le discriminateur doit identifier les résultats faux. Le générateur est d'abord initialisé avec des variables aléatoires dont le résultat est transmis au discriminateur, qui ajuste les poids en fonction du résultat (vrai ou faux) et les transmet de nouveau au générateur (voir la figure 2.18).

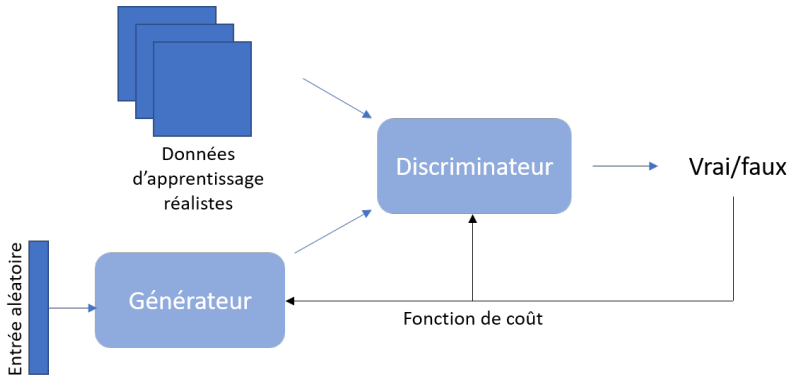


FIGURE 2.18 – Schéma d'une structure *GAN*

La figure 2.19 présente un exemple de structure de type *GAN* pour la restauration d'images dans le brouillard. L'image en entrée du générateur est une image réaliste avec du brouillard.

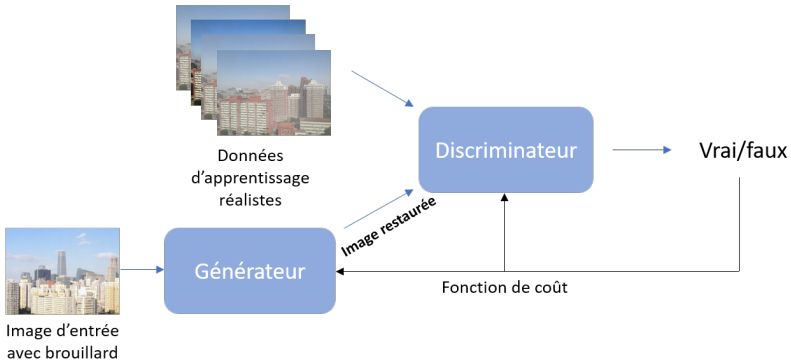


FIGURE 2.19 – Schéma d'une structure *GAN* pour une application de restauration d'image dans le brouillard.

Pour s'assurer que le résultat en sortie du générateur soit assez performant, le discriminateur le compare avec les données réelles de la base d'apprentissage.

Qu et al. (2019) ont proposé une amélioration de l'architecture de réseau pix2pix (Wang et al. (2018)), appelé EPDN, pour la suppression de brouillard dans les images. Il intègre dans un réseau de type *GAN*, deux modules de post-traitement. La partie *GAN* est composée d'un générateur multi-résolution et d'un discriminateur multi-échelles, pour créer une image pseudo-réaliste à une échelle grossière. Les deux modules d'amélioration qui suivent doivent produire une image réaliste sans brouillard. Basés sur le modèle de champ réceptif, ils renforcent la restauration à la fois dans la couleur et les détails.

Dong et al. (2020) ont proposé un réseau antagoniste génératif avec un discriminateur de fusion (FD-GAN) pour la restauration de brouillard dans les images. Le discriminateur de fusion proposé utilise les informations fréquentielles des images comme *a priori* supplémentaires, permettant de générer des images dévoilées plus naturelles et réalistes avec moins de distorsion des couleurs et moins d'artefacts.

2.2.3 Les méthodes hybrides

Certaines méthodes tentent de combiner les avantages des méthodes traditionnelles basées sur un modèle physique et celles des approches par apprentissage. Par exemple, en réalisant l'apprentissage des *a priori* ou en utilisant ces *a priori* dans la fonction de coût.

Yang and Sun (2018) figurent parmi les premiers à avoir eu recours à une architecture CNN pour l'apprentissage d'*a priori*, ici le *Dark Channel* et la transmittance. Cette méthode permet d'apprendre de manière plus discriminante les *a priori* relatifs à la restauration de brouillard à partir de données d'apprentissage. Il s'agit d'une méthode multi-étage pour l'estimation du *DCP*, de la carte de transmission et de l'image restaurée résultante. Plus tard, les auteurs ont proposé une extension de leurs travaux par une reformulation du modèle, en introduisant un module supplémentaire d'apprentissage de l'image sans brouillard (Yang and Sun (2021)). Le calcul du voile atmosphérique (*airlight*) a également été introduit dans le processus d'apprentissage. De plus, les performances ont été améliorée et d'autres évaluations ont été ajoutées.

Certains ont recours à des méthodes basées sur des modèles physiques pour mettre en place leur fonction de coût. Golts et al. (2020) introduisent une méthode non-supervisée basée sur un réseau CNN avec des convolutions dilatées et ont utilisé le *DCP* dans la fonction de coût. Chen et al. (2021) proposent une méthode (PSD) d'amélioration des performances de la restauration d'images avec du brouillard. Elle est déclinée en deux sous-modèles : un premier modèle pré-

entraîné sur des données de synthèses, et un deuxième modèle PSD qui exploite des données réelles afin de raffiner le modèle pré-entraîné de manière non-supervisée. Les résultats montrent que l'utilisation de la combinaison de plusieurs *a priori* physiques dans la fonction de coût (incluant le *DCP*) améliore les performances de restauration. Pour intégrer le *DCP* dans la fonction de coût, ils ont reformulé cet *a priori* en fonction d'énergie :

$$L_{DCP} = E(t,\tilde{t}) = t^T Lt + \lambda (t-\tilde{t})^T (t-\tilde{t}) \tag{2.16}$$

où t et $\tilde{t}$ correspondent respectivement à la transmittance estimée avec l'algorithme *DCP* et leur réseau. L est une matrice de type Laplacien. Finalement, pour éviter des images restaurées trop sombres, leur fonction de coût est formulée par une combinaison de trois *a priori*.

2.2.3.1 Concaténation/fusion par canaux des trames adjacentes dans une vidéo

Des travaux ont montré qu'il était possible de modéliser l'information temporelle en empilant, par concaténation le long d'un axe, des trames consécutives à l'entrée d'un réseau.

Ren et al. (2019) ont proposé un réseau basé sur une architecture de type auto-encodeur, exploitant les informations obtenues entre les trames d'une séquence, par concaténation, pour l'estimation de la carte de transmission. Une particularité de la méthode est qu'elle exploite les informations sémantiques de l'image afin de mettre en valeur les discontinuités de l'image, tout en lissant les zones uniformes. Les images de la séquence vidéo sans brouillard sont ensuite générées à partir du modèle physique de Koschmieder. Park and Kim (2018) proposent une méthode basée sur plusieurs schémas de restauration rapide de vidéos dans le brouillard qui s'appuie sur la méthode *DCP*. Le voile atmosphérique (*airlight*), assure la cohérence temporelle entre les trames consécutives car il évolue très lentement. La valeur de l'*airlight* est mise à jour lorsque sa valeur change trop rapidement entre les trames.

Li et al. (2018a) proposent une méthode de suppression de brouillard dont l'objectif est d'estimer directement l'image restaurée sans passer par les étapes intermédiaires d'estimation de la carte de transmission, en exploitant la cohérence temporelle entre les trames vidéo consécutives. Le cœur de la méthode est basé sur leur travaux antérieurs (Li et al. (2017)). Ils reposent notamment sur l'estimation conjointe de la carte de transmission et de l'image restaurée via un module qui unifie les paramètres de transmittance et d'intensité du ciel. Dans cette méthode, les modules de restauration et d'alignement sont fusionnés.

Li (2021) propose une méthode d'alignement et de restauration progressive pour la suppression du brouillard des vidéos, méthode qui repose sur un mécanisme de fusion. Cette méthode permet l'alignement des trames voisines consécutives étape par étape sans utiliser de technique d'alignement explicite, comme le flot optique. Le processus de restauration est non seulement mis en œuvre dans le cadre du processus d'alignement, mais il utilise également un réseau de raffinement pour améliorer les performances du réseau. Le réseau proposé comprend quatre sous-réseaux de fusion et un sous-réseau de raffinement.

2.2.4 Les méthodes d'évaluation

2.2.5 Les fonctions de perte

La fonction de coût (en anglais, *Loss function*) est utilisée pour optimiser un modèle. Le choix de cette fonction affecte significativement les performances des algorithmes de *deep learning*. Elle évalue l'erreur entre les prédictions et les valeurs réelles (un label, une vérité terrain)

utilisées lors de l'étape d'apprentissage. Plus cette erreur est minimisée sur la base d'apprentissage, meilleur seront les résultats obtenus avec cet algorithme.

Sur les images, la fonction *loss* MSE (*Mean Squared Error* ou L2), très populaire, s'est révélée produire des images floues avec des artefacts. En effet, elle pénalise les erreurs importantes, mais reste tolérante aux petites erreurs. D'autres fonctions de perte ont été proposées pour pallier ce problème.

2.2.5.1 L1 *loss*

La fonction de coût L1 est définie par la fonction suivante :

$$L1(P) = \frac{1}{N} \sum_{p \in P} |\bar{y}(p) - y(p)| \tag{2.17}$$

où p est l'indice du pixel et P correspond à une image (ou patch). $\bar{y}(p)$ et $y(p)$ sont respectivement les valeurs des pixels de l'image estimée et de la vérité terrain. Cette fonction pénalise moins les erreurs importantes et semble mieux conserver les couleurs par rapport à la fonction MSE.

2.2.5.2 (MS-)SSIM *loss*

Le SSIM (en anglais, *Structural Similarity Index Metric*) de Wang et al. (2004) est une métrique populaire également utilisée comme fonction de coût. Cette méthode est basée sur la vision humaine et mesure la similarité entre l'image dégradée et sa référence. La dégradation d'une image par rapport à sa référence est définie par un changement de perception de l'information structurelle qui prend en compte la luminance et le contraste dans l'image. Elle se définit par :

$$L_{SSIM}(P) = \frac{1}{N} \sum_{p \in P} (1 - SSIM(p)) \tag{2.18}$$

La formule de SSIM(p) est la suivante :

$$SSIM(p) = [l(p)]^{\alpha}.[c(p)]^{\beta}.[s(p)]^{\lambda} \tag{2.19}$$

où l correspond à la luminance, c au contraste, s à la structure. α, β, λ sont des paramètres à ajuster en fonction de l'importance des trois composantes de l'équation et doivent être strictement supérieurs à zéro.

Plus spécifiquement et si l'on considère $p = (x, y)$, la formule du SSIM prend la forme suivante :

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2\mu_y^2 + C_1)(\sigma_x^2\sigma_y^2 + C_2)} \tag{2.20}$$

où μ_x et μ_y sont les moyennes de l'intensité selon x et y, σ_x^2 et σ_y^2 les variances de l'intensité selon x et y, et C_1 et C_2 des constantes qui évitent la division par zéro lorsque $(\mu_x^2\mu_y^2 + C_1)$ et $(\sigma_x^2\sigma_y^2 + C_2)$ sont très proches de zéro.

Cette métrique surpasse les performances du MSE et a été reprise dans de nombreux travaux. En effet, elle préserve le contraste et la netteté dans les régions hautes-fréquences des images, au niveau des structures. Wang et al. (2003) ont étendu cette méthode après avoir observé un

lien (un facteur) comme la distance objets/observateur. Ils proposent donc MS-SSIM, une version multi-échelles de la métrique SSIM avec une pondération du SSIM calculée à différentes échelles (eq.2.21).

$$L_{MSSSIM}(P) = \frac{1}{N} \sum_{p \in P} (1 - MSSSIM(p)) \tag{2.21}$$

2.2.5.3 Les fonctions de coût basées sur les caractéristiques des images

Une nouvelle catégorie de fonctions de perte, qui utilise des réseaux de neurones comme extracteurs de caractéristiques, a récemment gagné en popularité. Ordinairement, la fonction est calculée comme la distance L2 entre les fonctions d'activations des couches cachées d'un réseau de classification d'images. Le plus couramment utilisé pour cette application est le réseau *VGG* (Simonyan and Zisserman (2014)), voir la figure 2.20.
L'équation correspondante est la suivante :

$$L_{perceptual}(P) = \sum_{i=1}^{n} ||(\phi(x) - \phi(y))||^2 \tag{2.22}$$

où $\phi(x)$ et $\phi(y)$ correspondent respectivement à l'espace des caractéristiques de l'image prédite et de la vérité terrain.

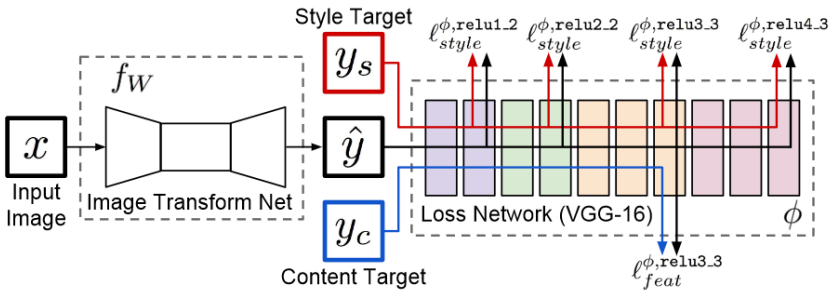


FIGURE 2.20 – Architecture réalisée par Johnson et al. (2016) qui utilise une fonction *perceptual loss* pour une application de transformation d'image.

Dans la figure 2.20, le réseau de droite définit une partie de la fonction de coût, avec une partie de reconstruction des caractéristiques l^{ϕ}_{feat}, et une partie de reconstruction du « style » l^{ϕ}_{style}. Cette fonction permet de mesurer la différence en termes de contenu et de style entre l'image en entrée et l'image en sortie.

Cependant, les principaux inconvénients de ces fonctions de perte sont leur dépendance à un grand réseau pré-entraîné ainsi qu'à l'objectif du réseau. Ils sont en effet plus adaptés aux algorithmes de classification. Mustafa et al. (2021) ont mis l'accent sur le point suivant : En plus de rendre l'étape d'apprentissage coûteux en mémoire, cette fonction se concentre sur les régions de l'image qui correspondraient plus spécifiquement à une tâche de classification d'images. Ils ont par ailleurs proposé une fonction de coût basée sur les caractéristiques des images plus spécifiques aux tâches de restauration d'images comme la super-résolution ou la compression. Les performances surpassent celles du *perceptual loss* habituelles pour des tâches de restauration d'images.

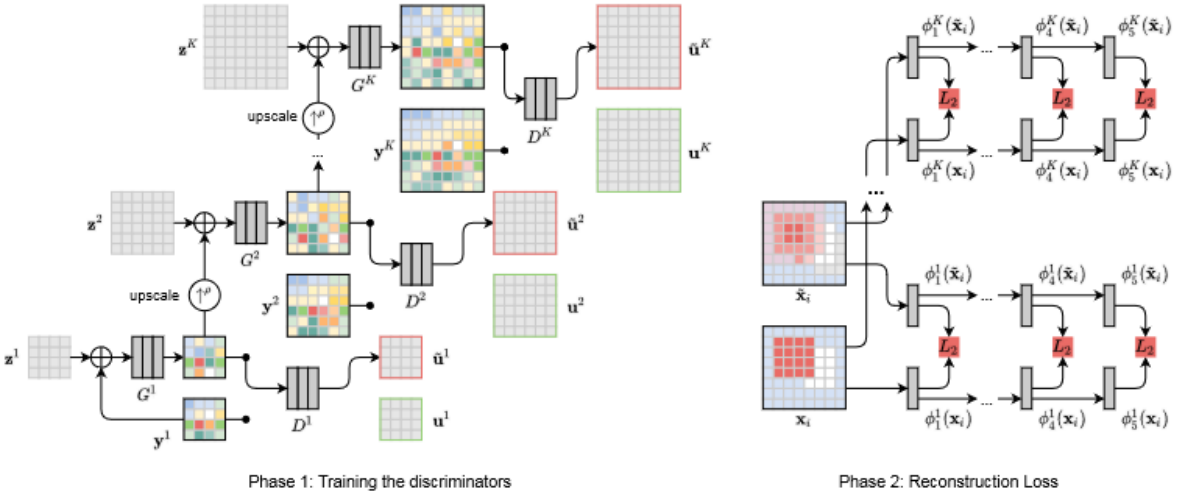


FIGURE 2.21 – Schéma de la méthode de Mustafa et al. (2021)

La figure 2.21 présente la méthode basée sur une architecture *GAN*. La première phase désigne l'apprentissage multi-échelles des discriminateurs. Dans deuxième phase, les discriminateurs sont fixés et utilisés comme extracteurs de caractéristiques. Un calcul de distance L2 est réalisé entre la vérité terrain x_i et la sortie restaurée $\bar{x}_i$ à chaque échelle k et les couches intermédiaires du discriminateur.

2.2.6 Les métriques d'évaluation

Les métriques permettent de juger les performances des algorithmes. Pour les applications de traitement d'images et de vidéos, nous utilisons des métriques pour évaluer la qualité des images obtenues (en anglais, *Image Quality Metrics*, IQMs). Parmi les IQMs, nous distinguons les métriques avec références, qui reçoivent en entrée l'image vérité terrain et l'image à évaluer, de celles sans références qui reçoivent uniquement en entrée l'image à évaluer. Parmi les méthodes existantes, il y a celles qui mesurent la différence entre la vérité terrain et l'image restaurée au niveau du pixel, comme le RMSE (en anglais, *Root Mean Squared Error*) ou encore le PSNR (en anglais, *Peak Signal to Noise Ratio*). L'équation du PSNR est la suivante :

$$PSNR = 10.log_{10}\frac{m^2}{MSE} \tag{2.23}$$

où m correspond à la valeur maximum d'un pixel dans une image. $m = 255$ pour une image 8-bits.

Il existe également des métriques qui détectent les différences structurelles dans les images comme le SSIM et le MS-SSIM (Wang et al. (2003), Wang et al. (2004)) introduit dans le paragraphe précédent, et la métrique FSIM qui mesure la similitude entre les caractéristiques de l'image dégradée et de sa référence (Lin Zhang et al. (2011)). La dernière méthode fait intervenir deux critères, la congruence de phase PC et l'amplitude du gradient S. L'équation de la métrique FSIM est définit par :

$$FSIM = \frac{\sum_{x\in\Omega} S_L(x).PC_m(x)}{\sum_{x\in\Omega} PC_m(x)} \tag{2.24}$$

où Ω correspond au domaine spatial de l'image.

Les fonctions loss peuvent être utilisées comme métrique et vice versa.

Dans la suite de la thèse, j'utilise des métriques avec référence.

2.2.7 Limites

2.2.7.1 Apprentissage supervisé VS non-supervisé

Les algorithmes basés sur des CNNs rivalisent et surpassent souvent les algorithmes basés sur des modèles physiques. Toutefois, ils sont soumis à de fortes contraintes, dont celle de la base d'apprentissage. Pour la restauration, les méthodes d'apprentissage supervisées nécessitent des bases de données avec un nombre important de paires d'images incluant des images avec brouillard et la vérité terrain associée (sans brouillard). Seulement, les vérités terrain sont compliqués à acquérir et difficiles à créer dans la mesure où les conditions lumineuses doivent être proches, voire identiques aux scènes avec du brouillard. Pour remédier à ce problème, il existe des bases de données d'images de synthèse, mais les résultats obtenus avec ces bases sont difficilement généralisables aux images réelles. Très récemment, pour tenter de pallier cette difficulté, des méthodes d'apprentissages non supervisées ont été créées. L'avantage des méthodes non supervisées réside dans l'absence de paires d'images avec vérité dans la base d'apprentissage. Certaines méthodes basées sur des architectures de type *GAN* s'affranchissent de cette contrainte et promettent des résultats intéressants. Cependant, ce type de structure introduit d'autres complications puisqu'ils sont connus pour être instables et difficiles à entraîner (difficultés de convergence et d'évaluation).

2.2.7.2 Modèles traditionnels VS modèles d'apprentissages

L'essor du *deep learning* tend à faire oublier les méthodes plus traditionnelles de traitement d'image. Pour autant, les deux approches ont chacune des avantages et inconvénients. O'Mahony et al. (2020) ont récemment confronté ces deux approches. Les algorithmes de *deep learning* nécessitent une base d'apprentissage importante. Ils apparaissent plus flexibles que les algorithmes traditionnels puisque les données d'apprentissage peuvent être adaptées et le réseau peut continuellement être entraîné et raffiné. Ainsi, ils permettent d'obtenir de meilleures performances et sont plus spécifiques pour une tâche donnée. Cependant, ils ont souvent des biais et des problèmes de généralisation dus aux données présentes dans la base d'apprentissage. Dans le cas des bases de données avec du brouillard, lorsque la base d'apprentissage est exclusivement composée d'un seul type de brouillard, par exemple un brouillard constant, cela provoquera des problèmes de généralisation du réseau. Ainsi, le réseau ne fonctionnera pas correctement selon les résultats attendus sur des images avec un brouillard plus dense, alors qu'il sera efficace sur des images du même type que celles présentes dans la base d'apprentissage. Idéalement, cette base doit être composée d'images avec plusieurs densités de brouillard. Un autre problème entre également en ligne de compte et concerne la taille des images de la base d'apprentissage. Alors que ce problème ne se pose pas pour des méthodes traditionnelles, des problèmes de d'allocation mémoires apparaissent lorsque la taille des images qui composent la base d'apprentissage est trop élevée. Ce qui implique d'effectuer un pré-traitement des images comme un recadrage ou un redimensionnement.

Comme l'ont expliqué O'Mahony et al. (2020), l'approche traditionnelle nécessite de sélectionner nous-même les bonnes caractéristiques de l'image en fonction du problème à résoudre.

Cette approche peut-être particulièrement lourde d'un point de vue calculatoire et d'un point de vue de recherche de paramètres optimum. Il est néanmoins possible de s'appuyer sur une compréhension physique, par exemple, du problème. Avec le *deep learning*, on doit construire la base d'apprentissage avec des données labellisées pour que le réseau puisse s'entraîner sur ces données et tenter de déterminer les caractéristiques sous-jacentes les plus descriptives (voir schéma récapitulatif à la figure 2.22). Par exemple, un label associé à une image peut être le nom de ce qu'elle représente. Ce procédé est répandu en reconnaissance ou classification d'objets. Pour la restauration d'image, il faut alimenter la base d'apprentissage avec des paires d'images associées, composées d'une image dégradée et d'une vérité terrain. Dans le cas spécifique du brouillard, les paires seraient composées d'une image avec du brouillard et d'une image sans brouillard.

Des méthodes hybrides qui associent le *deep learning* avec une approche plus traditionnelle ont déjà été proposées, dans l'espoir des bénéficier des avantages des deux approches.

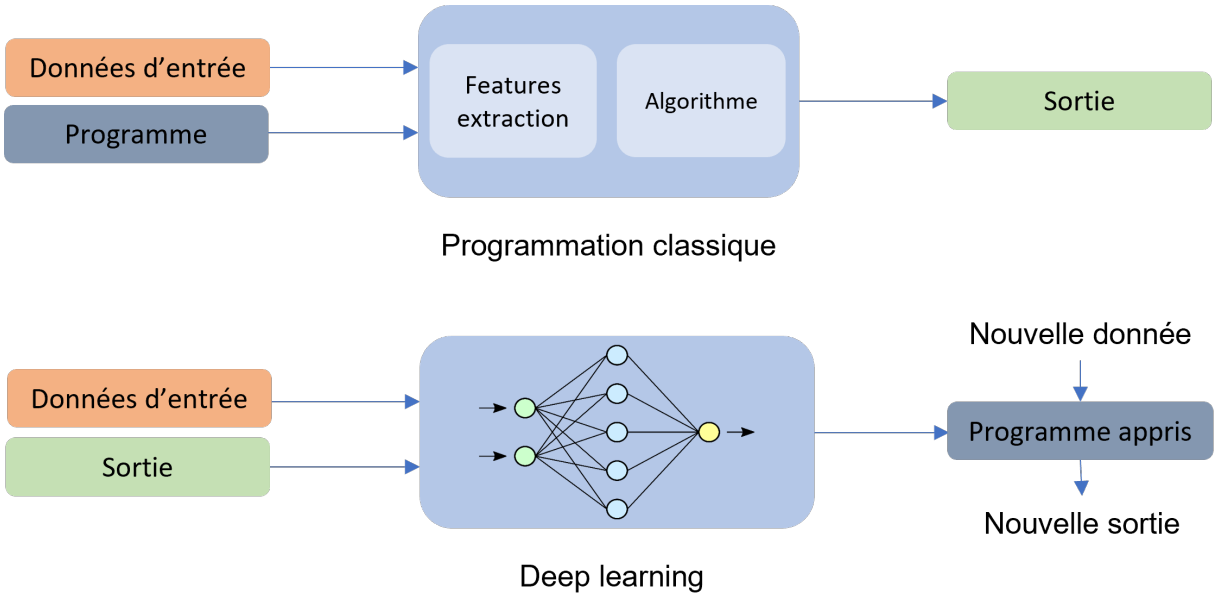


FIGURE 2.22 – Schéma illustrant la différence entre un algorithme traditionnel et un algorithme de *deep learning*

2.2.8 Discussion sur la notion d'*a priori*

Dans la majorité des travaux, il est fréquent de constater que l'état de l'art des méthodes de restauration de brouillard est scindé en deux parties. Une partie repose sur les méthodes « basées sur des *a priori* » et l'autre sur les méthodes basées sur des techniques d'apprentissage profond. La section dédiée aux méthodes avec *a priori* regroupe les méthodes dites traditionnelles, utilisant un modèle physique du brouillard, et l'autre section comporte les méthodes d'apprentissage. Cependant, certaines méthodes basées sur l'apprentissage intègrent des *a priori* déterminés par les méthodes traditionnelles comme le *DCP*.

Par ailleurs, le choix de la structure des algorithmes de *deep learning* et de la fonction de coût implique également des *a priori* implicites. De plus, les CNNs ne permettent pas d'obtenir des informations concrètes sur les *a priori* inscrits dans le réseau.

2.3 Les bases de données d'images et de vidéos avec du brouillard

2.3.1 Les bases de données d'images

Il existe diverses bases de données, de différentes tailles, particulièrement utilisées pour les approches de reconnaissance et de classification. ImageNet est actuellement la base de données la plus importante, composée de millions d'images et contenant 1000 classes différentes. Dans le domaine de la restauration, le nombre de bases de données est plus faible, en particulier les bases de données avec du brouillard. Phénomène particulièrement peu fréquent, il est difficile d'obtenir des paires d'images avec et sans brouillard dans des conditions de luminosité identiques. Pour remédier à ce problème, des bases de données de synthèses ont été élaborées. Des bases de données ont tout d'abord été créées pour l'évaluation objective des algorithmes.

FRIDA, réalisée par Tarel et al. (2010), est la première base composée d'images avec du brouillard de synthèse conçue pour les systèmes avancés d'aide à la conduite (ADAS). Elle comporte des paires de 420 scènes de route générées par ordinateur avec un brouillard et une luminosité homogène et non homogène. Cette base a été créée en prenant en compte la profondeur des images et en calculant le brouillard avec le modèle de Koschmieder. Cette base de données ne permet pas de généraliser à des scènes réalistes. Elle est trop peu fournie en images pour l'apprentissage.

(a) (b)

FIGURE 2.23 – Images de la base de données d'images FRIDA : a) image sans brouillard, b) image avec du brouillard de synthèse.

La base de données D-Hazy de Ancuti et al. (2016a) a été formée à partir des cartes de profondeurs des bases de données Middlebury (Scharstein et al. (2014)) et NYU-Depth V2 (Silberman et al. (2012)) composées de plus de 1400 images d'intérieur. Bien que ce type de scènes ne soit pas représentatif des scènes typiques de brouillard, les scènes de brouillard en intérieur se généralisent assez bien à celles en extérieur.

La base HazeRealisticDataset (HazeRD Zhang et al. (2017c)) a été composée pour l'analyse comparative des algorithmes de restauration de brouillard. Elle est composée de quinze scènes d'extérieur brumeuses différentes, prises dans cinq conditions météorologiques différentes.

Li et al. (2019a) ont proposé REalistic Single Image DEhazing (RESIDE), un ensemble de données à grande échelle pour évaluer et comparer équitablement les algorithmes de restauration de brouillard dans des images. La base d'apprentissage contient 13990 images de brouillard de synthèse, générées à l'aide de 1399 images sans brouillard à partir des base de données existantes NYU-Depth V2 et stéréo de Middlebury.

Les bases de données NTIRE (New Trend in Image Restoration and Enhancement) sont apparues en 2018. Elles sont issues de la mise en place de workshop et de challenges en res-

(a) (b)

FIGURE 2.24 – Images de la base de données D-Hazy à partir des données de NYU-Depth V2 : a) image sans brouillard, b) image avec brouillard.

(a) (b)

FIGURE 2.25 – Images de la base de données RESIDE (SOTS : brouillard de synthèse) : a) image sans brouillard, b) image avec brouillard.

tauration et amélioration d'images, par exemple la super-résolution ou encore la suppression de brouillard. À l'issue de ces challenges, les bases de données ont été rendues accessibles en ligne.

Il existe des bases de données avec du brouillard léger, créées par Ancuti et al. (2018), une avec un brouillard très dense (Ancuti et al. (2019)) et une base avec un brouillard non homogène (Ancuti et al. (2020)). Le brouillard introduit dans les différentes scènes proposées a été généré par des machines à brouillard professionnelles. Les paires d'images de scènes avec et sans brouillard ont été photographiées dans les mêmes conditions lumineuses.

(a) (b) (c)

FIGURE 2.26 – Images de la base de données NTIRE : a) image sans brouillard, b) image avec brouillard épais (base de 2019), c) image avec un brouillard non homogène (base de 2020).

TABLE 2.1 – Nombre d'images par base de données.

Datasets	**indoor**	**outdoor**
FRIDA Tarel et al. (2010)	0	420
D-Hazy Ancuti et al. (2016a)	1400	0
HazeRD Zhang et al. (2017c)	0	15
RESIDE Li et al. (2019a)	14490	72135
NTIRE18 Ancuti et al. (2018)	0	45
NTIRE19 Ancuti et al. (2019)	0	33
NTIRE20 Ancuti et al. (2020)	0	55

2.3.2 Les bases de données de vidéos

La raison principale pour laquelle il y a peu de travaux en restauration de vidéos, est qu'il n'existe que très peu de bases de données de vidéos contenant du brouillard. Ren et al. (2019) ont généré une base de synthèse contenant les clips vidéo et les cartes de profondeur correspondantes de NYU-Depth V2. Les images résultantes sont similaires aux images de la base D-Hazy illustrées à la figure 2.24. Cependant, il s'agit d'un brouillard de synthèse. Très récemment, Zhang et al. (2022) ont mis au point un système d'acquisition de séquences vidéo, appelé REVIDE. Le système se compose d'un bras robotisé, d'un appareil photo monté à l'extrémité du bras et de deux machines à brouillard. Comme le robot peut atteindre à plusieurs reprises le même emplacement avec une très haute précision, il est possible de répéter la même trajectoire afin de collecter des trames avec et sans brouillard pour réaliser des paires de vidéos. La base en contient 48, de quatre scènes différentes et avec plusieurs conditions lumineuses et densité de brouillard.

(a) (b)

FIGURE 2.27 – Images de la base de données REVIDE : a) image sans brouillard, b) image avec brouillard.

2.4 Conclusion

2.4.1 Synthèse de l'état de l'art

Dans ce chapitre, un état de l'art des méthodes de restauration d'images et de vidéos dans le brouillard, ainsi que différentes bases de données existantes ont été présentés.

2.4.1.1 Les différentes méthodes de restauration d'images dans le brouillard.

Les méthodes de restauration de la visibilité dans des images acquises par temps de brouillard peuvent être regroupées en trois catégories :

— Les méthodes basées sur des modèles physiques : les images restaurées sont obtenues par inversion de l'équation de Koschmieder.

— Les méthodes basées sur des données : les images restaurées sont obtenues directement à partir des images d'entrées par apprentissage des données.

— Les méthodes hybrides : ces méthodes résultent de la combinaison des deux méthodes précédentes, par exemple via l'apprentissage de paramètres.

TABLE 2.2 – Avantages et inconvénients des méthodes traditionnelles et des méthodes de *deep learning*.

	Modèles physiques	**Modèles basés sur des jeux de données**
Avantages	— Pas de contrainte de limite de taille des images d'entrée	— Plus performant et spécifique à une application donnée — Apprentissage implicite des caractéristiques sous-jacentes des images
Inconvénients	— « Hand-Crafted method » pouvant mener à des difficultés d'obtention des paramètres optimums	— Besoin d'une base d'apprentissage de grande taille — Nécessite un pré-traitement et redimensionnement des images — Difficulté de généralisation

Le tableau 2.2 présente les avantages et inconvénients des méthodes fondées sur des modèles physiques et sur des jeux de données. Les méthodes hybrides, citées dans la sous-section 2.2.3, bénéficient des avantages de chacun, mais également des inconvénients des modèles basés sur les données. L'apprentissage des *a priori* conduit souvent à de meilleurs performances que dans le cas où les *a priori* sont déterminés empiriquement.
Cependant, ce type de méthodes fait souvent intervenir des étapes supplémentaires d'estimation de la carte de transmission ou du voile atmosphérique, comme certaines méthodes basées exclusivement sur les données (par exemple Cai et al. (2016)). Ces étapes intermédiaires de calcul

ne sont jamais précises à cause de l'absence de vérité terrain des cartes de transmission, ce qui impacte la qualité de l'image restaurée (avec la méthode de Liu et al. (2019), par exemple).

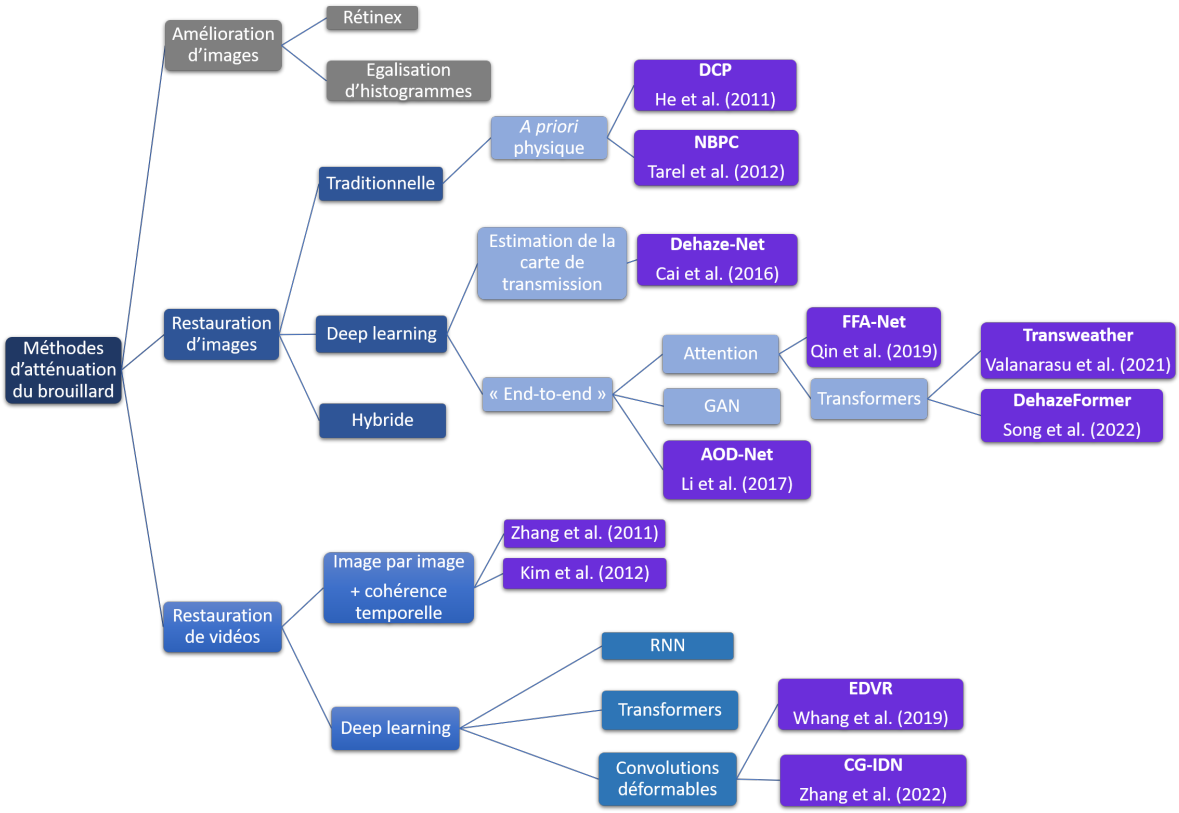


FIGURE 2.28 – Classification des méthodes d'atténuation du brouillard introduites dans l'état de l'art.

La figure 2.28 présente un schéma récapitulatif des méthodes d'atténuation du brouillard vu dans ce chapitre. Cette figure sera reprise tout au long de la thèse afin d'en faciliter la compréhension et la lecture.

2.4.1.2 L'importance de la cohérence temporelle en restauration de vidéo

En restauration de vidéos, des informations supplémentaires issues de la temporalité offrent de nouvelles perspectives par rapport à la restauration d'images statiques. Les méthodes de restauration de la visibilité par temps de brouillard dans les vidéos sont moins nombreuses que les méthodes de restauration de la visibilité dans les images. Une des principales raisons est le manque de données pour l'apprentissage de vidéos avec du brouillard. Il n'existe actuellement que deux bases de données exploitables (voir la sous-section 2.3.2). En effet, il existe beaucoup plus de bases de données pour d'autres applications de restauration dans des vidéos, comme pour la super-résolution ou le défloutage (*denoising*), menant à une quantité plus importante de travaux.

2.4.2 Verrous scientifiques et contributions

L'objectif de cette thèse est de concevoir des algorithmes de restauration d'images et de vidéos acquises en conditions météorologiques dégradées. La première partie de mes travaux consiste à réaliser un algorithme de restauration d'images statiques basés sur un modèle physique, puis, dans une seconde partie, un algorithme appliqué à la restauration de vidéos après avoir construit une base de tests composée de vidéos avec et sans brouillard. Les principales contributions peuvent être organisées en trois parties, listées ci-dessous.

2.4.2.1 Modèle physique VS *deep learning* pour la restauration d'images avec du brouillard

Il existe différentes approches pour la restauration d'images avec du brouillard. Les premières méthodes étaient basées sur un modèle physique, puis les méthodes basées sur le *deep learning* ont émergées à partir de 2016. Dans la plupart des articles, les auteurs qui s'appuient sur des approches de *deep learning*, insistent sur le fait que ces méthodes surpassent celles basées sur des modèles physiques. Le paragraphe 2.2.7.2 montre, au travers de l'article de O'Mahony et al. (2020), les avantages et les inconvénients de chaque méthode. Même si l'approche physique (dite traditionnelle) nécessite de déterminer les bonnes caractéristiques (*features*) empiriquement, l'autre approche mène souvent à des difficultés de généralisation dans le cas de la restauration d'images dans le brouillard, ainsi qu'à des problèmes liés à la taille de la base d'apprentissage.
La première méthode que j'ai proposée est une méthode de restauration d'images fondée sur le modèle physique du brouillard, avec une volonté de confrontation avec les méthodes de *deep learning*. L'algorithme doit être généralisable, c'est-à-dire qu'il doit pouvoir restaurer des images avec différents types de particules fines, de différentes couleurs et de différentes densités (brouillard de jour et de nuit, brume, poussière, pollution). De plus, cette méthode nous permet de nous affranchir de la contrainte de la taille de la base de données.

Parmi les méthodes qui reposent sur un modèle physique, le *Dark Channel Prior* a retenu notre attention. La possibilité de tester l'algorithme a permis de me rendre compte du rôle du paramètre ω dans le calcul de la restauration. Les résultats obtenus avec cette méthode ont montré que la restauration était réalisée de manière identique sur l'ensemble des images. Or, comme la densité de brouillard augmente avec la distance, il y a relativement peu de brouillard dans la zone proche de la caméra, dans la partie inférieure de l'image, entraînant une sur-restauration dans cette zone. Nous pensons que cette conséquence est en lien avec le paramètre ω, malgré le fait qu'il ait été introduit pour éviter un phénomène de sur-restauration.
Le paramètre ω du *Dark Channel Prior* va être réinterprété selon nos propres hypothèses. En effet, la plupart des méthodes restaurent les images dans leur globalité, alors que la densité du brouillard augmente avec la distance. Dans la zone proche de la caméra, il n'y a pas de brouillard. Le fait de restaurer une zone sans brouillard engendre une sur-restauration et donc un résultat irréaliste. À partir d'une analyse de cet algorithme, un nouvel *a priori* est proposé afin d'obtenir une meilleure atténuation du brouillard.

2.4.2.2 La contrainte des données en *deep learning*

Le manque de base de données d'images et de vidéos dans le brouillard avec une vérité terrain associée, est une réelle contrainte dans la réalisation d'algorithmes de *deep learning* pour la restauration de la visibilité dans le brouillard, en particulier dans des vidéos. La section 2.3 liste

les bases de données avec du brouillard existantes. S'il y a actuellement plusieurs types de bases de données d'images (de synthèse et réaliste), jusqu'à très récemment, il n'existait qu'une seule base de données de vidéos avec des scènes d'intérieur sur lequel était ajouté du brouillard de synthèse. Zhang et al. (2020b) ont apporté une solution avec une base de données de vidéos du nom de REVIDE, avec du brouillard ajouté à l'aide d'une machine à brouillard. Cette base contient assez de données pour être utilisée comme base d'apprentissage.
Finalement, j'ai réalisé une base de données de test grâce à la mise en place d'un dispositif de chambre à brouillard et l'utilisation d'une machine à brouillard. Une telle machine permet d'obtenir des séquences d'image avec un brouillard réaliste et de pouvoir contrôler la densité dans la scène. Cette base de test permet également de s'assurer de la qualité de généralisation de tous les algorithmes de restauration.

2.4.2.3 L'utilisation d'algorithme dans le contexte des ADAS

Pour répondre au besoin de l'atténuation du brouillard dans des images et vidéos dans le contexte des ADAS, il serait utile de réaliser un algorithme de restauration de vidéos. En effet, les ADAS mettent en œuvre des algorithmes de vision en temps réel, en raison de la mobilité et de la nécessité d'être réactif aux changements dans l'environnement (pour la détection de panneaux de circulation routière, de piétons ou encore de marquages au sol, par exemple). L'utilisation d'algorithmes de restauration d'images isolées, qui traitent donc les données trames par trames, fournit souvent des résultats avec des artefacts visuels (scintillement, décalages de couleur). Ces scintillements peuvent conduire à des difficultés d'analyse de l'environnement, aussi bien pour les capteurs que pour le conducteur. Exploiter l'information temporelle des vidéos est nécessaire pour obtenir un rendu fluide sans scintillement. Afin de diminuer la présence de ces artefacts et d'améliorer la qualité de la restauration de la visibilité, j'ai réalisé un algorithme basé *deep learning* pour la restauration de vidéos. Les performances de l'algorithme font l'objet d'une évaluation qualitative et quantitative.
Afin d'exploiter la redondance temporelle entre trames adjacentes, il est possible d'utiliser des réseaux de neurones récurrents, tels que les RNN et les LSTM. Il existe cependant quelques limites à leurs utilisations, comme indiqué au paragraphe 2.2.2.3. De nombreuses méthodes ont émergé en restauration de vidéos. Les méthodes qui s'appuient sur les architectures *Transformers* semblent se démarquer des autres architectures depuis très récemment, malgré une utilisation qui peut être très coûteuse en calcul lors de l'étape d'apprentissage. L'architecture de l'approche proposée sera un auto-encodeur constituée de modules *Transformers* multi-dimensions pour assurer la cohérence temporelle, et de blocs CNNs pour capturer l'information spatiale. La méthode devra également ne pas être trop coûteuse en mémoire et temps de calcul, l'objectif étant de se rapprocher le plus possible du temps réel. Une évaluation de la qualité de la restauration vidéo sera également réalisée afin de s'assurer du bénéfice du traitement vidéo dans le cas des ADAS et de la qualité de la restauration.

Chapitre 3

Restauration d'images avec du brouillard
Atténuation du voile atmosphérique à l'aide de nouveaux *a priori* pour une meilleure généralisabilité

Ce chapitre présente une méthode de restauration d'images avec du brouillard basée sur un modèle physique du brouillard.

3.1 Introduction

La restauration d'images est un problème bien connu dans les applications de vision par ordinateur et de photographie numérique, en particulier lorsque les conditions météorologiques sont dégradées par le brouillard, la brume, la pluie et/ou la neige. Ces mauvaises conditions météorologiques provoquent des dégradations dans les images telles que la perte de contraste et le décalage des couleurs, contribuant à la réduction de la visibilité de la scène. En l'absence de visibilité, les systèmes automatisés basés sur la segmentation d'images (Tremblay et al. (2020)) et la détection d'objets peuvent être contre-performants. Cela nécessite donc de réaliser un pré-traitement des images pour restaurer la visibilité (Hautière et al. (2007)). Le faible contraste des images de brouillard est dû à la présence d'un voile atmosphérique constitué de gouttelettes en suspension. Pour la pluie et la neige, cela est dû à la présence de gouttes ou de flocons en suspension. Afin d'atténuer, voire de supprimer les effets du brouillard, une méthode de restauration de la visibilité dans des images a été mise en œuvre en réinterprétant un paramètre du *Dark Channel*, une méthode décrite dans le chapitre 2. Pour rappel (voir paragraphe 2.2.1.1), la carte de transmittance a pour équation :

$$t(x) = 1 - \omega \min_{c}\left(\min_{y \in \Omega(x)}\left(\frac{I^c(y)}{A^c}\right)\right) \tag{3.1}$$

Le paramètre ω, introduit dans l'équation de la carte de transmittance pour atténuer le phénomène de sur-restauration peut être considéré comme un *a priori* à part entière. La figure 3.1 montre que lorsque le paramètre ω diminue, l'image est uniformément plus claire après avoir appliqué la méthode de restauration. Lorsque $\omega = 0.95$, valeur proposée par He et al. (2011), la partie restaurée de la route est très assombrie, tandis qu'avec $\omega = 0.75$, cette même partie

restaurée est visuellement plus proche de celle de la vérité terrain. Il reste cependant davantage de brouillard résiduel près de l'horizon.

TABLE 3.1 – Évaluation sur une cinquantaine d'images de chaque base de données avec les métriques SSIM et PSNR de l'algorithme *DCP* pour deux valeurs de ω.

Méthodes	FRIDA	SOTS	NTIRE20
DCP avec $\omega = 0.95$	0.70/**12.26**	0.89/18.91	0.44/12.77
DCP avec $\omega = 0.75$	**0.75**/11.40	**0.91/22.36**	**0.47/13.15**

FIGURE 3.1 – Résultats de l'algorithme *DCP* avec différentes valeurs de ω sur une image de la base FRIDA : a) image avec brouillard, b) vérité terrain, c) image restaurée avec $\omega = 0.95$, d) image restaurée avec $\omega = 0.75$

Notre interprétation du paramètre ω est la suivante : le terme $\min_c(\min_{y\in\Omega(x)}(I^c(y)))$ dans l'équation de la transmittance (3.1) permet d'obtenir une première estimation du voile atmosphérique. Pour rappel, cette estimation est basée sur les hypothèses que le brouillard est blanc et localement lisse. Le résultat est un mélange du voile atmosphérique réel et de la luminance des objets perçus par la caméra que nous allons appeler le pré-voile. Le pourcentage de ce pré-voile, correspondant au voile atmosphérique réel, est inconnu et supposé constant sur toute l'image, de valeur ω. Par conséquent, le paramètre ω est le pourcentage du voile atmosphérique dans la carte du pré-voile. La validité de cet *a priori* est testée à la sous-section 3.2.1. On considère également l'hypothèse suivante : le brouillard apparaît plus dense près de l'horizon que près de la caméra. Le traitement de restauration dépend donc de la distance des objets dans la scène. Ainsi, le brouillard doit être plus atténué près de l'horizon, pour un rendu plus naturel.

Afin de répondre à ces problématiques et de prendre en compte nos hypothèses, nous avons réalisé des observations empiriques sur une base de données composée d'images de synthèse. La fonction la plus pertinente retenue est celle de Naka-Rushton (tone-mapping), utilisée pour moduler le voile atmosphérique. Les paramètres de cette fonction sont définis à partir des caractéristiques de l'image en entrée. L'utilisation de cette fonction empêche un traitement de restauration excessif et préserve ainsi la restitution du ciel des artefacts et du bruit.

L'algorithme, appelé *MFP* (en anglais, *Modulation Function prior*), basé sur cette fonction, se généralise à différents types de brouillard, de particules en suspension dans l'air et de conditions d'éclairage. La méthode proposée est étendue aux images nocturnes et sous-marines en calculant le voile atmosphérique sur chaque canal de couleur. Des évaluations qualitatives et quantitatives montrent l'intérêt de l'algorithme proposé. L'évaluation quantitative montre l'efficacité de l'algorithme sur quatre bases de données avec différents types de brouillard, démontrant ainsi la capacité de généralisation de l'algorithme proposé et sa capacité à égaler voire surpasser certains algorithmes basés sur des méthodes de *deep learning*.

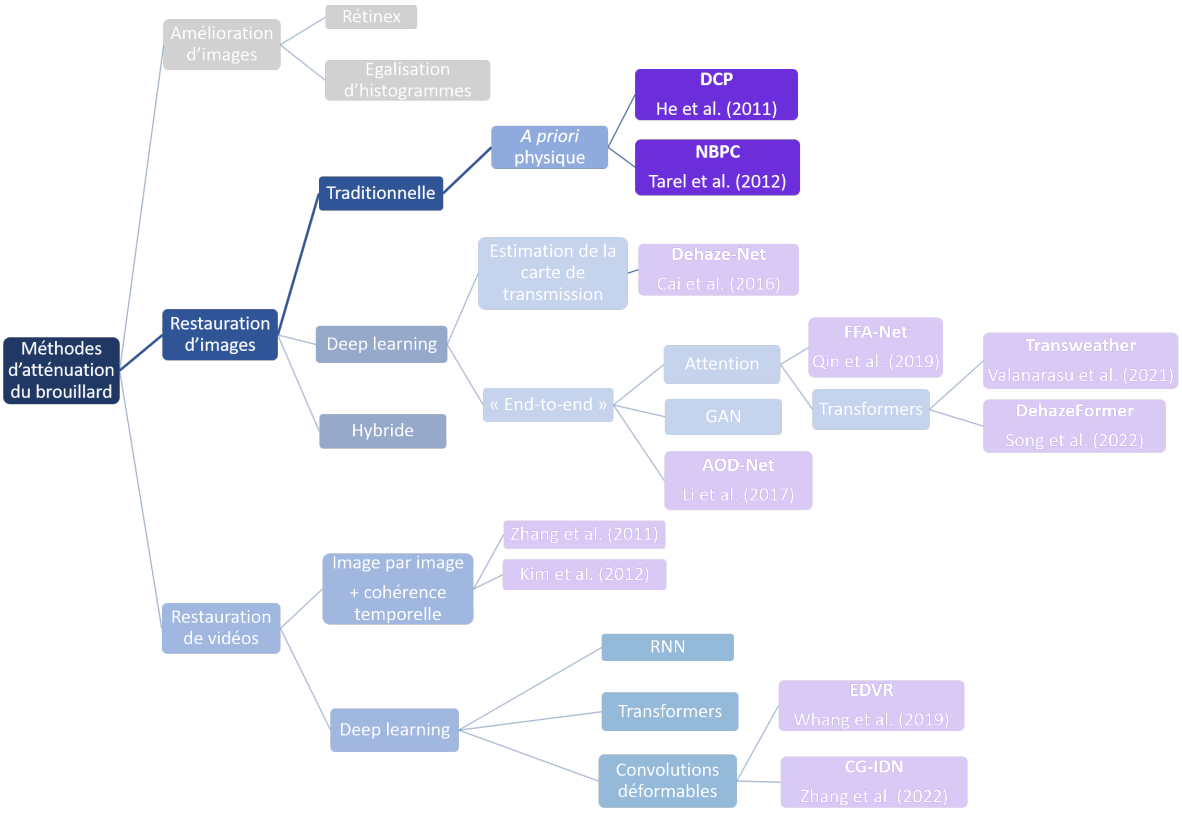


FIGURE 3.2 – Diagramme récapitulatif de l'état de l'art.

La figure 3.2 permet de situer les travaux présentés dans ce chapitre par rapport aux travaux présentés dans le chapitre sur l'état de l'art.

3.2 Description de la méthode

3.2.1 ω est-il un a priori valide ?

Le voile atmosphérique est généralement obtenu par la méthode *DCP* en appliquant le paramètre ω. Pour tester la validité de cet a priori, cela revient à étudier la relation entre l'intensité des pixels dans les images du voile réel et du pré-voile. Cependant, il n'est pas possible de connaître le voile réel d'images acquises dans le brouillard dans des conditions réelles (il n'existe pas de vérité terrain). Il est néanmoins possible d'obtenir le voile réel des images avec du brouillard de synthèse. Nous avons utilisé, pour cela, la base de données de synthèse FRIDA (Tarel et al. (2012)). Comme les images avec du brouillard sont construites à partir de cartes de profondeur, il est possible d'obtenir le voile réel. Ainsi, la carte du voile atmosphérique est calculée pour chaque image de la base de données.

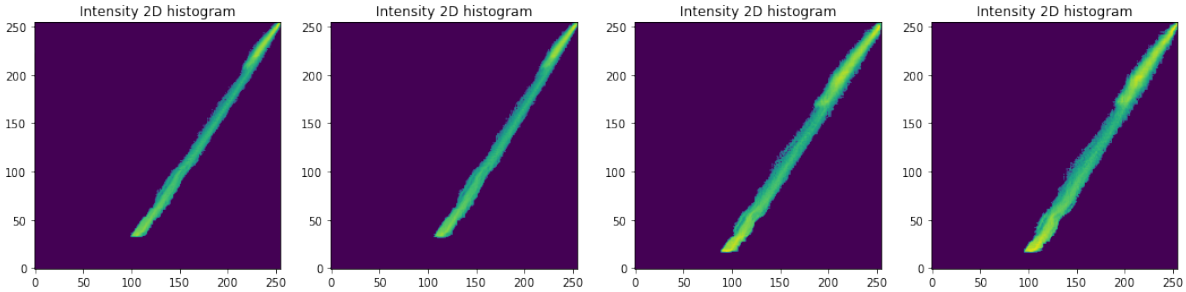


FIGURE 3.3 – Lien entre les intensités des pixels du pré-voile et du voile sur une moyenne de cinquante images de la base de données FRIDA. De gauche à droite : brouillard et éclairage homogènes, brouillard homogène et éclairage hétérogène, brouillard hétérogène et éclairage homogène, brouillard et illumination hétérogènes.

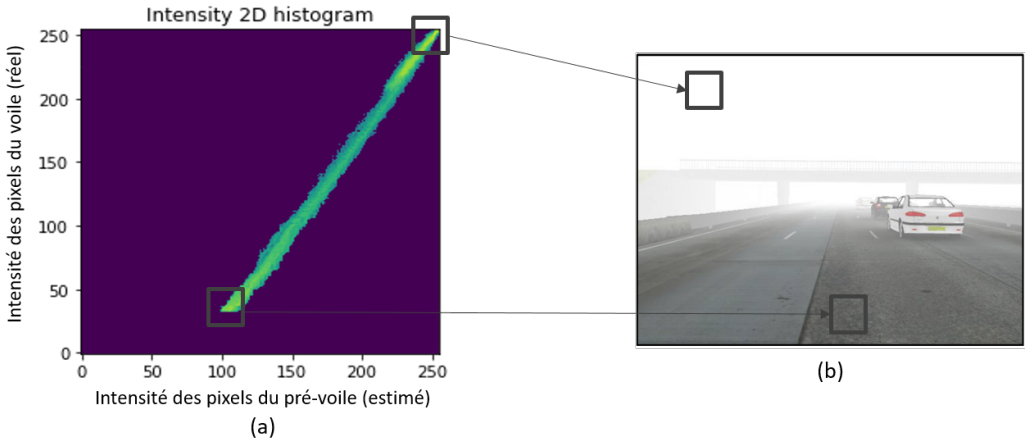


FIGURE 3.4 – Lien entre l'intensité des pixels du voile réel et l'intensité des pixels du pré-voile : (a) histogramme illustrant le lien entre les intensités des pixels du voile et du pré-voile (moyenne sur cinquante images de la base de données FRIDA), (b) image d'entrée avec du brouillard.

La figure 3.3 montre que la relation entre l'intensité des pixels de brouillard et l'intensité des pixels du voile associé est approximativement affine. Le voile atmosphérique a une intensité élevée dans le ciel et une faible intensité au sol, au niveau de la caméra. Si ce lien est de forme affine, il ne serait pas correct de le modéliser qu'avec le seul paramètre ω. Une fonction affine serait plus adéquate. La figure 3.4(a) montre l'histogramme obtenu à partir de cinquante images avec du brouillard de synthèse, avec l'intensité des pixels de l'image du pré-voile sur l'axe horizontal et l'intensité des pixels de l'image du voile atmosphérique de la vérité terrain sur l'axe vertical.

3.2.2 Un nouvel a priori : une fonction de modulation

Afin d'éviter une sur-restauration dans des zones de l'image où il n'y a pas de brouillard (en général en bas de l'image), tout en s'assurant que la restauration soit maximale dans des zones de l'image où le brouillard est présent (en haut de l'image), une fonction de modulation f est nécessaire pour estimer le voile atmosphérique à partir du pré-voile. Afin d'éviter les artefacts visuels dans l'image restaurée, la fonction choisie doit être lisse.

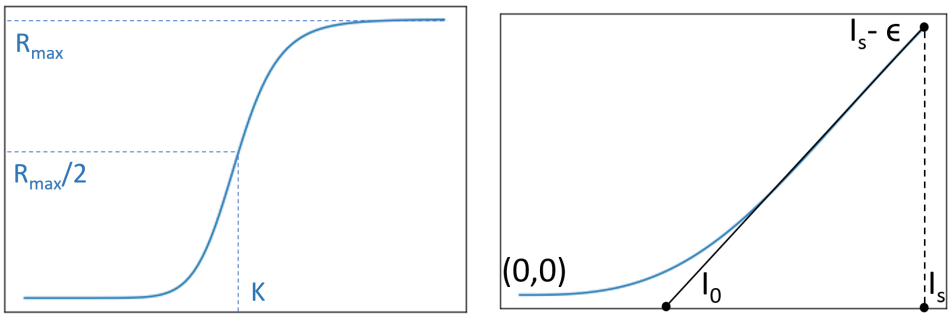


FIGURE 3.5 – À gauche : la fonction de Naka-Rushton avec ses paramètres R_{max}, K et n. À droite : notre fonction de modulation avec les paramètres I_s, I_0 et ε. En abscisse : Intensité des pixels du pré-voile.

À partir de l'histogramme de la figure 3.4(a), les contraintes suivantes sont proposées pour le choix d'une fonction f appropriée :

1. I_s est l'intensité de la région (du ciel) la plus claire. Pour éviter des valeurs trop sombres dans les zones correspondantes, $f(I_s)$ doit être légèrement inférieure à I_s. On introduit donc un paramètre ε tel que $f(I_s) = I_s - \varepsilon$. I_0 est l'intensité de la région (du ciel) la plus sombre.
2. La fonction f doit être linéaire sur une large plage d'intensités des pixels. Cette plage est notée $[I_0, I_s]$. Nous introduisons ici la pente a de f à I_s, soit $f'(I_s) = a$.
3. La fonction est proche de zéro sur la plage d'intensité $[0, I_0]$, i.e. pour les intensités près de la caméra où le brouillard n'est pas visible.
4. La fonction ne doit pas être négative.

Parmi les différentes fonctions que nous avons testées, la fonction proposée par Naka and Rushton (1966) était la plus simple à paramétrer. Cette fonction a été introduite pour la première fois afin de décrire la réponse biologique d'un neurone, et a ensuite été utilisée en infographie pour apporter une solution au problème du tone-mapping. Elle est définie comme :

$$R(x) = R_{max}\frac{x^n}{x^n + K^n} \tag{3.2}$$

où R_{max} est sa borne supérieure, K est la position horizontale du point d'inflexion et n est lié à la pente au point d'inflexion (voir figure 3.5). La forme de la première partie de la courbe de l'image de gauche de la figure 3.5 correspond à nos attentes, comme le montre l'image de droite de la figure 3.5. Cette première partie nous permet donc de définir la fonction de modulation du voile. Le point d'inflexion de coordonnées $(K, R_{max}/2)$ doit correspondre à la fonction de modulation f pour $I = I_s$.

3.2.3 Les paramètres de la fonction Naka-Rushton

R_{max}, K et n sont les paramètres de la fonction Naka-Rushton, tandis que les paramètres de la fonction de modulation f sont I_0, I_s et ε. Dans la section précédente, K était défini sur I_s. Suite à la contrainte 4, $f(I_s)$ est fixé à $I_s - \varepsilon$. Ainsi, $R_{max} = 2(I_s - \varepsilon)$. La pente en I_s vaut a. Cette pente dans la fonction Naka-Rushton étant $nR_{max}/4K$, nous avons $n = \frac{2I_s a}{I_s - \varepsilon}$ et $a = \frac{I_s}{I_s - I_0}$. Finalement, la fonction de modulation proposée f est :

$$f(x) = f_0\frac{x^n}{x^n + k^n} \tag{3.3}$$

où $f_0 = 2(I_s - \varepsilon)$, $k = I_s$, $n = \frac{2I_s a}{I_s - \varepsilon}$ et $a = \frac{I_s}{I_s - I_0}$. Cette fonction de modulation n'a que trois paramètres : I_0, I_s et ε. Les deux premiers paramètres peuvent être calculés à partir de l'image en entrée tandis que le dernier doit être fixé, et nous proposons d'utiliser $\varepsilon = 1\%$.

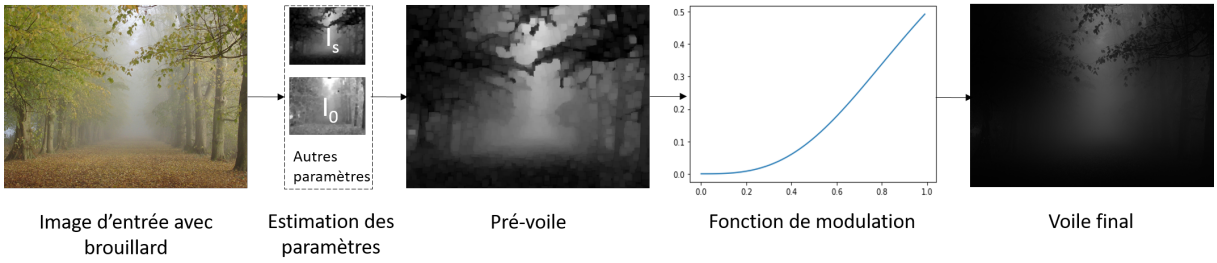
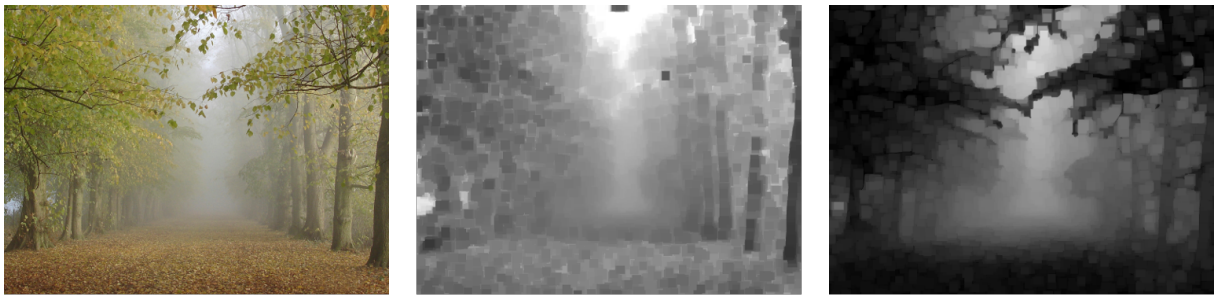


FIGURE 3.6 – Estimation du voile atmosphérique à partir du pré-voile en utilisant la fonction de Naka-Rushton comme fonction de modulation.

Sur la figure 3.6, I_s représente l'intensité du ciel et I_0 l'intensité du sol près de la caméra. Nous avons étudié la meilleure façon d'estimer I_0 et I_s : prendre le maximum des intensités des images pour I_s et le minimum pour I_0 rend l'algorithme trop sensible au bruit. Par conséquent, I_0 et I_s sont calculés en prenant, respectivement, le minimum et le maximum des intensités de l'image en entrée avec du brouillard après avoir appliqué, respectivement, un filtre morphologique de fermeture et d'ouverture (voir figure 3.7). En effet, l'ouverture permet de conserver les parties claires de l'image égales à l'élément structurant (dans notre cas un carré de taille 20×20 pour une image $X \times Y$). Sur la figure 3.7c, on remarque que les parties de l'images plus claires correspondent au ciel et au brouillard. Au contraire, la fermeture conserve les parties sombres égales à l'élément structurant. Les parties de l'image plus sombres correspondent aux arbres en premier plan et à la zone proche de la caméra.



(a) (b) (c)

FIGURE 3.7 – Estimation de I_s et I_0 à partir des images filtrées : (a) image d'entrée, (b) image filtrée avec une fermeture, (c) image filtrée avec une ouverture.

Cependant, l'image du pré-voile estimée introduit des effets de blocs au niveau des contours des structures, à cause de l'utilisation du filtrage. Dans la sous-section suivante, nous justifions l'utilisation d'une méthode de raffinement pour corriger ce problème.

3.2.4 Le raffinement

Les effets de blocs introduits par filtrage dans le calcul de l'image du pré-voile provoque l'apparition de halos autour des contours dans les images restaurées, visibles à la figure 3.8b. L'utilisation d'une méthode de raffinement permet d'atténuer ces effets de bloc. Aussi appelés artefacts, ces effets ont dans un premier temps été constatés avec les travaux de He et al. (2010). Les auteurs ont présenté les méthodes de Soft Matting (très coûteuse en temps de calcul) puis, plus tard, de filtrage guidé (He et al. (2011)) dans le but de raffiner les cartes de transmission pour supprimer ces artefacts.

(a) (b) (c)

FIGURE 3.8 – Image restaurée avec la méthode *DCP*, avec et sans méthode de raffinement : a) Image avec du brouillard, b) image restaurée sans raffinement, c) image restaurée et raffinée avec la méthode de filtrage guidé.

Le filtre guidé calcule une sortie de filtrage q en considérant le contenu d'une image guide I. Dans notre contexte, l'image guide est l'image d'entrée p, et peut être utilisée comme opérateur de lissage préservant les contours, comme le filtre bilatéral (Tomasi and Manduchi (1998)), et le filtre bilatéral guidé (Caraffa et al. (2015)). La sortie de filtrage au pixel i est exprimée comme une moyenne pondérée par la formule suivante :

$$q_i = \sum_j W_{ij}(I) p_j \tag{3.4}$$

où i et j sont les indices de pixels et W_{ij} correspond au noyau de filtrage fonction de l'image guide I.
La figure 3.8c présente une image restaurée raffinée avec le filtre guidé. Les blocs au niveau des contours des feuilles au premier plan ont été lissés, et la zone où se trouve le bâtiment semble moins contrastée. On constate que l'apparence globale de l'image raffinée offre une meilleure qualité visuelle. Pour vérifier l'impact du raffinement sur les contours, nous présentons à la figure 3.9, des images filtrées à l'aide d'un filtre de Canny sur des images de la base de données FRIDA. La fonction Canny de la bibliothèque OpenCV a été utilisée avec des valeurs de seuil minimales et maximales fixées respectivement à 120 et 180. Les images RGB correspondantes sont référencées à la figure 3.1.

La figure 3.9b présente les contours de l'image raffinée avec le filtre guidé et la figure 3.9c présente les contours de l'image sans avoir appliqué de méthode de raffinement. En comparaison avec les contours de la vérité terrain, ceux de l'image non raffinée en conserve davantage par rapport à l'image raffinée (voir les contours surlignés en vert dans l'image de droite). Cependant, les effets de blocs, introduits par le filtrage morphologique, introduisent des "faux contours" dans l'image non raffinée (voir les contours surlignés en rouge dans l'image de droite). Le filtre guidé lisse certains contours et permet d'obtenir un rendu visuel de meilleure qualité. Sans raffinement, l'apparition de faux contours peut perturber ou fausser l'interprétation de ces images. Dans la suite du chapitre, le filtre guidé comme méthode de raffinement est admis.

3.2.5 Application aux canaux de couleurs

La loi de Koschmieder s'applique lorsque le brouillard et l'éclairage sont homogènes. Lorsque la densité et/ou la lumière sont hétérogènes, comme lors des nuits brumeuses, la loi de Koschmieder n'est plus valable. En effet, la nuit, les halos sont causés par la diffusion du brouillard par des sources de lumière artificielle. La présence de ces halos implique que, bien qu'il soit uniforme en densité, le voile atmosphérique n'est pas uniformément éclairé. Les images sous-marines souffrent également de grandes variations d'éclairage dues à l'absorption de la lumière par les eaux profondes.

De nombreux algorithmes ont été proposés pour la suppression du brouillard de nuit.
Li et al. (2015) ont proposé un algorithme pour supprimer les effets de halos en les séparant du reste de l'image. D'après Ancuti et al. (2016b), en partant du principe que la suppression du brouillard est un processus local, ils proposent une approche de fusion multi-échelles pour la restauration des images de nuit. Zhang et al. (2017b) ont mis en œuvre une méthode consistant à calculer la réflectance maximum d'une image (MRP) pour l'estimation de l'éclairage ambiant. Très récemment, Lou et al. (2020) ont proposé une nouvelle technique de correction des couleurs basée sur la technique du MRP, et ils ont utilisé une corrélation inverse entre la transmittance et la densité du brouillard pour estimer la carte de transmission.

L'hypothèse du brouillard blanc n'est plus valable pour des images nocturnes et sous-marines. Cependant, des résultats intéressants peuvent être obtenus malgré les limites théoriques de la loi de Koschmieder. En effet, les halos de brouillard colorés sont généralement dus à la couleur des sources de lumière artificielle, telles que les lampadaires ou les phares des voitures (voir la figure 3.10). Dans les scènes sous-marines, l'absorption de la lumière varie en fonction de la longueur d'onde, et dépend de la nature des particules en suspension dans l'eau.

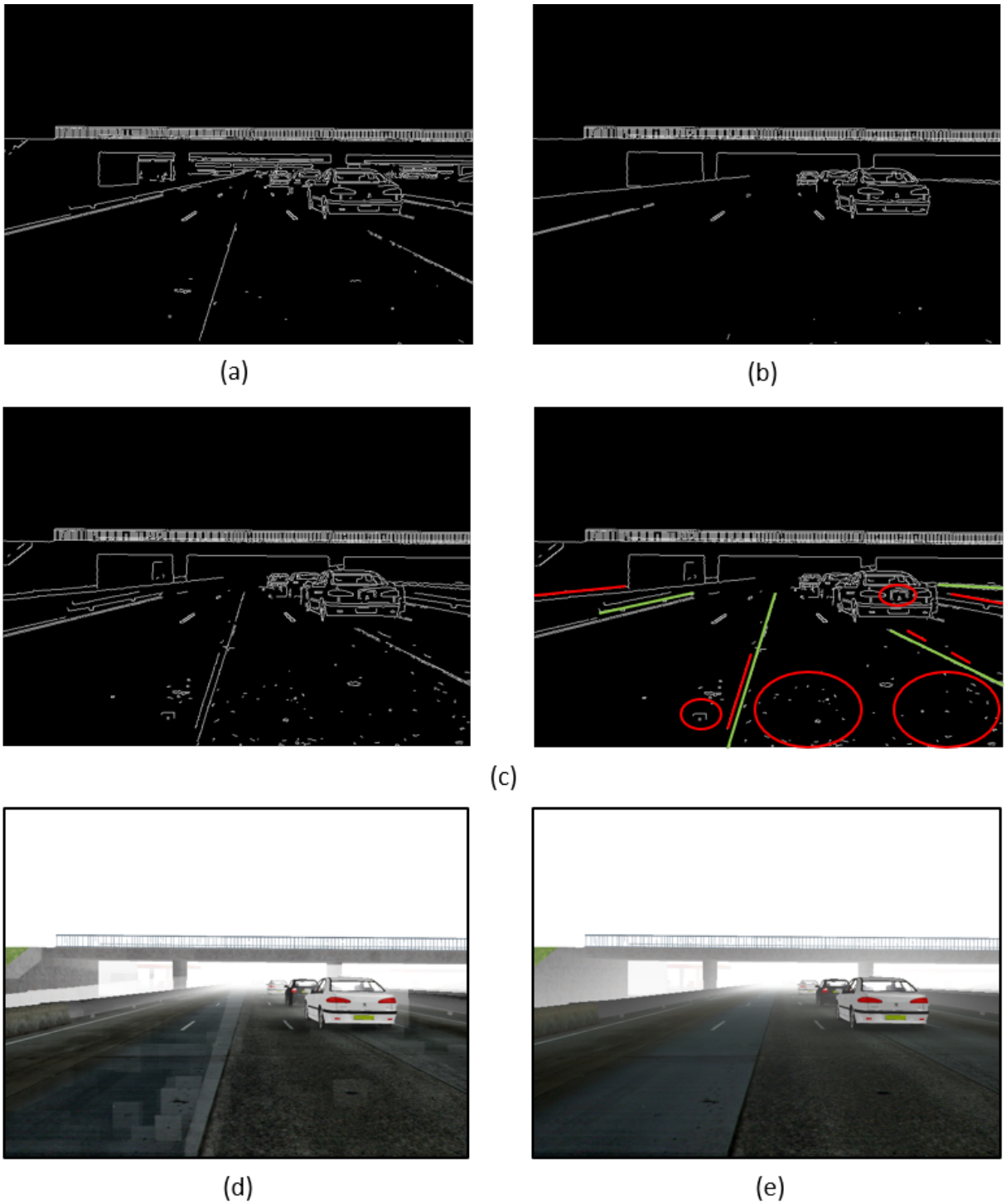


FIGURE 3.9 – Gradients des images calculés avec le filtre de Canny : a) Gradient de la vérité terrain, b) Gradient de l'image restaurée raffinée avec le filtre guidé, c) Gradient de l'image restaurée sans raffinement. Sur l'image de droite, les « faux coutours » sont en rouge et les contours disparus avec le filtre guidé sont en vert, d) image restaurée sans raffinement, e) image restaurée avec le filtrage guidé.

Habituellement, il y a un décalage de couleur important vers le bleu ou le vert. Les limites de l'hypothèse du brouillard blanc peuvent également être observées dans d'autres situations telles

FIGURE 3.10 – Image avec du brouillard de nuit

que la pluie à grande distance, la fumée et la poussière.

Afin de mieux traiter le voile atmosphérique coloré, nous proposons une méthode simple qui consiste à appliquer l'algorithme sur chacun des canaux de couleurs. Cette application n'est possible que parce que la méthode de suppression du voile atmosphérique proposée est capable de traiter des images en niveaux de gris. Traiter les canaux de couleurs séparément permet d'obtenir trois estimations du paramètre I_s, et ainsi, de déduire la couleur du voile.

La figure 3.11 présente les valeurs des pixels sur chacun des trois canaux de couleur d'une image avec un brouillard blanc. Les trois valeurs des pixels au centre des carrés noirs sont très proches les unes des autres. Par conséquent, il est possible d'estimer le pré-voile à partir du canal de couleur dont l'intensité moyenne est minimale (voir *DCP*). Comme le montre l'histogramme à la figure 3.11b, l'intensité des pixels sur chaque canal est approximativement identique. En revanche, la figure 3.11c montre que les valeurs des pixels au milieu des carrés noirs sont différentes selon les canaux et dépendent de la zone de l'image. Par exemple, les valeurs RVB sont R = 223, V = 165, et B = 84 dans le brouillard jaune, et R = 9, V = 75 et B = 109 dans le brouillard bleu. Ces différences d'intensité entre les trois canaux RVB sont illustrées sur la figure 3.11d. Ainsi, la technique d'estimation du pré-voile pour le brouillard blanc diurne ne fonctionne pas avec le brouillard coloré de nuit : un pré-voile coloré doit être estimé (voir la figure 3.12c).

Afin de traiter le voile atmosphérique coloré, nous proposons une méthode simple : traiter chaque canal de couleur séparément avec l'algorithme. La figure 3.12 montre des exemples de voile atmosphérique estimé à partir du canal de couleur dont l'intensité moyenne est minimale sur l'ensemble de l'image (figure 3.12 a, b), ainsi qu'un exemple de voile atmosphérique estimé sur chaque canal de couleur (figure 3.12c). Ceci n'est possible que parce que la méthode de suppression du voile atmosphérique proposée est capable de traiter des images en niveaux de gris grâce à l'utilisation de la fonction de modulation. En traitant chaque canal de couleur séparément, I_s est estimé sur chaque canal et permet de déduire la couleur du voile.

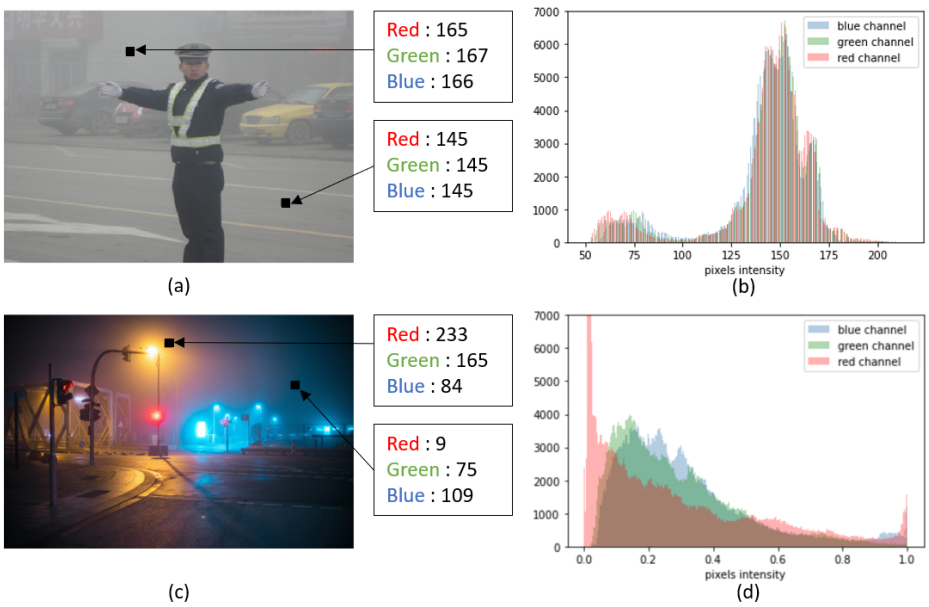


FIGURE 3.11 – Comparaison des valeurs RVB dans les images : (a) image avec du brouillard blanc de jour provenant de la base de données RESIDE (Li et al. (2019a)), (b) histogramme de l'image (a), (c) image avec un brouillard coloré (avec la permission de Broyer (2017), copyright 2021 Mark Broyer) , (d) histogramme de l'image (c). Cette figure montre les valeurs RVB des pixels au centre des carrés noirs. L'intensité des pixels est comprise entre 0 et 255.

FIGURE 3.12 – Le voile atmosphérique avant l'étape de raffinement : (**a**) estimation du voile atmosphérique d'une image de brouillard blanc, (**b**) estimation du voile atmosphérique d'une image brumeuse colorée à partir du canal couleur avec l'intensité minimale, (**c**) estimation du voile atmosphérique d'une image brumeuse colorée de chaque canal de couleur (adapté avec la permission de Broyer (2017). Copyright 2021 Mark Broyer).

3.3 Schéma global de l'algorithme

L'algorithme proposé nécessite un *a priori* sur la nature de la scène observée. Deux catégories sont considérées :

— Le brouillard blanc de jour : le pré-voile est calculé à partir du canal de couleur qui a l'intensité la plus faible. Le brouillard est supposé homogène et peut varier lentement en densité d'un pixel à l'autre.

— Le brouillard coloré : le voile ne peut pas être supposé complètement blanc. Ainsi, le voile atmosphérique est calculé sur chaque canal de couleur. La densité des particules en suspension peut varier lentement d'un pixel à l'autre.

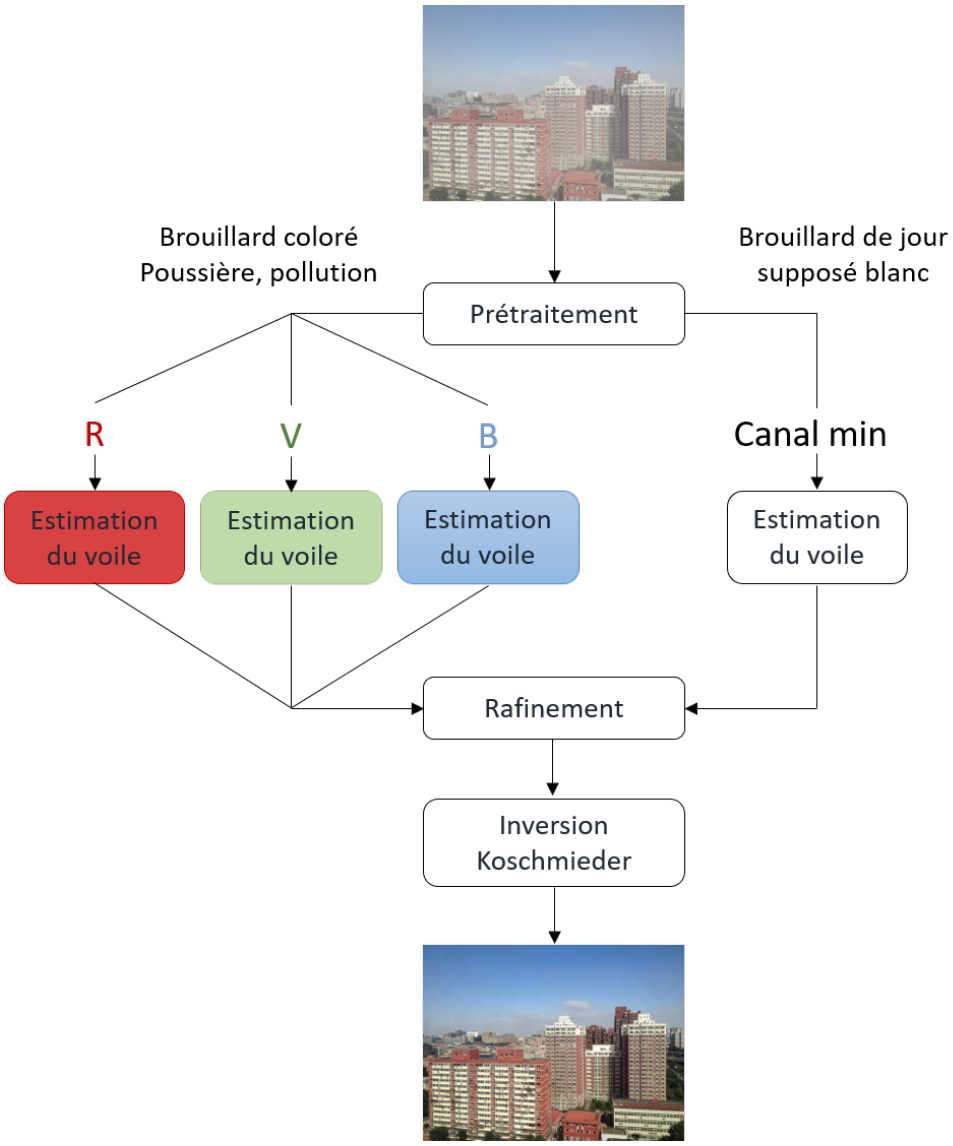


FIGURE 3.13 – Organigramme de l'algorithme *MFP*.

L'organigramme de l'algorithme *MFP* est proposé à la figure 3.13. À partir de l'étape de calcul du pré-voile, le voile atmosphérique est obtenu en utilisant la fonction de Naka-Rushton. Les paramètres de cette fonction sont calculés à partir des caractéristiques de l'image en entrée. Ensuite, un raffinement du voile atmosphérique est effectué à l'aide d'un filtrage guidé par l'image d'entrée. Nous avons sélectionné le filtre guidé pour cette étape. La dernière étape consiste à calculer l'image restaurée en inversant l'équation de la loi de Koschmieder sur l'image en entrée comme suit :

$$J = \frac{(I - V)I_s}{I_s - V} \tag{3.5}$$

où V correspond au voile estimé.

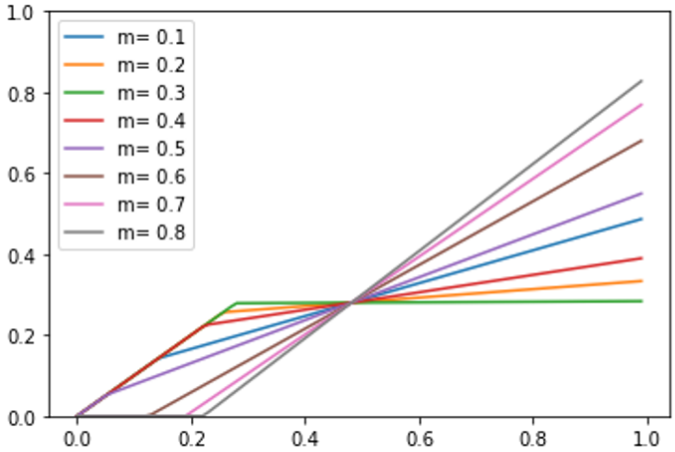
3.4 Extension de l'algorithme à des images avec du brouillard constant

Des analyses qui s'appuient sur différentes bases de données avec des images de brouillard ont montrés que l'algorithme proposé est moins efficace sur des images où le voile est spatialement uniforme (qui ne dépend pas de la profondeur de l'image). On retrouve ce type de brouillard sur des images satellitaires. Notre algorithme a donc été modifié pour pouvoir traiter les types de voile dépendant, ou non, de la profondeur de la scène, ce qui a pour conséquence de rendre la méthode plus générique. Dans le cas où le voile est uniforme spatialement, la méthode de restauration doit être basée sur une fonction constante puisque le brouillard à une profondeur constante. Cette fonction constante est alors définie comme $g(x) = I_0$. I_0 correspond à la valeur de pixel minimale du pré-voile, soit l'intensité des zones les plus sombres de l'image filtrée. Dans le cas d'un brouillard réaliste, cette valeur doit être proche de zéro, puisqu'elle représente la zone proche de la caméra sans brouillard.

Plusieurs méthodes ont été considérées pour le traitement des images avec du brouillard spatialement uniforme. La méthode d'interpolation entre deux fonctions f et g a été retenue. Deux fonctions ont été testées. Pour f : la fonction de Naka-Rushton de l'équation (3.3), nommée f_1 ; et la fonction qui s'accorde avec la partie affine de la fonction de Naka-Rushton : $f_2(x) = \frac{x-I_0}{1-I_0}$, qui est proche de la forme de l'histogramme représenté à la figure 3.4. La fonction d'interpolation est la suivante :

$$k_{m,p,i}(x) = \frac{I_0^p g(x) + (m - I_0)^p f_i(x)}{I_0^p + (m - I_0)^p} \tag{3.6}$$

Les paramètres m et p permettent d'ajuster la pondération des deux fonctions. Des valeurs élevées de p conduisent à un basculement brutal entre f_i et g ;



m	SSIM	PSNR
0,1	0.91	22.40
0,2	0,92	23,17
0,3	**0,92**	**23,31**
0,4	0,92	23,13
0,5	0,91	22,22
0,6	0,89	20.97
0,7	0,88	19,90
0,8	0,86	19,11

FIGURE 3.14 – Test de la fonction d'interpolation afin de déterminer le paramètre m optimal pour une valeur de p=2 sur la base de données SOTS.

La figure 3.14 présente un exemple de test préliminaire réalisé pour l'estimation du paramètre m. Ce test a été réalisé sur la base de données SOTS, composée de 492 images. Ces dernières ont été évaluées avec les métriques SSIM et PSNR. Ce test a été réalisé avec plusieurs valeurs de p et a montré que la valeur $m = 0.3$ est optimale, mais que sa valeur précise n'est pas critique.

Cinq méthodes d'interpolation ont été comparées avec les métriques SSIM et PSNR sur la même base de données (voir le tableau 3.2) :

1. Notre fonction de modulation : $f_1(x)$;

2. Une fonction d'interpolation entre les fonctions $g(x)$ et $f_1(x)$;
3. Une fonction d'interpolation entre les fonction $g(x)$ et $f_2(x)$;
4. Un switch entre $g(x)$ et $f_1(x)$ avec un seuil à $x = I_0$;
5. Un switch entre $g(x)$ et $f_2(x)$ avec un seuil à $x = I_0$.

TABLE 3.2 – Comparaison des métriques SSIM et PSNR sur les quatre bases de données avec les cinq fonctions d'interpolation listées dans le texte avant le tableau. I est pour l'interpolation, S pour le switch. Les deux meilleurs résultats sont en gras.

	SSIM/PSNR			
Fonctions	**FRIDA**	**SOTS**	**NTIRE20**	**O-HAZE**
f_1	**0.81/13.34**	0.85/18.40	**0.50/13.22**	0.64/16.32
$I(f_1,g)$	0.78/11.65	**0.93/24.40**	0.49/12.75	**0.66/16.57**
$I(f_2,g)$	0.78/11.74	**0.93/24.00**	0.48/12.57	0.66/16.47
$S(f_1,g)$	0.79/13.09	0.91/23.01	**0.49/12.85**	**0.66/16.62**
$S(f_2,g)$	**0.79/13.29**	0.88/21.62	0.46/12.01	0.64/16.07

Le tableau 3.2 montre que la fonction f_1 seule est efficace sur les bases de données FRIDA et NTIRE20. Par contre, les fonctions interpolées $I(f_1,g)$ et $I(f_2,g)$ donnent de meilleurs résultats sur les bases de données SOTS et O-HAZE. L'utilisation de f_2 au lieu de f_1 n'améliore pas les performances de l'algorithme, pas plus que l'utilisation d'un switch entre $f_{1|2}$ et g. Les résultats sur les bases de données où le voile est spatialement proche d'un voile uniforme (SOTS et O-HAZE) montrent que la fonction constante est une alternative satisfaisante pour le traitement de ce type de brouillard. La fonction de modulation interpolée entre une constante et la fonction Naka-Rushton $I(f_1,g)$ traite plusieurs types de brouillard, et elle est compétitive avec les autres algorithmes listés dans la Section 3.5.1 sur tous les ensembles de données.

3.5 Évaluation

Dans la suite de l'évaluation nous distinguons trois versions de l'algorithme MFP :

1. MFP-W : version de l'algorithme en supposant un brouillard blanc
2. MFP-C : version de l'algorithme en supposant un brouillard coloré
3. MFP-I : version interpolée de l'algorithme introduit dans la section 3.4

3.5.1 Évaluation Quantitative

3.5.1.1 Évaluation avec des métriques standards

Les tableaux 3.3 et 3.4 montrent que toutes les méthodes sont compétitives, mais que notre méthode MFP-W surpasse les autres sur les deux critères pour les jeux de données FRIDA, NTIRE20 et O-HAZE. Sur le jeu de données O-HAZE, MFP-C, présente de meilleures performances car il parvient à atténuer les distorsions de couleur. Les résultats sur le jeu de données SOTS montrent que l'algorithme proposé est moins efficace sur les images où le voile est spatialement proche d'un brouillard uniforme, comme il est décrit dans la section 3.4. MFP-I permet de traiter les images avec des voiles constants et fournit de meilleurs résultats sur le jeu de données SOTS avec les métriques SSIM et PSNR. Cependant, de légères contre-performances ont

été relevées sur les trois autres bases de données par rapport aux deux autres versions de notre méthode, *MFP-W* et *MFP-C*. Ces résultats montrent que notre méthode peut se généraliser à plusieurs types de brouillard. Enfin, les performances générales de la méthode interpolée sont supérieures à celles des autres algorithmes.

TABLE 3.3 – Comparaison de l'indice PSNR sur quatre jeux de données : 50 images des jeux de données FRIDA et SOTS et 45 images des jeux de données NTIRE20 et O-HAZE. Les meilleurs résultats sont en gras.

PSNR

Méthodes	**FRIDA**	**SOTS**	**NTIRE20**	**O-HAZE**
DCP, He et al. (2011)	12.26	18.91	12.77	16.95
NBPC, Tarel (2009)	11.59	18.07	12.24	15.85
Zhu et al. (2018)	12.15	16.06	11.98	16.58
Zhu et al. (2021)	11.93	19.13	13.29	16.81
AOD-Net, Li et al. (2017)	10.73	19.39	11.98	15.04
Dehaze-Net, Cai et al. (2016)	10.87	23.41	12.33	15.41
GCA-Net, Chen et al. (2019)	**12.79**	22.68	12.82	16.43
FFA-Net, Qin et al. (2019)	10.38	**34.10**	12.40	16.19
MFP-W	**12.62**	18.40	**13.16**	**17.02**
MFP-C	12.27	16.77	**13.90**	**18.32**
MFP-I	11.65	**24.40**	12.69	16.57

TABLE 3.4 – Comparaison de l'indice SSIM sur quatre jeux de données : 50 images des jeux de données FRIDA et SOTS, 45 images des jeux de données NTIRE20 et O-HAZE. Les meilleurs résultats sont en gras.

SSIM

Méthodes	**FRIDA**	**SOTS**	**NTIRE20**	**O-HAZE**
DCP, He et al. (2011)	0.70	0.89	0.44	**0.66**
NBPC, Tarel (2009)	0.75	0.89	0.41	0.61
Zhu et al. (2018)	0.72	0.88	0.45	**0.66**
Zhu et al. (2021)	0.75	0.86	**0.55**	**0.67**
AOD-Net, Li et al. (2017)	0.73	0.85	0.41	0.54
Dehaze-Net, Cai et al. (2016)	0.65	0.90	0.44	0.60
GCA-Net, Chen et al. (2019)	0.70	0.91	0.47	0.61
FFA-Net, Qin et al. (2019)	0.73	**0.98**	0.46	0.63
MFP-W	**0.81**	0.85	**0.51**	0.65
MFP-C	**0.81**	0.82	**0.51**	**0.67**
MFP-I	0.78	**0.93**	0.48	**0.66**

Le tableau 3.5 montre que nos algorithmes sont compétitifs sur les quatre jeux de données. Les résultats obtenus avec la métrique FSIMc contribuent à prouver la généricité de notre méthode. En particulier, l'algorithme *MFP-I* est en concurrence avec des algorithmes d'apprentissage profond tels que *Dehaze-Net* et *FFA-Net* sur l'ensemble de données SOTS. Sur le jeu de données FRIDA, notre méthode et la méthode *DCP* sont les plus performantes. Les algorithmes *GCA-Net* et *AOD-Net* semblent surpasser les autres algorithmes sur les jeux de données

NTIRE20 ; Cependant, l'évaluation qualitative montre des distorsions de couleur importantes et des assombrissements dans les images restaurées. Nos méthodes donnent de très bons résultats sur ces deux jeux de données. De plus, *MFP-C* semble jouer un rôle dans l'amélioration des performances sur les jeux de données O-HAZE et NTIRE20. En effet, la version où le brouillard est supposé coloré semble plus performante que les autres pour gérer les distorsions de décalage vers le bleu.

TABLE 3.5 – Comparaison de l'indice FSIMc sur quatre jeux de données : 50 images des jeux de données FRIDA et SOTS, 45 images des jeux de données NTIRE20 et O-HAZE. Les meilleurs résultats sont en gras.

	FSIMc			
Méthodes	**FRIDA**	**SOTS**	**NTIRE20**	**O-HAZE**
DCP, He et al. (2011)	**0.83** (0.06)	0.96 (0.02)	0.55 (0.07)	**0.85** (0.06)
NBPC, Tarel (2009)	0.81 (0.06)	0.96 (0.01)	0.53 (0.06)	0.80 (0.08)
Zhu et al. (2018)	0.82 (0.06)	0.96 (0.01)	0.55 (0.06)	0.82 (0.08)
Zhu et al. (2021)	0.81 (0.06)	0.95 (0.02)	**0.73** (0.05)	**0.85** (0.08)
AOD-Net, Li et al. (2017)	0.79 (0.06)	0.93 (0.02)	0.67 (0.06)	0.78 (0.08)
Dehaze-Net, Cai et al. (2016)	0.81 (0.06)	**0.98** (0.01)	0.53 (0.07)	0.78 (0.09)
GCA-Net, Chen et al. (2019)	0.80 (0.05)	0.97 (0.02)	**0.74** (0.06)	**0.87** (0.06)
FFA-Net, Qin et al. (2019)	0.79 (0.06)	**0.99** (0.004)	0.66 (0.07)	0.80 (0.09)
MFP-W	**0.83** (0.06)	0.95 (0.02)	0.70 (0.06)	0.84 (0.07)
MFP-C	**0.83** (0.06)	0.95 (0.02)	0.71 (0.06)	**0.85** (0.07)
MFP-I	0.82 (0.06)	**0.98** (0.01)	0.69 (0.06)	0.83 (0.08)

Le PSNR mesure le rapport signal sur bruit de l'image reconstruite par rapport à sa référence alors que les métriques SSIM et FSIM mesurent respectivement la similarité entre les structures et les caractéristiques des images reconstruites et leur référence. Alors que le PSNR est très sensible au bruit dans les images, le SSIM estime la qualité des images par mesure de la similarité entre deux images en terme de luminance et de contraste. Cette métrique semble peu sensible aux variations de couleurs, mais l'est davantage à l'apparition d'artefacts. Le FSIM effectue au préalable un calcul de caractéristiques des images avant de mesurer la similarité. Cette métrique semble plus sensible aux variations basse fréquence dans l'images.

3.5.1.2 Évaluation avec d'autres métriques

PSNR, SSIM et FSIMc sont des métriques très largement utilisées en restauration d'image, mais ils ne sont pas suffisants pour évaluer efficacement la qualité des images après avoir appliqué une méthode de restauration. La figure 3.15a est un exemple de restauration réussi au niveau des objets en arrière-plan (voir la flèche à gauche), alors que l'image de la voiture au premier plan est dégradée (flèche à droite). À l'inverse, les résultats de la restauration de la figure 3.15b montrent que le premier plan est préservé, et l'arrière-plan dégradé. Malgré cela, les résultats obtenus avec les métriques SSIM et PSNR sont approximativement identiques :
SSIM = 0.89 et PSNR = 17.9 (Figure 3.15a) et SSIM = 0.89 et PSNR = 17.7 (Figure 3.15b).
Cet exemple révèle que les indices de qualité, tels que le PSNR, FSIMc et SSIM, peuvent être inadaptés, soit parce que la restauration est inefficace, soit parce que l'algorithme sur-restaure le contraste dans les zones où il n'y a pas de brouillard. En plus de ces métriques, nous proposons

d'estimer la qualité des images restituées avec deux cartes de poids (voir la figure 3.15c, d), qui sont denses dans les zones où le brouillard est respectivement dense ou léger.

De plus, selon l'application, diverses caractéristiques de l'image peuvent être conservées. Nous avons considérés trois caractéristiques importantes pour l'évaluation : l'intensité de l'image, l'intensité du gradient dans les images et nous reprenons la similarité de structure de l'image (SSIM). Dans la suite de cette sous-section, les performances des trois algorithmes proposés, nommés *MFP-W*, *MFP-C* et *MFP-I*, sont évaluées avec ces nouvelles métriques (voir la figure 3.16) :

— d1 : Distance pondérée entre la carte SSIM de la vérité terrain et de la carte SSIM des images restaurées (figure 3.16a).

— d2 : Distance pondérée entre la carte des gradients de la vérité terrain et la carte des gradients des images restaurées (figure 3.16b).

— d3 : Distance pondérée entre la vérité terrain et les images restaurées (figure 3.16c).

Les calculs de la distance pondérée sont définis par :

$$d_{Fog} = \frac{1}{p}\sum_{i=1}^{n} p(G_i - FR_i) \tag{3.7a}$$

$$d_{NoFog} = \frac{1}{(1-p)}\sum_{i=1}^{n} (1-p)(G_i - FR_i) \tag{3.7b}$$

où p est une carte de poids grossière associée aux régions brumeuses de l'image, G la vérité terrain, R l'image restaurée, et F un facteur de compensation d'intensité entre l'image restaurée et la vérité terrain. Dans cette partie, la carte de poids est une version normalisée du prévoile, permettant à p d'être fonction de la distance. d_{NoFog} accentue les parties de l'image proches de la caméra (figure 3.15c), alors que d_{Fog} accentue la partie brumeuse (figure 3.15d).

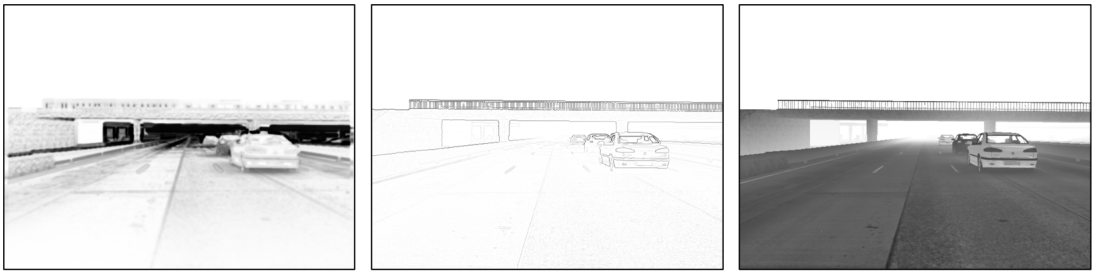


FIGURE 3.16 – (**a**) carte SSIM d'une image restaurée à partir du jeu de données FRIDA, (**b**) carte des gradients, (**c**) carte d'intensité.

En plus de ces métriques, un critère basé sur la restauration des contours a été considéré. Elle consiste à appliquer un filtre de contour, tel que le détecteur de contour de Canny (1986), à la fois sur la vérité terrain et sur les images restaurées, et de comparer les deux cartes de contour binaires afin d'évaluer le bénéfice de la restauration en terme de visibilité des bords. Ce critère peut être utile, par exemple, pour évaluer la restauration d'images des algorithmes de détection intégrées aux caméras embarquées. La figure 3.17 montre les cartes de contours obtenues avec le filtre de Canny sur la même scène sans brouillard, avec brouillard, et après élimination du brouillard.

TABLE 3.6 – Comparaison de trois indices de distance d1, d2 et d3 sur 50 images du jeu de données SOTS, avec l'indice d_{Fog} (dans la zone de brouillard). Les deux meilleurs résultats sont en gras. Le meilleur résultat est souligné.

	d_{Fog}		
Méthodes	**d1**	**d2**	**d3**
DCP, He et al. (2011)	**<u>0.01</u>** (0.05)	0.08 (0.003)	24.3 (8.6)
NBPC, Tarel (2009)	0.08 (0.02)	0.01 (0.003)	23.4 (10.8)
Zhu et al. (2018)	0.60 (0.03)	0.05 (0.02)	67.7 (12.7)
Zhu et al. (2021)	0.10 (0.05)	0.01 (0.004)	33.5 (15.8)
AOD-Net Li et al. (2017)	0.12 (0.04)	0.01 (0.003)	28.4 (16.6)
Dehaze-Net Cai et al. (2016)	0.06 (0.05)	**0.006** (0.002)	24.0 (15.6)
GCA-Net, Chen et al. (2019)	0.08 (0.04)	0.01 (0.004)	**22.7** (15.6)
FFA-Net, Qin et al. (2019)	**<u>0.01</u>** (0.005)	**<u>0.004</u>** (0.001)	**<u>4.97</u>** (2.3)
MFP-W	0.10 (0.06)	0.009 (0.002)	42.0 (16.8)
MFP-C	0.08 (0.06)	0.01(0.002)	30.3 (16.8)
MFP-I	**0.04**(0.03)	**0.006**(0.003)	23.5 (14.9)

TABLE 3.7 – Comparaison de trois indices de distance d1, d2 et d3 sur 50 images du jeu de données SOTS, avec l'indice d_{NoFog} (dans la zone sans brouillard). Les deux meilleurs résultats sont en gras. Le meilleur résultat est également souligné.

	d_{NoFog}		
Méthodes	**d1**	**d2**	**d3**
DCP, He et al. (2011)	0.09 (0.05)	0.01 (0.003)	16.4 (6.5)
NBPC, Tarel (2009)	0.10 (0.04)	0.01 (0.003)	15.0 (4.5)
Zhu et al. (2018)	0.70 (0.05)	0.05 (0.002)	54.5 (5.25)
Zhu et al. (2021)	0.15 (0.08)	0.02 (0.005)	19.8 (5.0)
AOD-Net, Li et al. (2017)	0.20 (0.06)	0.02 (0.004)	18.7 (6.9)
Dehaze-Net, Cai et al. (2016)	0.12 (0.1)	0.007 (0.002)	14.8 (6.6)
GCA-Net, Chen et al. (2019)	0.11 (0.05)	0.01 (0.005)	15.4 (8.2)
FFA-Net, Qin et al. (2019)	**<u>0.01</u>** (0.008)	**<u>0.005</u>** (0.001)	**<u>4.40</u>** (1.9)
MFP-W	0.20 (0.1)	0.009 (0.003)	24.5 (5.7)
MFP-C	0.11 (0.1)	0.01 (0.003)	18.8 (5.9)
MFP-I	**0.08** (0.06)	**0.006** (0.002)	**14.2** (6.1)

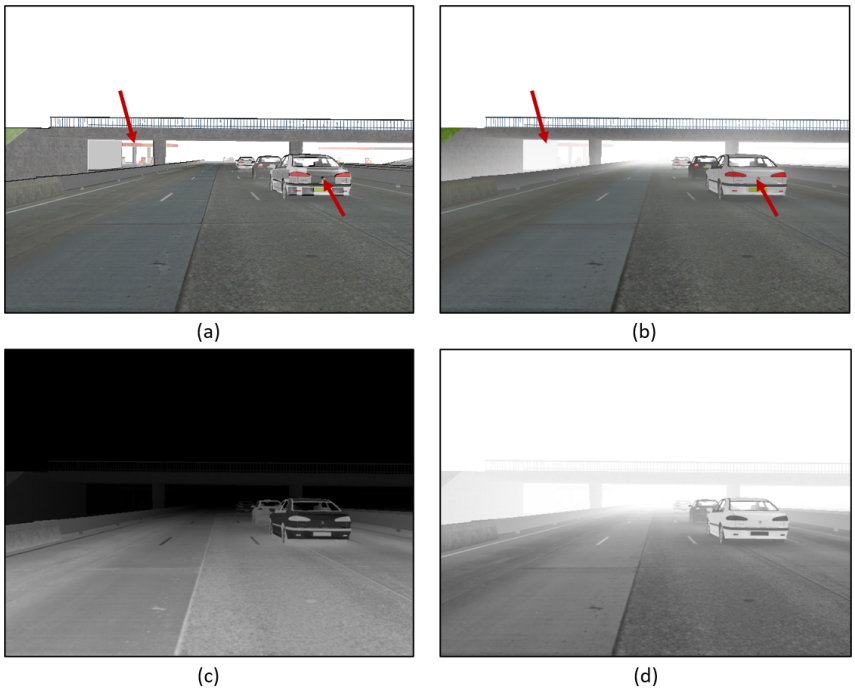


FIGURE 3.15 – Images de la base de données FRIDA : (**a**) image restaurée ; SSIM = 0.89 et PSNR = 17.9 ; (**b**) image restaurée ; SSIM = 0.89, PSNR = 17.7, (**c**) carte de poids pondérée sur la partie inférieure de l'image d'entrée, et (**d**) carte de poids pondérée sur la partie brumeuse de l'image d'entrée.

Les deux cartes de contours binaires sont comparées à l'aide d'une courbe ROC, dont une variation du seuil minimale entre 0 et la valeur seuil maximale (voir la figure 3.18). L'image de gauche sur la figure 3.18 montre que la méthode de Zhu et al. (2018), *MFP-W*, le *DCP*, et la méthode multi-exposition de Zhu et al. (2021), produisent de meilleurs résultats que les autres en termes de restauration de la visibilité des contours. Notre algorithme d'interpolation, le *NBPC* et le *GCA-Net* sont également compétitifs pour la restauration des bords, avec des performances moindres.

L'image de droite de la figure 3.18 montre que, bien que *FFA-Net* parvient à mieux conserver des contours sur le jeu de données SOTS que les autres méthodes, il présente néanmoins de moins bons résultats sur le jeu de données FRIDA. Cet algorithme a été entraîné pour être très efficace sur des images avec un voile spatialement uniforme (comme dans les jeux de données RESIDE/SOTS). L'évaluation quantitative montre qu'elle ne se généralise pas bien aux différents types de brouillard. Après *FFA-Net*, *MFP-I* et Zhu et al. (2018) fournissent de très bons résultats, alors que *DCP* et *Dehaze-Net* sont un peu moins compétitifs sur le jeu de données SOTS. *MFP-W* donne de meilleurs résultats en termes de visibilité des contours que *MFP-I* sur le jeu de données FRIDA, alors que c'est l'inverse sur le jeu de données SOTS. Cela renforce l'idée que la version Interpolation donne de meilleurs résultats avec un brouillard spatialement uniforme.

3.5.2 Évaluation Qualitative

Des images du monde réel issues de travaux antérieurs sur l'élimination du brouillard dans des images ont été utilisées pour réaliser une évaluation qualitative. Les algorithmes *DCP*,

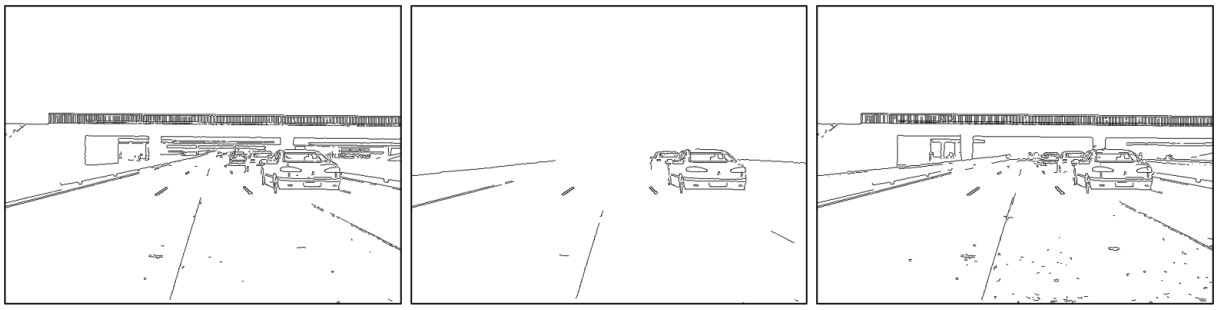


FIGURE 3.17 – Cartes de contours après application du filtre de Canny sur une image de la base de données FRIDA : (**a**) sans brouillard, (**b**) avec brouillard, (**c**) après restauration. Les seuils min et max du détecteur de bords sont fixés à 120 et 180.

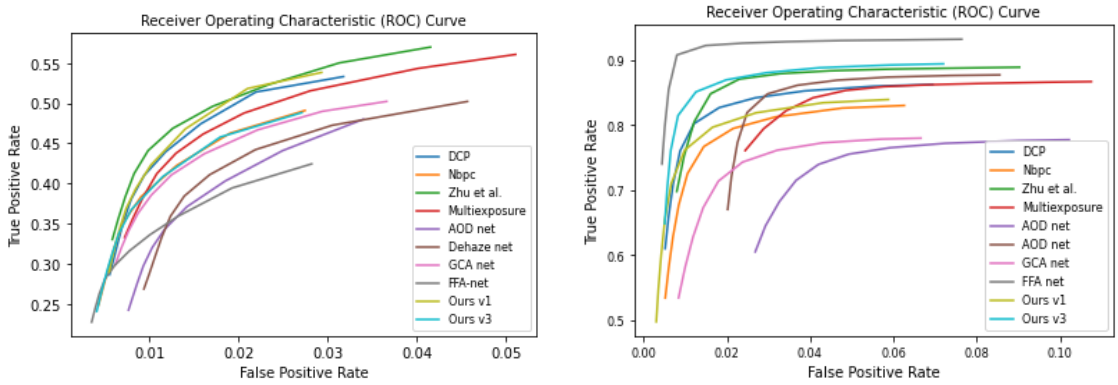


FIGURE 3.18 – Comparaison des courbes ROC pour différents algorithmes. (**Gauche**) : courbes ROC obtenues à partir d'une cinquantaine d'images du jeu de données FRIDA. (**Droite**) : courbe ROC obtenue à partir d'une cinquantaine d'images du jeu de données SOTS. Le seuil max du filtre Canny est fixé à 180. La légende multi-exposition correspond à la méthode de Zhu et al. (2021).

NBPC et celui de Zhu et al. (2018), restaurent les images avec de résultats satisfaisants (voir la figure 3.19). Cependant, les images obtenues avec *DCP* sont très lumineuses et contrastées, alors que les résultats de *NBPC* et de Zhu et al. (2018) sont plus sombres et plus pâles. La méthode de Zhu et al. (2021) fournit des résultats très contrastés et saturés, particulièrement notables dans les images avec des citrouilles et des montagnes. La méthode basée sur l'apprentissage *AOD-Net* fournit des images ternes et sombres, mais avec moins de halos lumineux. Dans les images avec l'arbre et dans celles avec des bâtiments, les méthodes *Dehaze-Net* et *FFA-Net* semblent retenir de la brume au loin. L'hypothèse de garder de la brume au niveau de l'horizon fonctionne mieux dans le ciel, évitant ainsi les artefacts et les distorsions de couleurs récurrents dans certaines méthodes. *GCA-Net* produit des artefacts dans le ciel de l'image de l'arbre et des images des deux premières lignes, mais fournit des résultats satisfaisants et colorés dans d'autres images. *MFP-W* fournit des images claires et supprime la brume sur l'ensemble des images. *MFP-C* réduit les distorsions de couleur.

La figure 3.20 montre les résultats sur les images de l'ensemble de données O-HAZE avec deux versions de notre algorithme. Les résultats obtenus avec *MFP-W* (figure 3.20c) sont décalés vers le bleu et bruités. *MFP-C* parvient à atténuer l'effet de décalage vers le bleu (voir la figure

FIGURE 3.19 – Comparaison des résultats après restauration des images contenant du brouillard sur des images réelles : (**a**) images d'entrée, (**b**) *DCP*, He et al. (2011), (**c**) *NBPC*, Tarel (2009), (**d**) Zhu et al. (2018), (**e**) Zhu et al. (2021), (**f**) *Dehaze-Net*, Cai et al. (2016) (**g**) *AOD-Net*, Li et al. (2017), (**h**) *GCA-Net*, Chen et al. (2019), (**i**) *FFA-Net*, Qin et al. (2019), (**j**) *MFP-W*, et (**k**) *MFP-C*.

3.20d) et contribue à améliorer le critère décrit dans la sous-section évaluation quantitative.

3.5.3 Robustesse de l'algorithme proposé

La figure 3.21 montre des exemples de résultats de notre algorithme appliqué à des images nocturnes. Pour tester la pertinence de notre algorithme ce type d'image, une comparaison a été réalisée avec deux algorithmes de l'état de l'art, Li et al. (2015) et Yu et al. (2019). La figure 3.21 montre que l'algorithme de Li et al. réussit à atténuer les halos. Cependant, les images sont colorées et bruitées, en particulier dans la région du ciel. Au contraire, l'algorithme de Yu et al. produit des images claires et contrastées avec une cohérence des couleurs et un rendu naturel, alors que les halos semblent accentués, en particulier dans la partie inférieure de l'image. La version couleur de notre algorithme réussit à atténuer les halos et à maintenir un rendu naturel des couleurs. Ce constat est particulièrement notable dans la partie inférieure de l'image ; dans la partie haute de l'image, les objets éloignés ne sont pas assez contrastés.

La figure 3.22 présente une comparaison entre les résultats produits par les méthodes *MFP-W* et *MFP-C* d'une image représentant un halo avec un brouillard orange. La figure 3.22b montre que *MFP-W* semble accentuer l'effet de halo et ne supprime pas le brouillard coloré. En revanche, *MFP-C* (voir la figure 3.22c) parvient à atténuer l'effet de halo et le décalage vers la couleur orange. L'histogramme de la figure 3.22d présente la distribution d'intensité de l'image. La zone bleue correspond à l'image en entrée. Elle est uniformément répartie sur la plage d'intensité [0.2,1.0], qui représente la zone brumeuse de l'image. Par conséquent, l'histogramme de l'image restaurée devrait, idéalement, culminer vers le côté gauche de l'histogramme.

La figure 3.23 montre une comparaison entre l'algorithme *MFP-C* et les deux autres algorithmes de l'état de l'art mentionnés dans la figure 3.21. Dans la figure 3.23b, les résultats

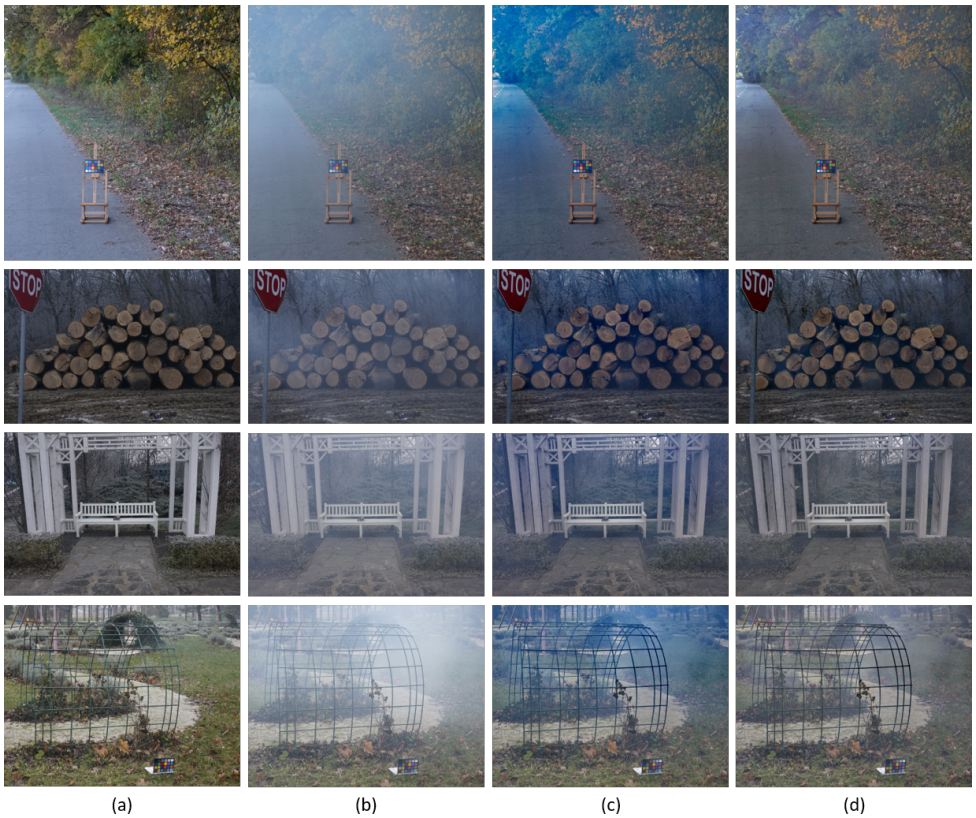


FIGURE 3.20 – Images de la base de données O-HAZE Ancuti et al. (2018) : (**a**) sans brouillard, (**b**) avec brouillard, (**c**) images restaurée avec MFP-W, (**d**) images restaurée avec MFP-C.

TABLE 3.8 – Comparaison des indices SSIM et PSNR sur vingt images nocturnes de synthèse.

Méthodes	**SSIM**	**PSNR**
Yu et al. (2019)	0.62	13.39
Li et al. (2015)	0.61	12.59
MFP-C	0.60	11.33

montrent que l'algorithme de Li et al. réussit à éliminer l'effet de halos, mais introduit des distorsions de couleur. Le résultat obtenus avec la méthode de Yu et al. (voir la figure 3.23c) montre que celle-ci semble contribuer à atténuer légèrement le brouillard coloré. Cependant, la distribution de l'histogramme (la partie jaune) montre que le brouillard autour de la source artificielle n'est pas complètement supprimé. Un pic dans la partie gauche de l'histogramme semble être un bon indicateur de la suppression du brouillard nocturne (Our C et Li et al. (2015)).

Le tableau 3.8 présente une petite évaluation quantitative (SSIM, PSNR) sur une base de données d'une vingtaine d'images nocturnes de synthèse créées par Zhang et al. (2020a) de l'ensemble de données Middlebury (voir un exemple dans la figure 3.24). Les résultats montrent que notre méthode fournit des résultats légèrement inférieurs, mais reste compétitive avec les méthodes dédiées à la restauration d'images nocturnes. Cela montre que la méthode simple, qui est d'appliquer notre méthode de restauration sur chaque canal de couleur, donne des résultats prometteurs. La figure 3.25 montre que notre algorithme, appliqué sur chaque canal de couleur, peut également être encourageant pour la restauration de la visibilité sous-marine.

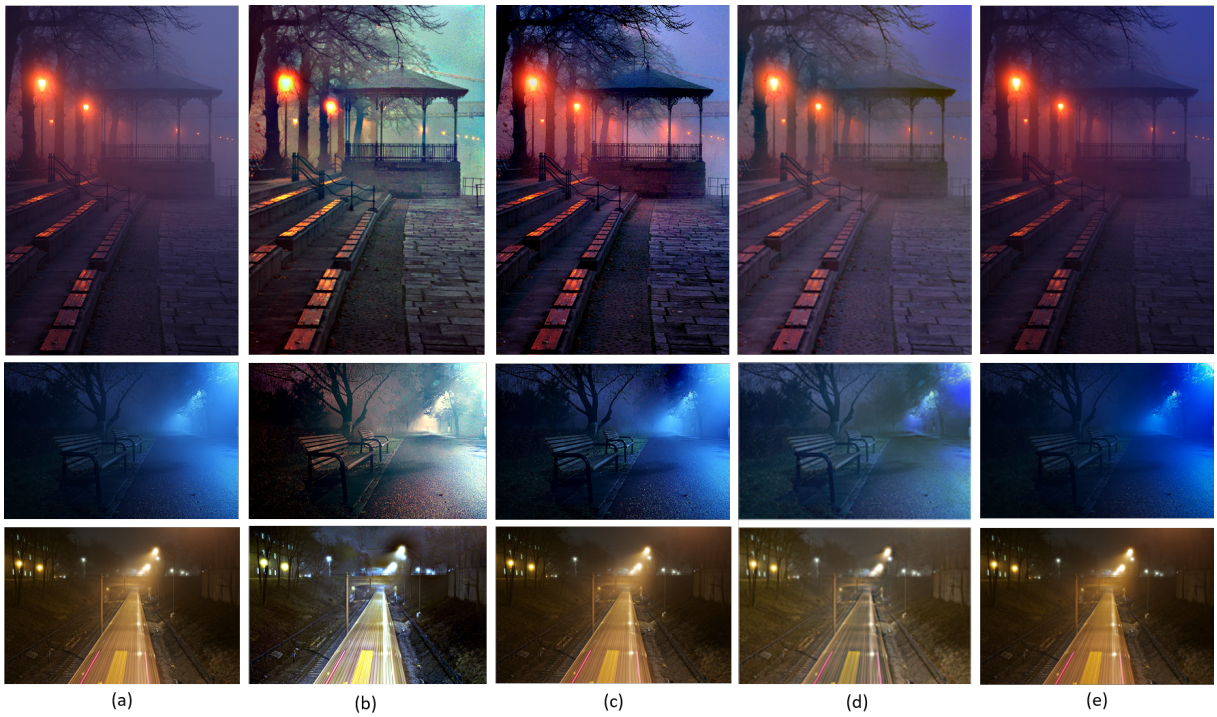


FIGURE 3.21 – Comparaison sur des images nocturnes, Zhang et al. (2017b) : (**a**) Images avec du brouillard, (**b**) Li et al. (2015), (**c**) Yu et al. (2019), (**d**) MFP-C, (**e**) MFP-W.

3.6 Conclusion et perspectives

Nous avons réinterprété la méthode *DCP* en fonction de trois *a priori*. Nous proposons d'améliorer le troisième a priori, associé au paramètre ω, avec une fonction de modulation lisse pour estimer le voile atmosphérique à partir du pré-voile. Les paramètres en entrée de cette fonction sont automatiquement estimés en fonction des intensités des pixels de l'image d'entrée dans les régions claires (ciel) et sombres (sol), après filtrage. De plus, notre méthode permet de généraliser à différents types de brouillard, à la fois le brouillard dépendant de la profondeur et le brouillard spatialement uniforme. L'évaluation et les résultats obtenus sur les différentes bases de données avec plusieurs types de brouillard témoignent de la généralisabilité de notre algorithme. La méthode proposée fournit des résultats très satisfaisants sur les images de synthèses et les images réelles d'objets, quelle que soit la distance par rapport à la caméra. Pour étendre l'algorithme proposé au traitement de la fumée, de la poussière et d'autres particules colorées en suspension dans l'air, nous traitons chaque canal de couleur séparément, afin d'atténuer le brouillard coloré. Cela permet d'appliquer l'algorithme aux images nocturnes ainsi qu'aux images sous-marines. Afin de répondre au traitement des images avec une luminance de voile plus ou moins uniforme, une interpolation doit être appliquée entre la fonction de Naka-Rushton et celle du voile constant. Les algorithmes proposés doivent être rigoureusement évalués, et les résultats correctement analysés pour les appliquer aux images de jour, de nuit et éventuellement sous l'eau. Des métriques complémentaires ont été également proposées en plus des métriques classiques pour l'évaluation de l'algorithme MFP.

Ce chapitre a fait l'objet d'un article de conférence (Duminil et al. (2021a)) et d'un article de journal (Duminil et al. (2021b)). L'article de conférence avait été initialement refusé car considéré comme marginal par rapport aux méthodes d'apprentissage actuellement en vogue.

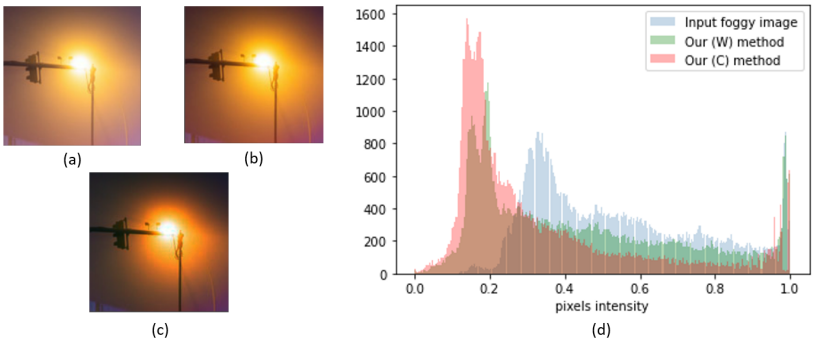


FIGURE 3.22 – Comparatif sur une imagette avec un halo : (**a**) image d'entrée avec du brouillard, (**b**) imagette restaurée avec notre méthode *MFP-W*, (**c**) imagette restaurée avec notre méthode colorée, (**d**) histogramme d'intensité de chaque imagette (les images de halos sont réutilisées avec la permission de Broyer (2017), Copyright 2021 Mark Broyer).

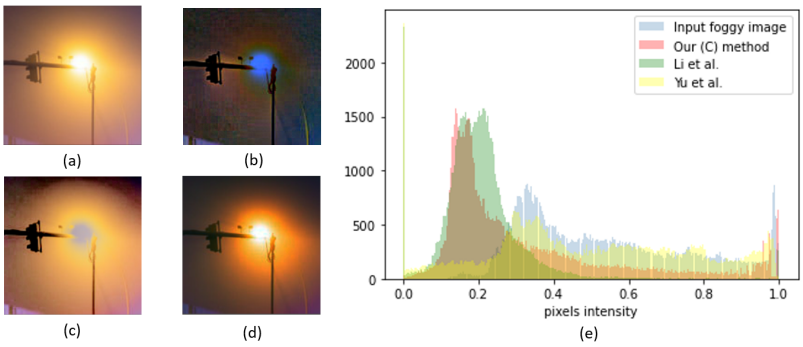


FIGURE 3.23 – Comparatif sur une imagette avec un halo :(**a**) image d'entrée avec du brouillard, (**b**) Li et al. (2015), (**c**) Yu et al. (2019), (**d**) imagette restaurée avec *MFP-C*, (**e**) histogramme d'intensité de chaque imagette (les images de halos sont réutilisées avec la permission de Broyer (2017), Copyright 2021 Mark Broyer).

FIGURE 3.24 – Image extraite de l'ensemble de données de Hirschmuller and Scharstein (2007) : (**a**) image d'entrée, (**b**) image synthétique nocturne de Zhang et al. (2020a). Dans le cas des images de synthèse, la taille du filtre d'ouverture est fixée à (30,30) au lieu de (10,10).

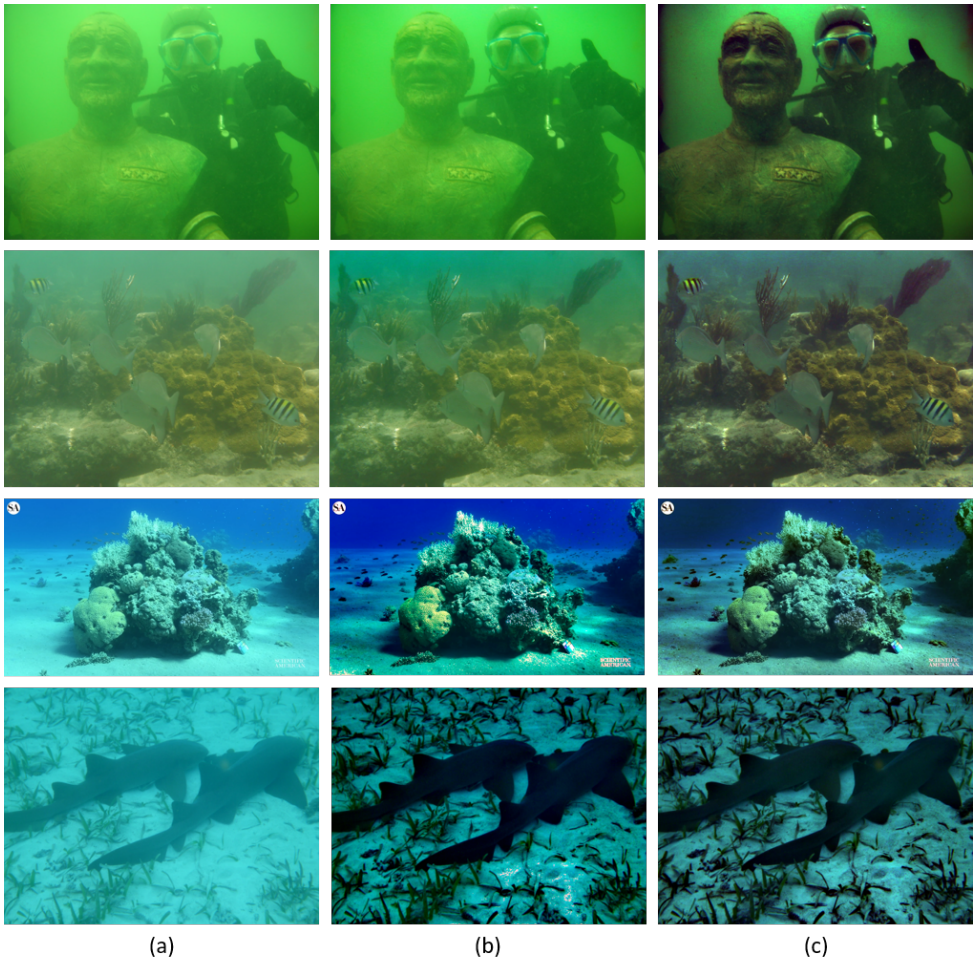


FIGURE 3.25 – Images sous-marines de Wang et al. (2019c) : (**a**) images d'entrée, (**b**) images restaurées avec MFP-W, (**c**) images restaurées avec MFP-C.

Chapitre 4

Restauration de vidéos avec du brouillard

Création d'une base de données vidéo et d'un algorithme hybride

4.1 Introduction

La restauration de la visibilité dans les images et les vidéos contenant du brouillard est un problème récurrent dans les applications de vision par ordinateur. Les artefacts visuels dans les images, tels que la perte de contraste et le changement de couleur, contribuent à réduire la visibilité de la scène. Une visibilité réduite dégrade les performances des algorithmes de vision par ordinateur, tels que la segmentation, la détection et la reconnaissance d'objet/scène. Par conséquent, il est nécessaire de mettre en œuvre des algorithmes d'atténuation de brouillard comme méthode de pré-traitement. De nos jours, de nombreux travaux sont réalisés en atténuation de brouillard dans des images uniques, avec des algorithmes toujours plus performants. Les algorithmes récents basés sur l'apprentissage profond fournissent d'excellents résultats. Des travaux récents mettant en œuvre des méthodes traditionnelles basées sur des *a priori* rivalisent avec ces nouvelles approches de deep learning, désormais majoritaires (voir la méthode mise en œuvre dans le chapitre 3).

L'utilisation de l'information temporelle des trames adjacentes d'une vidéo, doit être capable d'améliorer l'efficacité des méthodes d'atténuation de brouillard dans des vidéos. Cependant, les travaux concernant la restauration de vidéos contenant du brouillard sont encore assez rares. La technique de suppression du brouillard dans une vidéo semble moins accessible que pour les images, en raison du manque de bases de données de vidéos contenant du brouillard pour lesquelles il existe une vérité terrain. Récemment, Zhang et al. (2020b) ont proposé une base de données utilisable pour l'apprentissage et l'évaluation, appelée *REVIDE*. Le brouillard étant produit avec une machine à brouillard, les scènes sont plus réalistes que les images construites avec un brouillard de synthèse.

Les premiers travaux en restauration de vidéos étaient abordés comme des extensions des méthodes de restauration d'images uniques. Cependant, des artefacts visuels et des scintillements dues à un exposition inégale entre trames peuvent apparaître par manque de cohérence temporelle. Comme présenté dans l'état de l'art des méthodes de restauration dans des vidéos du chapitre 2, il a été proposé d'introduire un calcul de flot optique pour assurer une meilleure cohérence temporelle. Les meilleurs méthodes de flot optique étant par apprentissage CNN, autant adopter cette approche pour la restauration de vidéo. Dans le chapitre 2, de nombreuses

méthodes y sont citées avec différentes architectures composées de modules temporels. Parmi ces méthodes, celles qui sont fondées sur une architecture *Transformer* semblent surpasser les autres, notamment les méthodes basées sur des réseaux de neurones récurrents (RNNs). En effet, ces derniers sont lents à entraîner, peu parallélisables, et peinent à garder en mémoire l'information trop lointaine dans le passé. Les *Transformers* n'utilisent pas de réseaux récurrents mais un mécanisme d'attention qui constitue le cœur de leur architecture, et sont parallélisables. L'interdépendance au sein d'une séquence d'images est conservée en traitant les données en une seule fois, plutôt qu'un traitement image par image. Alors que des *Transformers* « purs » sont mis en œuvre dans les méthodes de restauration pour résoudre leurs tâches, je propose d'utiliser cette architecture pour traiter l'information temporelle des vidéos. La méthode de restauration de vidéos que je présente est dite hybride. L'architecture globale est basée sur un auto-encodeur inspiré de U-net et intègre des modules CNN pour traiter l'information spatiale, et des modules *Transformers* pour traiter l'information temporelle.

Dans ce chapitre, une nouvelle base de données de tests pour la comparaison d'algorithmes de restauration de vidéos contenant du brouillard, appelée VIREDA, a été créée pour les besoins de la thèse et la mise à la disposition de la communauté. Cette base de tests est accessible à partir du lien suivant : https://github.com/alex-dml/VIREDA-video-dehazing. Cette dernière contient des vidéos d'une scène avec plusieurs densités de brouillard et conditions d'illumination. Les méthodes de création du dispositif, d'acquisition et de traitement des données sont détaillées dans la section 4.2.1. Enfin, l'algorithme de restauration vidéo est présenté dans la section 4.3, et son évaluation quantitative et qualitative à la section 4.4 montre des résultats satisfaisants ainsi que l'intérêt de l'algorithme proposé.

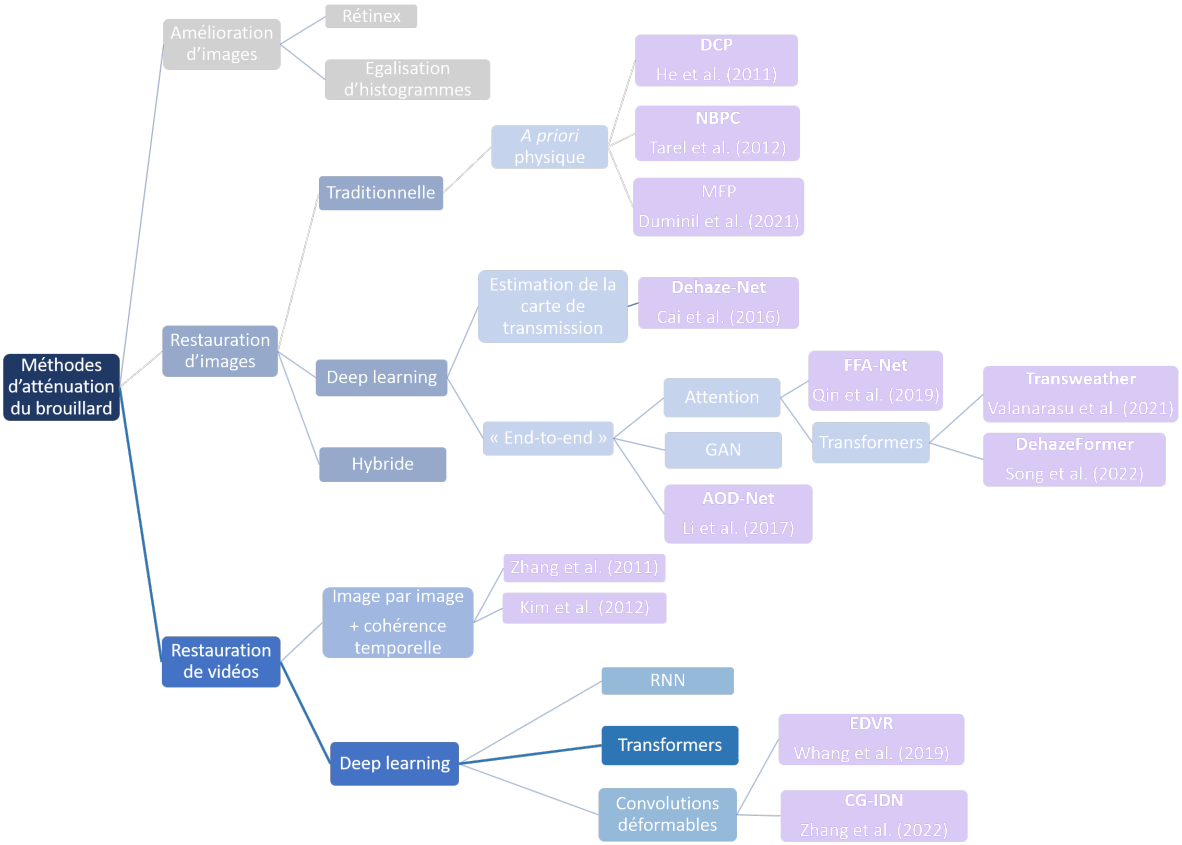


FIGURE 4.1 – Diagramme récapitulatif de l'état de l'art.

La figure 3.2 permet de situer les travaux présentés dans ce chapitre par rapport aux travaux de l'état de l'art.

4.2 Réalisation d'une base de données de test de vidéos contenant du brouillard

En complément de la base de données *REVIDE*, la base de données de tests VIREDA est proposée, composée de vidéos contenant du brouillard, de vérités terrain et de cartes de profondeur.

4.2.1 Dispositif

4.2.1.1 L'aquarium

L'objectif est de réaliser des vidéos contenant du brouillard de différentes densités dans différentes conditions d'illumination. Pour répondre à ce besoin, une chambre à brouillard a été mise en place afin de collecter des paires de vidéos avec et sans brouillard, ainsi que des cartes de profondeur associées. La chambre à brouillard est composée de plaques de polycarbonates de dimensions 2.05 x 1.25 x 0.61 mètres, et dans laquelle sont répartis divers objets de différentes tailles (voir figure 4.2).

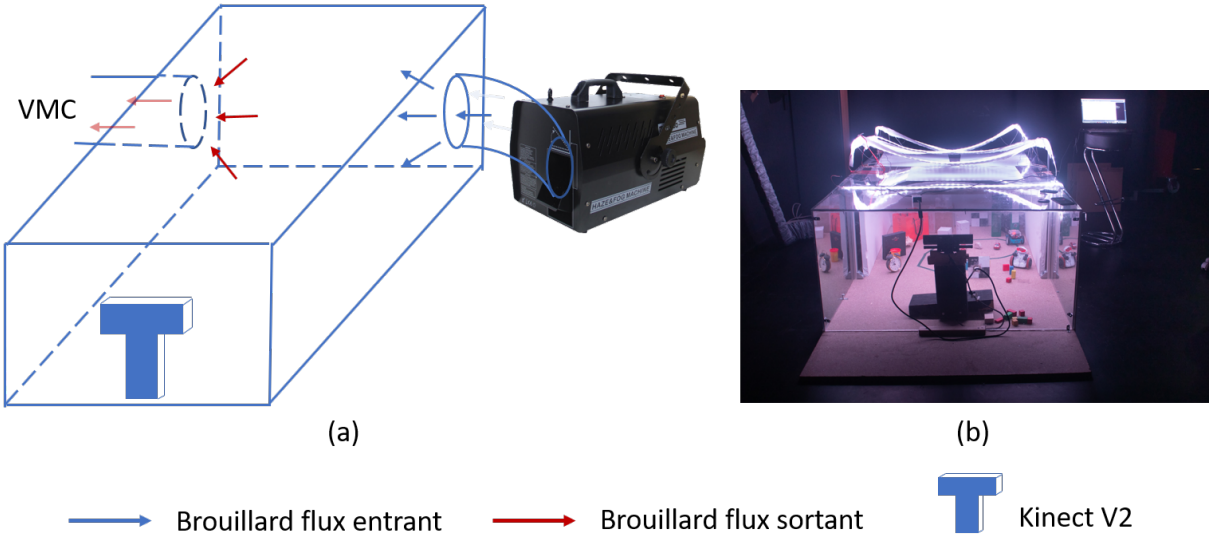


FIGURE 4.2 – La chambre à brouillard : (a) Schéma de la chambre à brouillard et de son fonctionnement, (b) Photo du dispositif.

La sortie de la machine à brouillard est connectée à la chambre pour injecter du brouillard dans le dispositif. Ce dernier dispose d'une ventilation pour l'évacuation du brouillard en fin d'acquisition d'une séquence vidéo. La machine à brouillard est illustrée par une photo à la figure 4.3. Cette machine hybride permet de basculer au besoin entre une machine à brouillard et une machine à fumée. En mode fumée, elle produit un fumée dense et opaque qui s'étend au sol. Pour notre application, le mode brouillard était plus adapté car la machine produit alors un brouillard modulable, plus léger, qui se propage dans tout le volume de manière plus homogène.

FIGURE 4.3 – Machine à brouillard J-Collyns MFH-900

Afin de diversifier les scénarios, des rubans à LED pilotés par une carte arduino Uno ont été positionnés au-dessus de la chambre à brouillard pour simuler différents types d'éclairage. L'objectif était de reproduire le plus fidèlement possible les conditions lumineuses réelles en obtenant une luminosité homogène afin de respecter les conditions de la loi de Koschmieder (brouillard et éclairage homogène). Pour cela, du tissu et du papier calque ont été utilisés pour favoriser la diffusion de la lumière dans la scène.

4.2.1.2 L'éclairage

L'utilisation de quatre rubans à LED pilotés et orientés vers l'intérieur de la chambre à brouillard a permis de produire différentes conditions de luminosité. Ainsi, six jeux d'éclairages différents alternent cycliquement pendant la phase d'acquisition des vidéos. Pour simuler un éclairage hétérogène (de nuit), le brouillard doit être de couleurs différentes et être visible autour de sources ponctuelles. Des bandes de LED bleues et une bande rouge ont été utilisées pour simuler l'éclairage de nuit. Les cinq autres éclairages sont réalisés avec des LED blanches pour simuler l'éclairage de jour. Dans certains cas, deux rubans de LED sur quatre sont éteints pour créer des zones d'ombre dans la scène (voir la figure 4.8).

4.2.1.3 Le brouillard

La machine à brouillard permet de moduler la densité du brouillard produit indiquée en pourcentage. Ainsi, plus le pourcentage est élevé, plus la densité est élevée. Cependant, ce système ne nous permet pas de quantifier la densité de brouillard dans le volume de la boîte. Pour obtenir une indication sur la densité de brouillard dans le dispositif, je me suis inspirée des travaux expérimentaux de Hautière et al. (2006) en l'équipant de deux mires damiers. Grâce à ces deux mires, des valeurs de contraste C, de coefficient d'extinction k et de distance de visibilité V_{met} ont été calculées. Pour calculer le contraste de chaque mire, nous nous sommes également inspirés de l'équation du contraste de Michelson défini comme suit :

$$C_{Michelson} = \frac{L_{max} - L_{min}}{L_{max} + L_{min}} \tag{4.1}$$

Avec L_{max} la luminance maximum de l'image et L_{min} la luminance minimum de l'image.
Les mires damiers ont été positionnées à deux distance différentes de la Kinect, l'une à 60 centimètres et l'autre à 1 mètre 80 (voir la figure 4.5). Ces mires damiers ont permis d'obtenir l'intensité des zones noires et des zones blanches des images. Afin de diminuer l'impact du bruit dans les résultats des calculs, un calcul de moyenne sur des fenêtres locales dans les zones noires et blanches de la mire est réalisé. L'équation 4.1 devient alors :

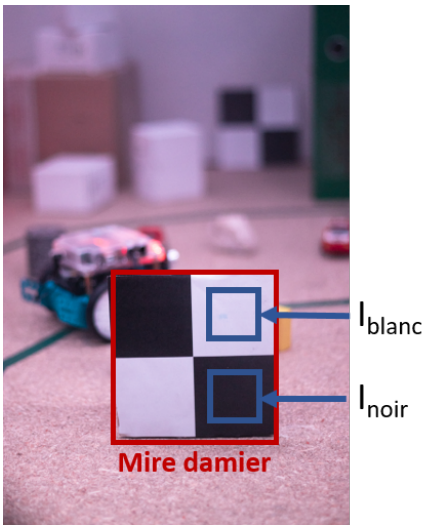


FIGURE 4.4 – Mire damier utilisée pour le calcul du contraste.

$$C = \frac{mean(I_{blanc}) - mean(I_{noir})}{mean(I_{blanc}) + mean(I_{noir})} \tag{4.2}$$

où I_{blanc} et I_{noir} correspondent respectivement aux intensités des pixels présents dans une fenêtre locale centrée sur les zones noires et blanches des mires (voir la figure 4.4). La fonction *mean* appliquée sur les fenêtres locales permet d'effectuer un lissage dans ces fenêtres.
Pour calculer la valeur de k, j'ai utilisé la loi d'atténuation de Duntley, rappelée dans le chapitre 2 à la sous-section 2.1.2.4. Pour rappel, elle est défini par la formule suivante :

$$C = C_0 e^{-kd} \tag{4.3}$$

où C correspond au contraste perçu dans le brouillard par l'observateur, C_0 est le contraste sans brouillard, k est le coefficient d'extinction atmosphérique et d est la distance observateur/cible. Les calculs de contraste, et par conséquent des coefficients d'extinctions, ont révélés qu'ils étaient différents pour chacune des mires.
Cette différence indique que le brouillard n'est pas homogène dans le dispositif. Il apparaît en effet plus dense au fond du dispositif. En conséquence, c'est la mire 2 qui a été utilisée pour déterminer les différentes densités de brouillard. Les valeurs de contraste de référence choisies pour la base de données sont des indications : 0.015, 0.05 et 0.15. Les valeurs de contraste réelles calculées à partir de la mire 2 ne peuvent pas être parfaitement égale à ces valeurs. Le tableau 4.1 ci-dessous récapitule les valeurs moyennes réelles de C, k et V_{met} pour chaque C_{ref}.

TABLE 4.1 – Valeurs moyennes réelles de C, k et V_{met} pour chaque valeur de contraste de référence C_{ref} choisie pour la base de données.

C_{ref}	C	k	V_{met}
0.015	0.03	1.66	1.79
0.05	0.07	1.51	2.31
0.15	0.16	0.83	3.60

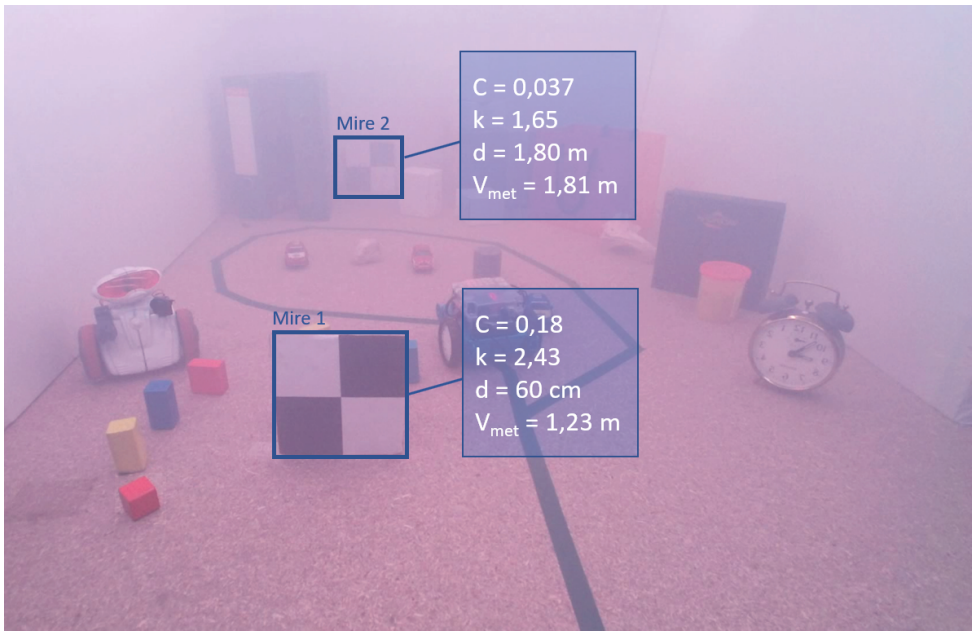


FIGURE 4.5 – Exemple de calculs des valeurs C, k et V_{met} à partir des mires.

A l'échelle 1 :10, les distances de visibilité météorologiques pour les trois valeurs de contrastes données dans le tableau 4.1 seraient de 17.9, 23.1 et 36.0 mètres, correspondant à des brouillards denses dans la réalité.

4.2.1.4 Le robot

Plusieurs objets ont été ajoutés dans le dispositif pour la mise en place de la scène. Composée de simples objets, la scène est statique et donc inadaptée à la vidéo. Afin d'obtenir une scène dynamique pour la production des vidéos de notre application, nous y avons ajouté un robot suiveur de ligne mBot. Ce dernier est un robot modulaire programmable et télécommandable, dotés de plusieurs capteurs. Le capteur suiveur de ligne entouré en rouge dans la figure 4.6 est utilisé pour notre application. Ce module comporte deux capteurs de contraste constitués chacun d'une LED émettrice et d'un phototransistor permettant de détecter une ligne, tracée sur le sol dans l'aquarium.

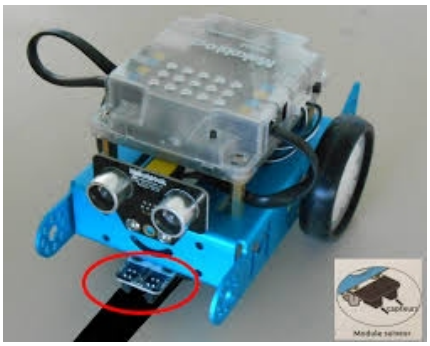


FIGURE 4.6 – Photo du robot mBot : le capteur suiveur de ligne est entouré en rouge. L'image provient du site http://sti.ac-bordeaux.fr/techno/coder/mbot/5_suivre_une_ligne.html.

4.2.1.5 La Kinect

Les données de la base ont été acquises par les caméras d'une Kinect V2. Elle offre la possibilité d'obtenir des cartes de profondeur et des images avec une résolution satisfaisante :

— Résolution de la caméra couleur : 1920×1080 pixels

— Résolution de la caméra de profondeur : 512×424 pixels

— Champs de vision : 70° (H), 60° (V)

— Intervalle de mesure : 0.5m - 4.5m

— Technologie du capteur de profondeur ToF

Cette caméra a permis d'enregistrer des cartes de profondeur par l'intermédiaire d'un capteur Time of Flight (ToF). Ce capteur mesure le temps mis par la lumière pour parcourir une distance entre l'émission du signal et son retour vers le capteur (voir le schéma figure 4.7). Comme il n'est pas sensible au changement de luminosité, une carte de profondeur par position du robot était suffisante.

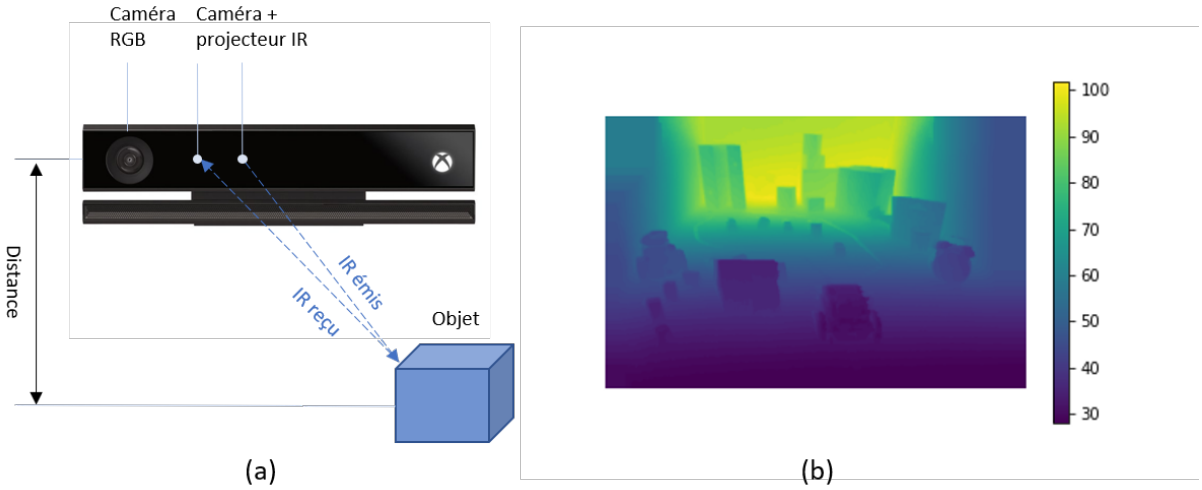


FIGURE 4.7 – a) Schéma de fonctionnement du capteur ToF d'une kinect, composé d'un projecteur et d'une caméra infrarouge (source image Kinect : fr.ifixit.com) b) Carte de profondeur avec une échelle en centimètres et en fausses couleurs.

Les cartes de profondeur constituent une source d'information supplémentaire dans la base de données. Par ailleurs, certains travaux en produisent pour la restauration d'images dans le brouillard. Peu de base de données de brouillard offrent la possibilité d'exploiter l'information sur la profondeur des images. Toutefois, il existe par exemple les bases de données NYU-Depth V2 (Silberman et al. (2012)) et Middlebury (Scharstein et al. (2014)) qui sont composées de vérités terrain et de cartes de profondeurs. Ces cartes de profondeur permettent également de générer des images contenant du brouillard de synthèse.
Notre base de données, en comparaison de celles de NYU-Depth V2 et Middlebury, fournit des images avec un brouillard réaliste, généré avec une machine à brouillard. Il est également possible de générer et d'ajouter du brouillard de synthèse sur nos images, grâce à la carte de profondeur.

4.2.2 Protocole expérimental

4.2.2.1 Création des vidéos avec et sans brouillard

L'utilisation d'un mobile permet d'obtenir des vidéos dynamiques. L'objectif de la création de cette base de tests est d'obtenir des paires de vidéos avec du brouillard et une vérité terrain. Cela implique que le robot doit avoir exactement la même position dans les deux scénarios et en tout point de la trajectoire. Le premier scénario envisagé consistait à imposer au robot une trajectoire complète, pour une densité de brouillard donnée et à faire une vidéo pour chaque densité de brouillard. Ce scénario est complexe à réaliser pour atteindre une bonne répétabilité. En effet, le robot ne reproduit jamais exactement la même trajectoire. La vérité terrain ne pouvait donc pas correspondre exactement aux vidéos enregistrées avec du brouillard. Nous avons donc utilisé la technique du *stop motion* où le mobile décrit sa trajectoire par intermittence. Ainsi, à chaque position du robot, des acquisitions de vidéos sans brouillard puis avec brouillard sont réalisées avec plusieurs cycles d'éclairages différents. La figure 4.8 schématise un cycle d'éclairage. Les traits de couleurs correspondent aux couleurs des LED de chaque bande. Lorsque les traits sont noirs, les LED sont éteintes. Les traits de couleurs jaunes, rouges et bleus correspondent respectivement aux couleurs de LED blanches, rouges et bleues.

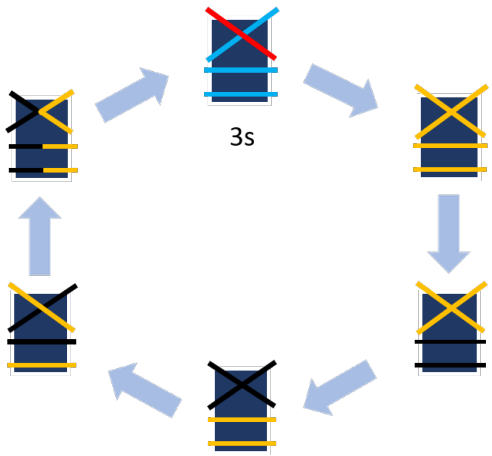


FIGURE 4.8 – Schéma d'un cycle d'éclairage. Chaque éclairage dure trois secondes.

Le temps d'acquisition de la vidéo avec brouillard laisse progressivement le temps au brouillard de s'évacuer pour obtenir différentes densités. La figure 4.9 présente le processus d'acquisition pour une seule position du robot. Cette timeline est reproduite pour chaque position du robot, pour environ 180 positions. Dans un premier temps, une vidéo de la scène sans brouillard est enregistrée avec les différents cycles d'éclairage. Ensuite, du brouillard est injecté dans le dispositif. Après quelques secondes d'attente de stabilisation du brouillard, l'acquisition de la vidéo commence en même temps que l'activation de la VMC. Pendant l'acquisition, le brouillard est évacué lentement pour obtenir des trames avec plusieurs densités de brouillard pour les différents éclairages (indiqué par la mention *N cycles d'éclairage* dans la figure 4.9). Soit un cycle d'éclairage illustré figure 4.8, N cycles d'éclairage correspondent à un nombre de cycle qui se déroule pendant les phases d'acquisitions. Comme le changement entre les phases est exécuté manuellement, le nombre de cycles n'est pas fixe. Par exemple, il est possible d'approximer le nombre N pour la phase d'acquisition des vidéos avec du brouillard. Si l'on considère que la deuxième phase d'acquisition dure 10 minutes et qu'un cycle dure 18 secondes, $N \simeq 33$ cycles.

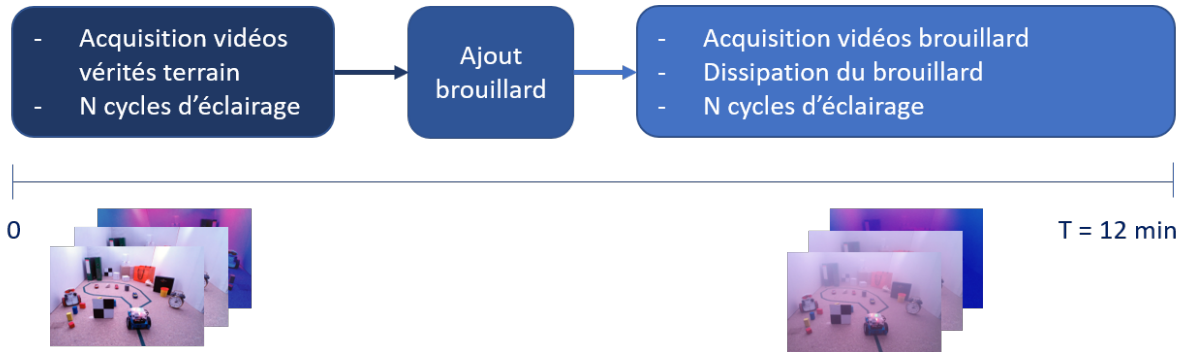


FIGURE 4.9 – Timeline du processus d'acquisitions des vidéos pour une position du robot

4.2.2.2 Les cartes de profondeur

Comme il est souvent le cas en imagerie 3D, il existe des artefacts dans les cartes de profondeur. Avec la technologie ToF, ces artefacts sont caractérisés par des pixels noirs représentant des données manquantes pour lesquelles le capteur ne renvoie pas de valeur. L'apparition de ces « trous noirs » peut être due aux occlusions, aux zones d'ombre ou aux objets réfléchissants. On retrouve également des erreurs au niveau des discontinuités de profondeur, en général autour des objets au premier plan. La figure 4.11a montre un exemple de carte de profondeur avec des erreurs. Des pixels noirs sont clairement visibles dans les coins supérieurs de l'image, ainsi qu'au niveau du contour des objets. Les pixels noirs dans les coins du dispositif seraient causés par des réflexions multiples dans les coins provenant du capteur, fournissant des données de profondeur incorrectes, voire inexistantes selon Kadambi et al. (2014) (voir figure 4.10).

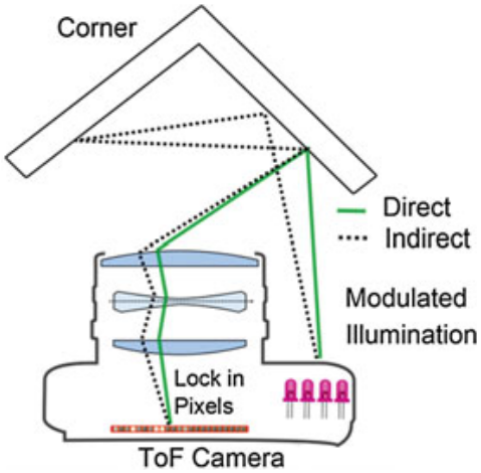
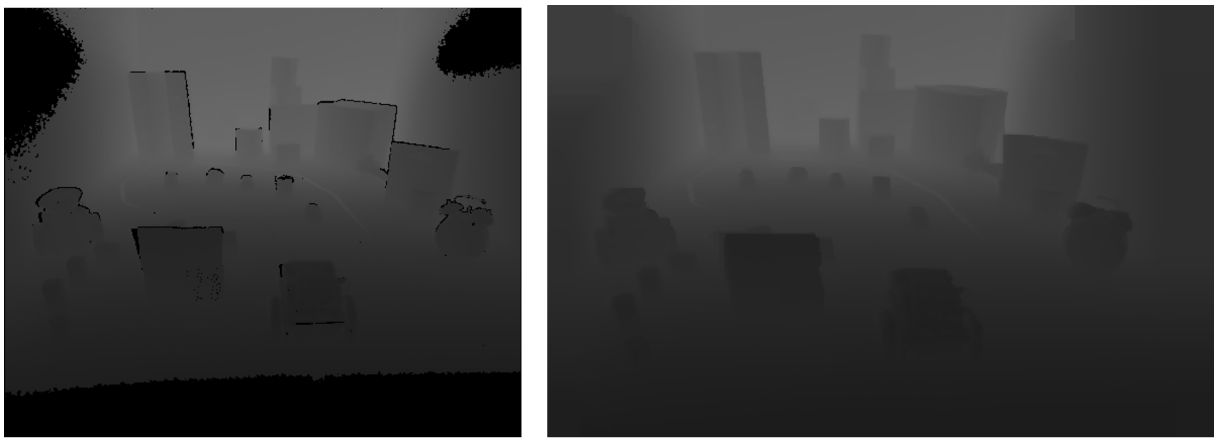


FIGURE 4.10 – Réflexions multiples dans les coins, d'après Kadambi et al. (2014).

Cet article cite plusieurs manières d'obtenir des cartes de profondeur denses, parmi lesquelles des méthodes de filtrage, de segmentation ou encore d'érosion morphologique des contours. Nous avons décidé d'utiliser un algorithme d'interpolation des cartes de profondeur par minimisation de l'énergie optimisée par l'algorithme Graph Cut. Le but de l'algorithme est d'interpoler la carte de profondeur uniquement aux pixels où la valeur est manquante. Soit D, cette carte de profondeur, et M le masque binaire des valeurs connues dans D. D doit être au préalable alignée avec une image I en niveaux de gris ou en couleurs sans pixel manquant. Des images en couleurs sont collectées à l'aide de la caméra RGB de la Kinect. Cependant, l'acquisition



(a) (b)

FIGURE 4.11 – Cartes de profondeur : a) Carte de profondeur avant interpolation b) Carte de profondeur après interpolation.

des données est réalisée sur des caméras de différentes résolutions et avec des angles de vue légèrement différents. Un calibrage géométrique précis entre les deux caméras est nécessaire. Cependant, la méthode classique des mires damiers n'est pas appropriée, puisque la caméra de profondeur ne peut pas restituer les contours. Quelques méthodes sont proposées pour pallier ce problème, mais nous avons décidé d'utiliser des paramètres intrinsèques par défaut, fournis par des utilisateurs de la Kinect.

Les images acquises par la caméra RGB sont utiles pour l'interpolation de D en favorisant les valeurs des pixels de profondeur proches de celles des pixels de couleurs dans un voisinage spatial. La carte de profondeur interpolée X est obtenue comme la solution de l'optimisation de l'énergie suivante :

$$\begin{aligned} X = argmin\Big(\sum p_M(i,j) - f \times \delta(D(i,j) = X(i,j)) + \sum p_X(i,j), \\ w_h(i,j)|X(i,j+1) - X(i,j)| + w_v(i,j)|X(i+1,j) - X(i,j)| \Big) \end{aligned} \tag{4.4}$$

où f est un facteur assez grand pour empêcher les valeurs des pixels connus dans D d'être modifiées et où les pondérations $w_h(i,j)$ et $w_v(i,j)$ sont des fonctions décroissantes vers zéro des différences $I(i,j+1) - I(i,j)$ et $I(i+1,j) - I(i,j)$ respectivement. $\sum p_M(i,j)$ et $\sum p_X(i,j)$ correspondent respectivement à la somme des pixels de coordonnées (i,j) dans M et dans X et δ est la fonction de Dirac. La figure 4.11b montre le résultat après application de l'algorithme d'interpolation. Nous obtenons bien une carte dense de même résolution que les images RGB.

4.2.3 Création des vidéos de la base de test

À la fin de l'expérience, 180 vidéos avec une position fixe du robot ont été créées. Certaines vidéos contiennent un brouillard qui s'évacue lentement, et d'autres sont sans brouillard. Chacune de ces vidéos a été acquise avec N cycles d'éclairages différents. Le but est d'obtenir des vidéos où le mobile se déplace pour une densité et un type d'éclairage donnés. Dans un premier temps, les éclairages ont été dissociés des vidéos à partir d'une extraction de trames. Au sein

d'une vidéo, les trames forment des lots consécutifs avec des éclairages différents (voir figure 4.12).

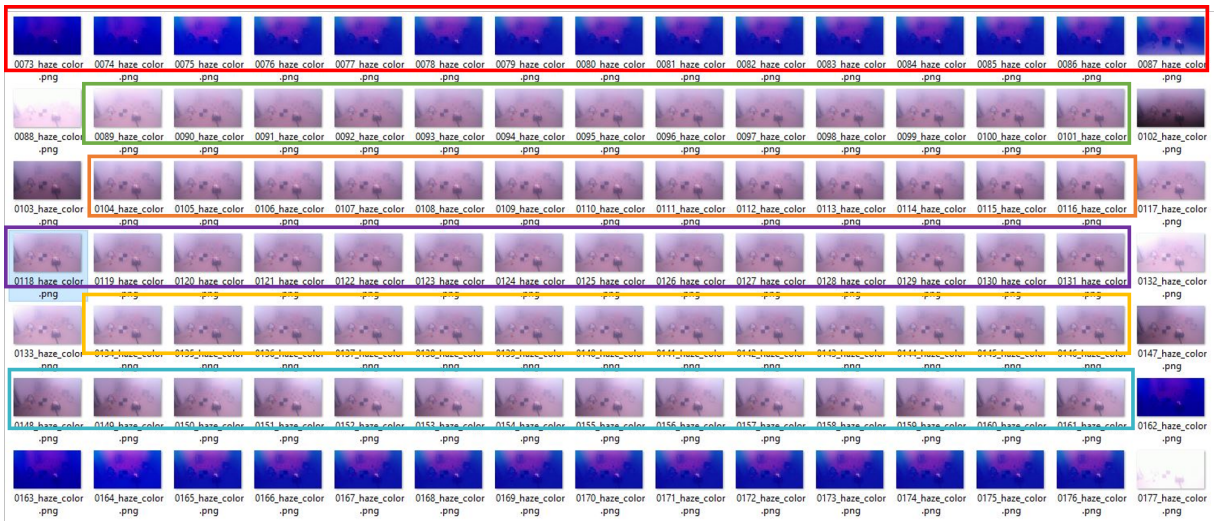


FIGURE 4.12 – Résultat d'une extraction de trames formant des lots consécutifs d'éclairages différents, à partir d'une vidéo acquise avec du brouillard. Les six jeux d'éclairages sont encadrés et correspondent à un cycle.

Les trames de transition sont généralement plus sombres ou plus lumineuses en raison du temps d'adaptation de la carte Arduino (latence) au changement de luminosité des rubans de LED. Pour les séparer, un calcul d'erreur quadratique moyenne a été réalisé entre deux trames consécutives. Si l'écart dépasse un certain seuil, nous considérons qu'il s'agit d'une transition et l'image n'est pas prise en compte. Ensuite, une séquence de trois images sous le même éclairage est moyennée afin de diminuer le bruit. Les images restantes ont été regroupées dans une matrice par type d'éclairage et ont été associées à une valeur de contraste, correspondant à la mire la plus proche. Ainsi, la figure 4.13 présente des images avec du brouillard de densité différente, de valeurs de contraste 0.015, 0.05 et 0.15 sur la mire. Plus la densité de brouillard est élevée, plus le contraste dans l'image est faible.

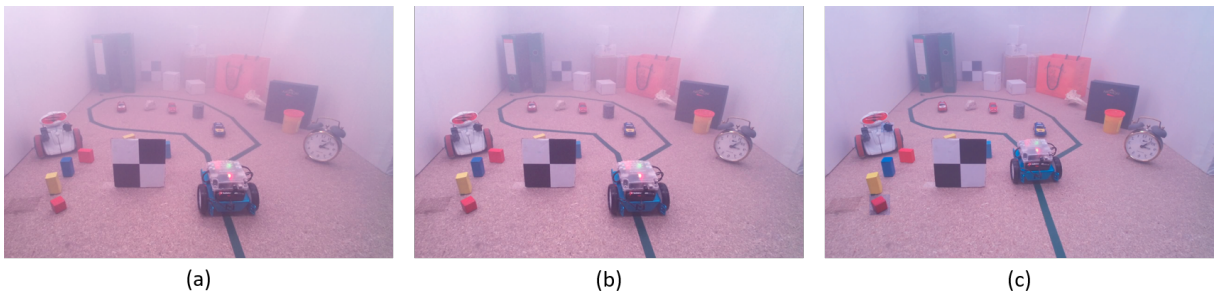


FIGURE 4.13 – Images avec des brouillards de différentes densités. Le contraste sur la mire la plus proche est de : a) c = 0.015, b) c = 0.05, c) c = 0.15.

La figure 4.14 présente une trame avec du brouillard et sa vérité terrain, pour six jeux d'éclairages différents. La colonne de droite indique les bandes de LED allumées. La première ligne correspond à l'éclairage de nuit (brouillard coloré) et toutes les autres à l'éclairage de jour.

FIGURE 4.14 – Trames contenant du brouillard et les vérités terrain associées pour chacun des six jeux d'éclairage.

4.2.4 Limites

4.2.4.1 Gestion des cycles d'éclairage avec une carte Arduino

Lors des acquisitions vidéo, les LED pilotées par une carte Arduino changent de couleurs et/ou d'état (allumé/éteint) toutes les trois secondes, conformément au code implémenté. Cependant, il est arrivé que l'un des schémas de couleur n'apparaisse pas dans un des cycles du scénario, et ce de manière aléatoire, faussant l'étape d'extraction automatique des trames selon le type d'éclairage. Le problème est survenu soit au niveau de la carte Arduino, soit lors de l'enregistrement des trames.

Le changement de couleur des LED induit des transitions d'intensités de couleur au sein d'un cycle. Ces transitions ont été très utiles pour extraire les images par jeux d'éclairage mais elles rendent aussi certaines images inexploitables (couleur différente, intensité trop élevée), comme le montre la figure 4.15. Ces transitions de couleur ont eu un impact sur la luminosité des images. De ce fait, avant l'étape de création des vidéos, une fonction de correction d'illumination a dû être appliquée afin de diminuer ces écarts de luminosité entre trames consécutives. Cette correction consiste à appliquer un facteur sur l'image courante (voir équation 4.5).

FIGURE 4.15 – Exemple de transition entre deux types d'éclairage.

$$f = \frac{mean(prec)}{mean(im)} \tag{4.5}$$

Avec $mean(prec)$ une moyenne sur l'image précédente et $mean(im)$ une moyenne sur l'image courante. L'équation de correction est alors définie par : $corr = f \times im$.

4.2.4.2 Inconvénient du *stop motion*

L'utilisation de la technique du *stop motion* implique beaucoup plus de temps et de rigueur pour l'acquisition de l'ensemble des vidéos. En effet, des sessions d'une douzaine de minutes se suivent, durant lesquelles il faut activer la machine à brouillard à un temps et pour une durée précise et lancer les acquisitions vidéo à la main. De plus, la batterie du mobile s'est déchargée plusieurs fois. Il a fallu la changer avec précaution afin d'éviter de bouger les éléments du décor. Le temps d'acquisition de l'ensemble des données a pris environ deux semaines sans tenir compte de la phase de prise en main et de test. La mise en place du dispositif et le traitement des données après les acquisitions ont pris également plusieurs semaines. Ce temps d'acquisition n'a pas permis de réaliser plus d'une scène. Réaliser ce processus sur plusieurs scènes aurait permis de créer une base de données plus importante et plus diversifiée.

Après avoir présenté la conception de la base de données VIREDA, je vais détailler la méthode de restauration de vidéos.

4.3 Élaboration d'une architecture Transformer-CNN pour une application de suppression de brouillard dans des vidéos

De nombreux travaux ont été proposés pour le traitement de séquences vidéos, certains sont basés sur des réseaux de neurones récurrents (voir la sous-section 2.2.2.3) et d'autres sur des *Transformers* (voir la sous-section 2.2.2.5). Les méthodes basées sur des architectures *Transformers* « purs » semblent surpasser de nombreuses méthodes, dont celles basées sur les architectures CNN et RNN en terme de performance. Seulement, pour être réellement efficace, les *Transformers* ont besoin d'une grande quantité de données d'apprentissage et ils sont coûteux en mémoire. En revanche, les architectures basées sur des réseaux de neurones récurrents peinent à assurer les dépendances sur le long terme au sein de séquences de données et ne permettent pas la parallélisation. Finalement, j'ai décidé d'utiliser une architecture *Transformer* pour composer les blocs temporels. Une évaluation appelée *Ablation study* est réalisée afin de vérifier la pertinence de ce choix. D'après Zhang et al. (2020b), la corrélation temporelle dans une vidéo est mieux exploitée dans le cadre d'une méthode multi-étages plutôt que par des méthodes qui n'agrègent les caractéristiques temporelles qu'une seule fois à une étape déterminée du réseau. J'ai également décidé d'adopter la même stratégie.

La méthode multi-étages, appelée TCVD (*Transformer-CNN architecture for Video Defogging*), permet de traiter les informations spatio-temporelles des séquences vidéos. À chaque étage, les données spatiales et temporelles sont traitées dans des blocs différents et sont ensuite fusionnées à la fin de ces étapes. La méthode est dite hybride, car elle associe deux architectures différentes : les CNNs pour traiter l'information spatiale et les *Transformers* pour assurer la cohérence temporelle. L'architecture globale du réseau de neurones a été inspirée de U-net, un des réseaux les plus utilisés et largement adapté à de nombreuses applications de restauration d'images. Une évaluation sur différentes bases de données est menée pour s'assurer de l'efficacité de l'algorithme. Elle est comparée à des méthodes traditionnelles ainsi que des méthodes de *deep learning*.

4.3.1 Architecture U-net

Pour capturer l'information spatiale des images de la séquence, je me suis inspirée de l'architecture du réseau U-Net, de Ronneberger et al. (2015).

U-net est une architecture auto-encodeur qui permet d'extraire les caractéristiques principales des images (voir schéma 4.16). Elle est composée de deux réseaux : un encodeur et un décodeur. L'encodeur apprend à compresser l'image originale en un petit ensemble de caractéristiques encodées, tandis que le décodeur apprend à restaurer l'image d'origine à partir des caractéristiques générées par l'encodeur. Le but est de reconstruire les données d'origine le plus exactement possible.
La figure 4.17 présente un exemple de schéma de cette architecture dans le cadre d'une application de suppression de bruit dans des images de chiffres. Des images en entrée labellisées de la base de données *Mnist* sont injectées dans l'encodeur avec du bruit. Les caractéristiques les plus importantes des données en entrée y sont extraites et traduites de manière compréhensible par le décodeur. La partie nommée *code* dans le schéma représente le codage le plus compressé. Symétriquement à l'encodeur, le décodeur va lire le code et produire des images restaurées sans bruit, basées sur ces informations.

Il s'agit d'un réseau symétrique composé de couches de convolutions utilisé à l'origine pour de la segmentation sémantique médicale. U-net est un des réseaux les plus utilisés et a

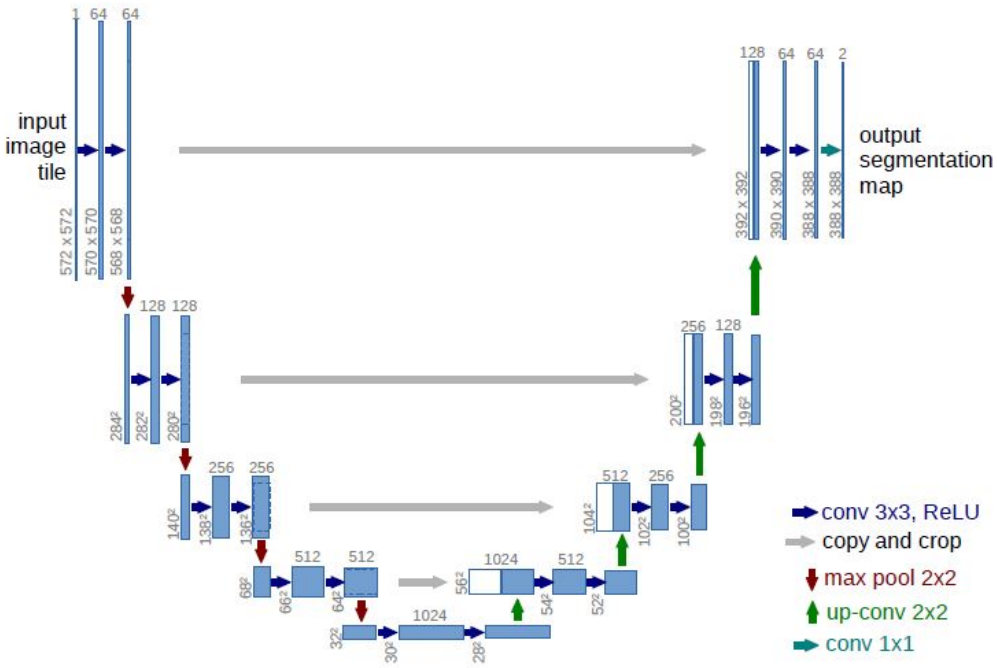


FIGURE 4.16 – Architecture neuronale du réseau U-net.

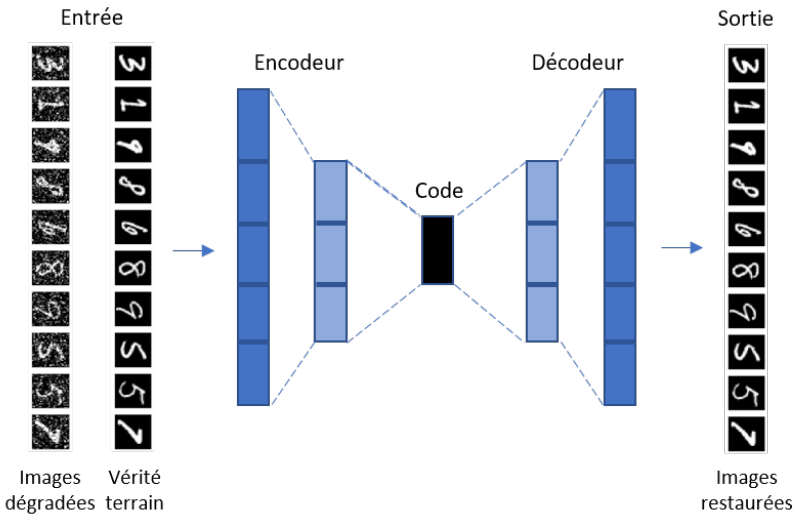


FIGURE 4.17 – Architecture neuronale de l'auto-encodeur : application à la suppression du bruit dans les images.

été largement repris et adapté à de nombreuses tâches, dont celle de la restauration d'images. En effet, plusieurs chercheurs s'en sont inspirés pour les tâches de traitement d'atténuation du brouillard, comme Ge et al. (2021). Dans la suite de ce chapitre, nous utilisons le réseau U-net comme point de départ de notre approche. Une série de modifications vont être réalisées afin de l'adapter à notre méthode de restauration de vidéos.

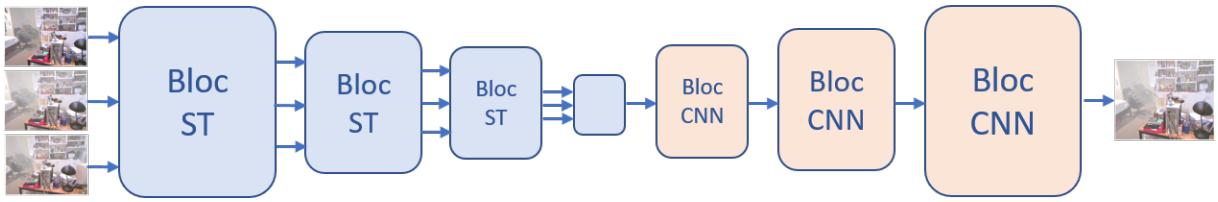


FIGURE 4.18 – Schéma de l'architecture de la méthode proposée. Les blocs ST correspondent aux blocs spatio-temporels détaillés dans la sous-section suivante.

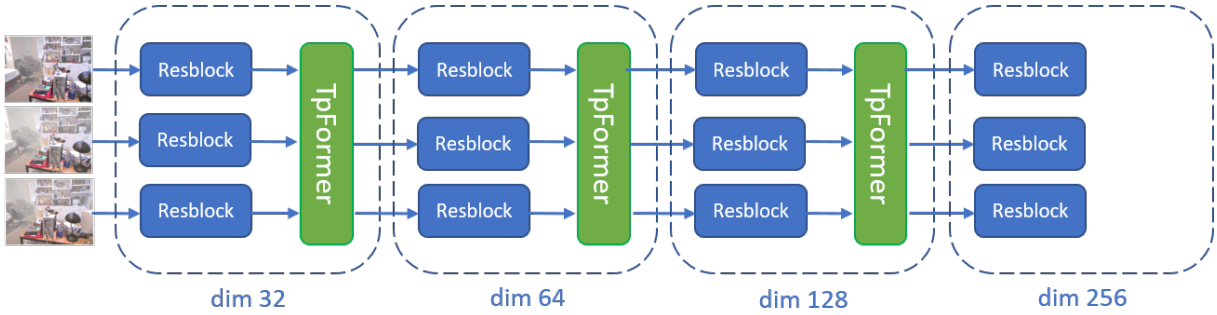
4.3.2 Un algorithme hybride

L'utilisation de l'architecture *Transformer* semble appropriée pour traiter des séquences de données en raison de sa capacité à lier les données au sein d'une séquence et à paralléliser les tâches. Cependant, le *Transformer* possède quelques inconvénients dont un faible biais inductif. Ce dernier représente l'ensemble des hypothèses faites par le modèle dans sa capacité à apprendre et généraliser au delà des données d'apprentissage. De ce fait, un modèle basé sur cette architecture a nécessité une très grande quantité de données d'apprentissage pour être réellement efficace. Les CNNs, quant à eux, ont un fort biais inductif, permettant au modèle d'atteindre des performances intéressantes avec moins de données d'apprentissage. En revanche, ce type de réseau de neurones ne permet pas de capturer l'interdépendance des images au sein d'une séquence vidéo.

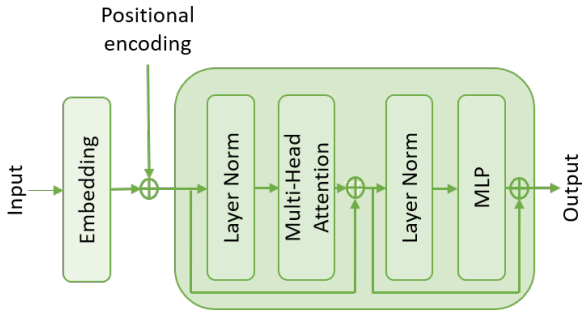
Un autre aspect de l'utilisation conjointe de ces deux architectures est la source d'information globale et locale dans la séquence d'images. L'idée principale est d'utiliser la capacité de modélisation globale du *Transformer* et locale du CNN pour retirer un maximum d'information. Afin de bénéficier des avantages des deux techniques, j'ai décidé de les combiner en utilisant une architecture CNN comme extracteur de caractéristiques et une architecture *Transformer* pour préserver la cohérence temporelle au sein d'une séquence d'images. Un schéma de l'architecture globale de la méthode proposée est présentée à la figure 4.18.

4.3.3 L'encodeur

La partie encodeur de notre architecture se compose de quatre modules CNN et de trois modules basés sur une architecture *Transformer*, appelés **TpFormer**. Le schéma d'architecture de l'encodeur est illustré par la figure 4.19. Chaque étape du processus, détaillée à la figure 4.19b, correspond à l'extraction des *features* de chaque image de la séquence, et à leur fusion à l'entrée du bloc **TpFormer** à plusieurs dimensions (filtres des couches de convolution 2D : 32, 64, 128 et 256). À l'issue des étapes d'extraction de caractéristiques, les triplets d'images successives dans la vidéo sont concaténés le long de l'axe temporel, puis chaque image est divisée en patches superposés. Ces patches constituent l'entrée du module **TpFormer**.
Le module **TpFormer**, détaillé à la figure 4.20, est composé de différentes couches dont la couche *Multi-Head-Attention* (MHA), détaillée dans le chapitre 2 à la sous-section 2.2.2.5. Pour rappel, elle permet de réaliser différents calculs d'attention en parallèle. L'utilisation de ce mécanisme permet au réseau d'apprendre la dépendance séquentielle entre les trames. Les caractéristiques des trois trames résultantes sont ensuite dissociées et concaténées aux caractéristiques spatiales résultantes du bloc CNN de la même étape. Les informations fusionnées constituent l'entrée de l'étape suivante.
Dans ce chapitre, nous allons traiter un type d'attention en particulier : le mécanisme de *Self-Attention*. Ce procédé, utilisé dans le module MHA, permet d'établir des connexions au sein



(a)



(b)

FIGURE 4.19 – Architecture de l'encodeur de la méthode proposée : (a) schéma global de l'architecture de l'encodeur, (b) détails d'une étape de l'encodeur.

d'une même séquence.

Nous allons voir dans la section suivante comment ce principe a été étendu aux images.

4.3.4 Le module TpFormer

Le module **TpFormer**, pour *TemPoral TransFormer*, est détaillé à la figure 4.20. Il est composé des couches MHA, d'un réseau *feed forward* pour le calcul des caractéristiques d'attention temporelle, et de couches *Layer-Normalisation* pour aider à la stabilisation du réseau.

4.3.4.1 Les données d'entrée du module TpFormer

Les séquences d'images ne sont pas directement compréhensible par les *Transformers*, en particulier par la couche MHA. Les données doivent au préalable être transformées en vecteur. La première étape consiste à diviser les images de la séquence en imagettes (ou patches) et à les « aplatir » sous forme de vecteur dans un sous-espace d'intégration (*embedding* en anglais). Il s'agit du processus de *tokenisation*, introduit par Dosovitskiy et al. (2021). Il existe plusieurs méthodes pour diviser les images de la séquence en imagettes. Une des méthodes possibles consiste à extraire des imagettes de chaque image de la séquence indépendamment. L'information temporelle est ensuite capturée lors de la phase d'intégration dans le sous-espace d'intégration. J'ai décidé d'exploiter directement la temporalité de la séquence en utilisant la méthode *Tubelet embedding* de Arnab et al. (2021). Cette méthode permet d'extraire directement des volumes spatio-temporels de la séquence d'entrée (ici trois trames successives) de dimension $T \times H \times W$ par l'intermédiaire d'un calcul de convolution 3D. Ces volumes contiennent les imagettes extraites de la séquence ainsi que l'information temporelle de celle-ci. L'ensemble

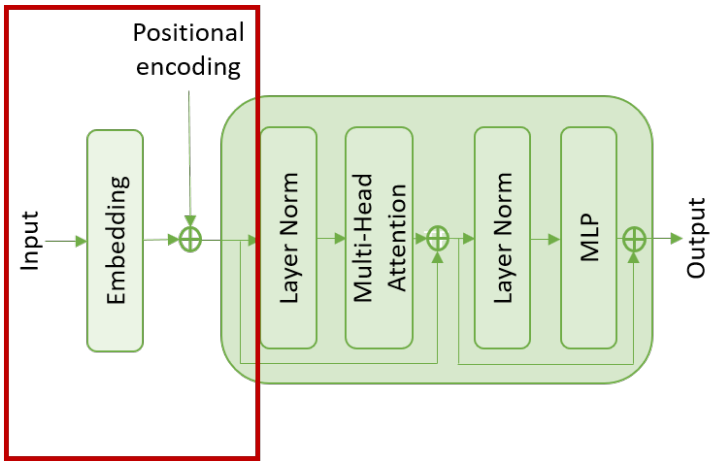


FIGURE 4.20 – Détails du module **TpFormer**. L'encadré rouge correspond à la zone d'intérêt de ce paragraphe.

est ensuite converti en un vecteur à une dimension. La figure 4.21 présente un schéma de la méthode.

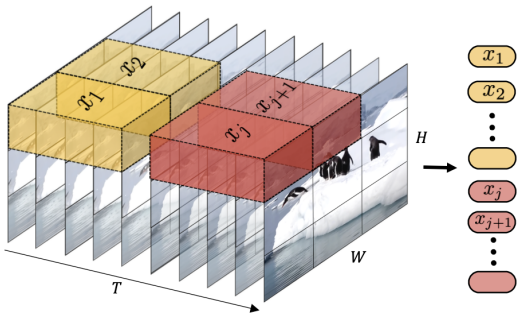


FIGURE 4.21 – Schéma de la méthode *Tubelet embedding*, d'après Arnab et al. (2021). x_1 et x_2 correspondent à des séquences d'imagettes 3D.

L'étape suivante du processus de transformation en données compréhensible par la couche MHA, est le codage positionnel (*positional encoding* en anglais) qui consiste à « sauvegarder » la position des imagettes au sein de la séquence d'images. En effet, il est nécessaire de conserver l'ordre des images dans la séquence. Ces étapes sont résumées dans le schéma de la figure 4.22.

Le codage positionnel donne la position d'une entité dans la séquence de sorte que chaque position se voit attribuer une représentation unique. Plus précisément, chaque imagette est transformée (par l'intermédiaire de couches *embedding*) en vecteurs de faible dimension et de longueur définie (un nombre de filtre égal à 32, 64 ou 128), afin de mieux représenter les différentes entités. Une position est ensuite attribuée à chacune des imagettes de la séquence vidéo.

Transformer des triplets d'images à plusieurs dimensions en vecteurs comme entrée du module MHA est une étape très coûteuse en temps de calcul. Pour remédier à cela, les images de la séquence sont au préalable sous-échantillonnées en images 28×28. La taille des imagettes que j'ai choisies est de 4×4 pixels.

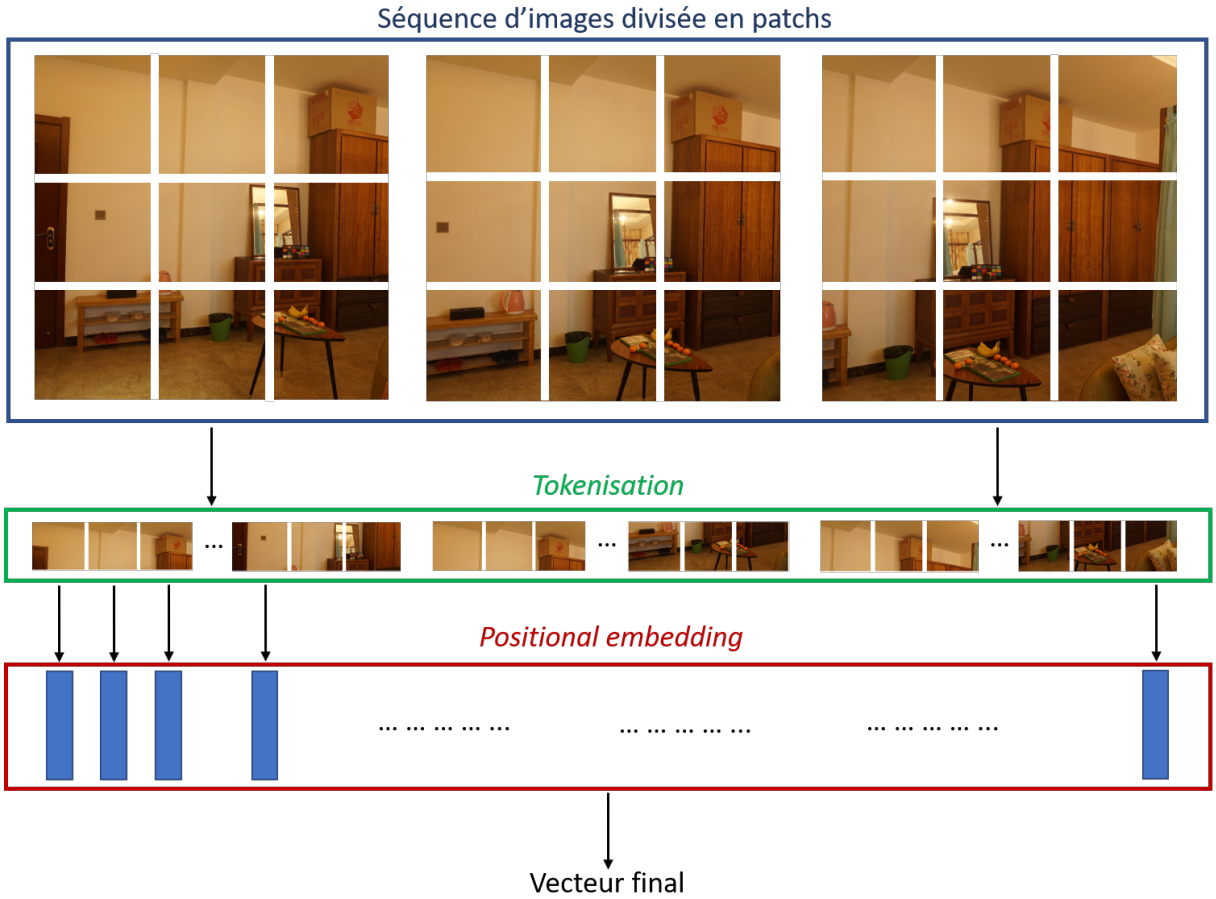


FIGURE 4.22 – Exemple de schéma d'encodage d'un triplet d'image en un vecteur compréhensible par la couche MHA. La division des images a été réalisée pour la compréhension du schéma.

4.3.5 Le décodeur

Une fois que l'encodeur a extrait l'ensemble des caractéristiques spatio-temporels des images de la séquence, la partie décodeur va, symétriquement à l'encodeur, sur-échantillonner les données encodées de l'image centrale, de manière à obtenir une image de dimension égale à celle des images en entrée du réseau neuronal. Pour sur-échantillonner, des couches Conv2DTranspose (Keras) ont été utilisées, ainsi que des sauts de connexions (voir la sous-section suivante) depuis l'encodeur pour conserver les détails fins des images.

4.3.6 Les éléments essentiels au bon fonctionnement du modèle

4.3.6.1 Les sauts de connexions

Les sauts de connexions, (*skips connections* en anglais), correspondent à diverses connexions au sein d'un réseau neuronal. Certaines couches sont ignorées pour établir des connexions avec la sortie d'autres couches comme entrée des couches suivantes.
Il existe deux types de sauts de connexions : les longs et les courts. Le premier est caractéristique des architectures symétriques, telles que U-net. Introduire ces connexions entre l'encodeur et le décodeur permet de conserver de l'information haute fréquence (détails fins dans les images)

qui aurait pu être supprimée pendant l'étape de sous-échantillonnage. Les sauts de connexions courts sont utilisés entre des couches de convolutions consécutives, dont la dimension d'entrée ne change pas (voir le paragraphe suivant). Ils semblent aider à la stabilisation du gradient dans les architectures profondes. L'importance de ces connexions a été largement prouvé. Li et al. (2018b) ont présenté des travaux permettant la visualisation de la surface de coût (*loss surface* en anglais) avec et sans les sauts de connexions. La figure 4.23 reprend ce schéma de l'article.

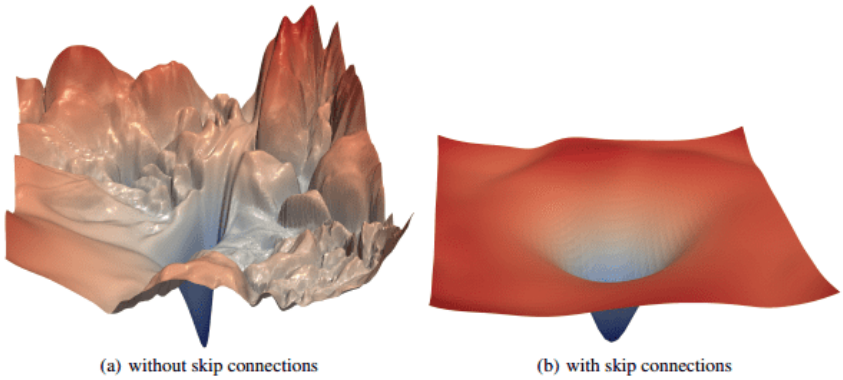


FIGURE 4.23 – Visualisation haute-résolution des surfaces de perte de l'architecture ResNet-56 (He et al. (2016)) avec et sans sauts de connexions. L'image provient de l'article de Li et al. (2018b).

La figure 4.23a représente une fonction de coût non convexe en trois dimensions, présentant plusieurs minimums locaux. Avec ce type de fonction, l'algorithme de descente de gradient risque de rester bloquer dans n'importe quel minimum local. La figure 4.23b représente une fonction de coût convexe en trois dimensions, lorsque des sauts de connexions sont utilisés, avec un unique minimum global. D'après l'article de Li et al. (2018b), cette représentation ne prouve pas seulement l'importance de l'utilisation des sauts de connexions, mais également l'impact du rôle de n'importe quel paramètres (le taux d'apprentissage, l'optimiseur, etc...) dans le processus de minimisation de la fonction de coût. L'ensemble des choix des paramètres doit conduire, selon l'utilisation, à un paysage (terme utilisé dans l'article) plus ou moins lisse, correspondant au niveau de généralisabilité de la méthode.

4.3.6.2 L'utilisation de blocs résiduels

Les réseaux de neurones résiduels (comme Res-Net) utilisent des sauts de connexions entre différentes couches du réseau en sautant des couches intermédiaires. L'utilisation de ces connexions permet de simplifier le réseau et peut résoudre les problèmes d'explosion et de disparition du gradient. Comme je l'ai précisé dans le paragraphe précédent avec la figure 4.23, en l'absence de connexions, l'algorithme de descente de gradient rencontre plus de risque de se bloquer dans un minimum local.

La figure 4.24 illustre la composition des blocs résiduels qui composent la partie encodeur du modèle présenté dans ce chapitre. ReLU est une fonction d'activation non linéaire qui signifie *Rectified Linear Unit* en anglais. Elle permet de ramener les résultats négatifs à zéro à chaque sortie d'une couche où elle est utilisée. *Max Pooling* est une couche qui permet de sous-échantillonner les images dans l'encodeur.

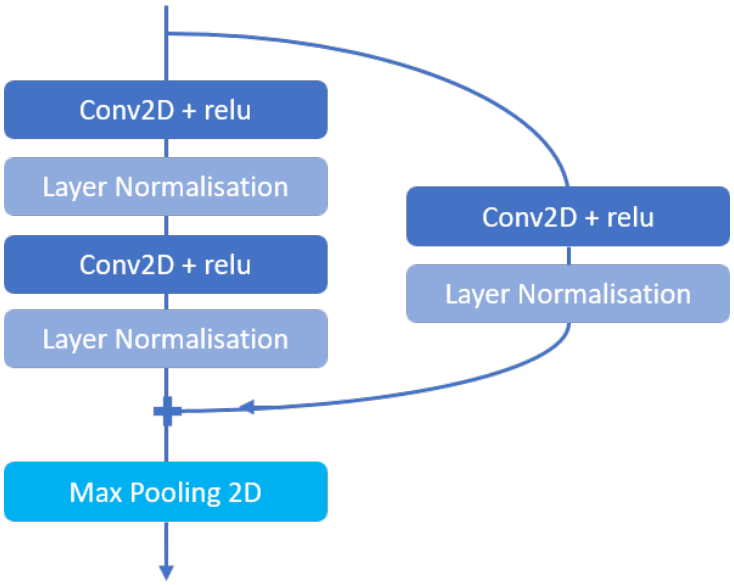


FIGURE 4.24 – Composition des blocs résiduels de l'encodeur.

4.3.6.3 Les techniques de normalisation

La Batch-normalisation (BN) est une technique de régularisation permettant de rendre un réseau de neurones plus stable, et donc plus facilement entraînable, par la normalisation de la sortie des couches à l'intérieur de ce réseau. Le terme « batch » normalisation indique que la normalisation est effectuée sur un lot de données, appelé batch en anglais. Malgré les bénéfices de cette technique (régularisation, stabilisation du réseau, ...), la marge d'erreur du BN augmente rapidement lorsque la taille du batch décroît, en raison d'une estimation inexacte des statistiques du batch, d'après Wu and He (2018). Ce qui pose question pour les applications qui demandent des petites tailles de batch en raison de la consommation élevée en mémoire. Par exemple, le réseau entraîné qui fait l'objet de ce chapitre ne peut pas excéder une valeur de batch de 4. D'autres alternatives existent au BN pour faire face à ce type de contrainte (voir la figure 4.25).

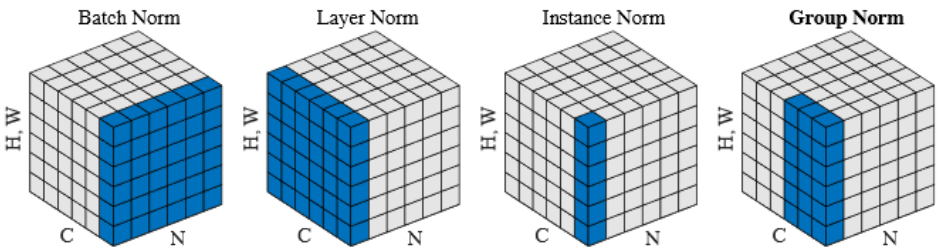


FIGURE 4.25 – Illustrations des différentes méthodes de normalisations, d'après Wu and He (2018). N correspond à l'axe du batch, C a l'axe du canal et (H,W) aux axes spatiaux.

D'autres méthodes, telles que les méthodes de Layer-normalisation, Instance-normalisation et Group-Normalisation, évitent la normalisation le long de la dimension du batch. Afin d'optimiser les performances du modèle, j'ai donc choisi d'utiliser la méthode de Layer-normalisation.

4.3.7 Implémentation

Notre méthode a été implémentée sur Google Colab Pro. Colab correspond à un environnement du notebook Jupyter et fonctionne dans le Cloud. Le plus pratique est qu'il ne nécessite aucune configuration et qu'il prend en charge les bibliothèques d'apprentissage automatique. Des ressources nous sont allouées et nous permettent d'avoir accès à des GPU (par exemple, avec la carte graphique Tesla K80) et TPU, ainsi qu'à une certaine quantité de mémoire RAM. Cependant, il existe des limites à son utilisation. Comme l'accès aux GPU est partagé entre de nombreux utilisateurs, la limite d'accès n'est pas garantie et le notebook se déconnecte au maximum après 12h d'utilisation. Colab me permet tout de même d'avoir accès aux GPU et à une mémoire de 50 gigas.

L'implémentation des algorithmes a été réalisée avec les frameworks Keras/Tensorflow. Nous utilisons l'optimiseur ADAM et un taux d'apprentissage de 0.0001. Le réseau est formé avec un total de 100 époques et une taille de batch de 4. Les limitations de la capacité mémoire, combinée à la taille de la base d'apprentissage et l'utilisation coûteuse des *Transformers* ne nous permettent pas d'augmenter la taille du batch. La fonction Loss est la combinaison entre le calcul de distance $L1$ et le calcul de $SSIM$, introduit dans l'état de l'art à la section 2.2.5. La fonction est définie comme suit :

$$Loss = \delta L_{SSIM}(P) + \gamma L_1(P) \tag{4.6}$$

où δ et γ sont des coefficients de pondération que j'ai choisis empiriquement et P une image. Le choix de cette fonction permet de conserver à la fois la luminance, les couleurs et le contraste dans les hautes fréquences.

4.4 Évaluation

Dans le cadre d'évaluations quantitatives et qualitatives, les méthodes de restauration d'images et de vidéos MFP et $TCVD$ (la méthode développée dans ce chapitre) seront confrontés à d'autres sur les bases de données REVIDE, NYU-Depth v2 et VIREDA. Nous les confronterons notamment avec les méthodes DCP (He et al. (2011)), FFA-Net (Qin et al. (2019)) et CG-IDN (Zhang et al. (2022)). Comme peu de codes associés aux méthodes d'atténuation du brouillard dans des vidéos sont disponibles publiquement, l'évaluation qualitative sera réalisée avec des méthodes dédiées aux images.

4.4.1 Évaluation qualitative

Dans cette section, une évaluation de la qualité des trames vidéos restaurées est réalisée. Parmi les vidéos contenant du brouillard, nous distinguons les vidéos acquises en intérieur avec une machine à brouillard (REVIDE, VIREDA) et les vidéos acquises en conditions réelles, c'est-à-dire en extérieur avec du vrai brouillard.

La figure 4.26 présente les résultats obtenus après restauration des trames vidéos appartenant à la base de données REVIDE, avec différentes méthodes. Si la méthode *DCP* améliore la visibilité des objets, elle modifie les couleurs et assombrit l'image par rapport à l'image recherchée. Avec la méthode MFP, le brouillard n'est pas assez atténué. Ce phénomène est potentiellement dû à la raison évoquée dans la partie évaluation quantitative, concernant l'estimation de la carte de transmission. Les méthodes FFA-*Net* (Qin et al. (2019)), *CG-IDN* (Zhang et al. (2022)) et

FIGURE 4.26 – Comparaison de la qualité visuelle de trames vidéo restaurées avec différentes méthodes.

TCVD obtiennent des résultats satisfaisants. Mais la méthode *FFA-Net* ne semble pas efficacement atténuer le brouillard de l'image de la première ligne. La couleur du brouillard semble en être la cause.

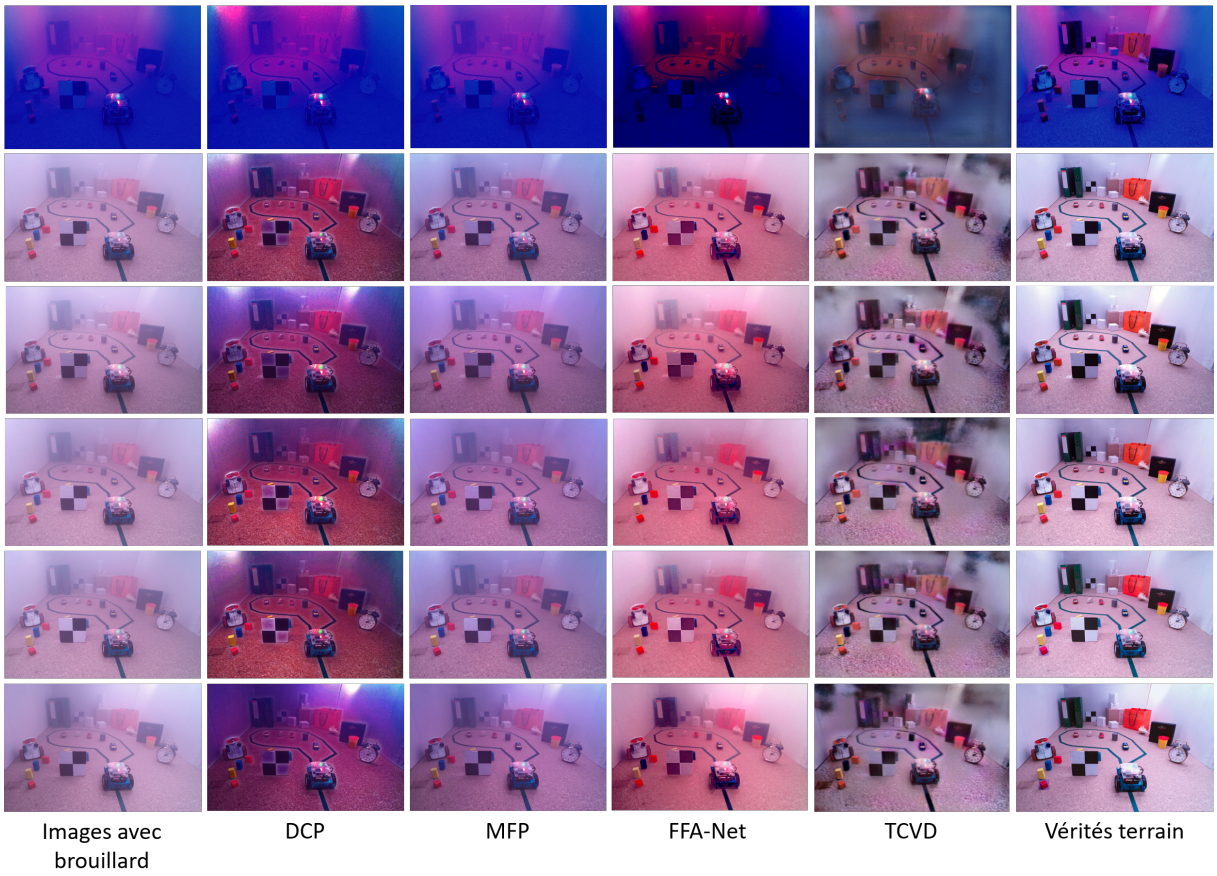


FIGURE 4.27 – Restauration de trames vidéo de la base de données VIREDA avec une densité de brouillard de 0.05.

Les figures 4.27 et 4.28 présentent les résultats obtenus avec différentes méthodes d'atténuation du brouillard sur la base de tests VIREDA. Chaque ligne de la figure correspond à un type d'éclairage différent. La figure 4.27 se caractérise par un brouillard de densité égale à la valeur de contraste c = 0.05 tandis que la figure 4.28 de densité égale à la valeur de contraste c = 0.15. Dans les deux cas, la méthode *DCP* augmente trop les contrastes en particulier dans les zone blanches et les couleurs sont trop saturées par rapport à l'image de référence. Les mé-

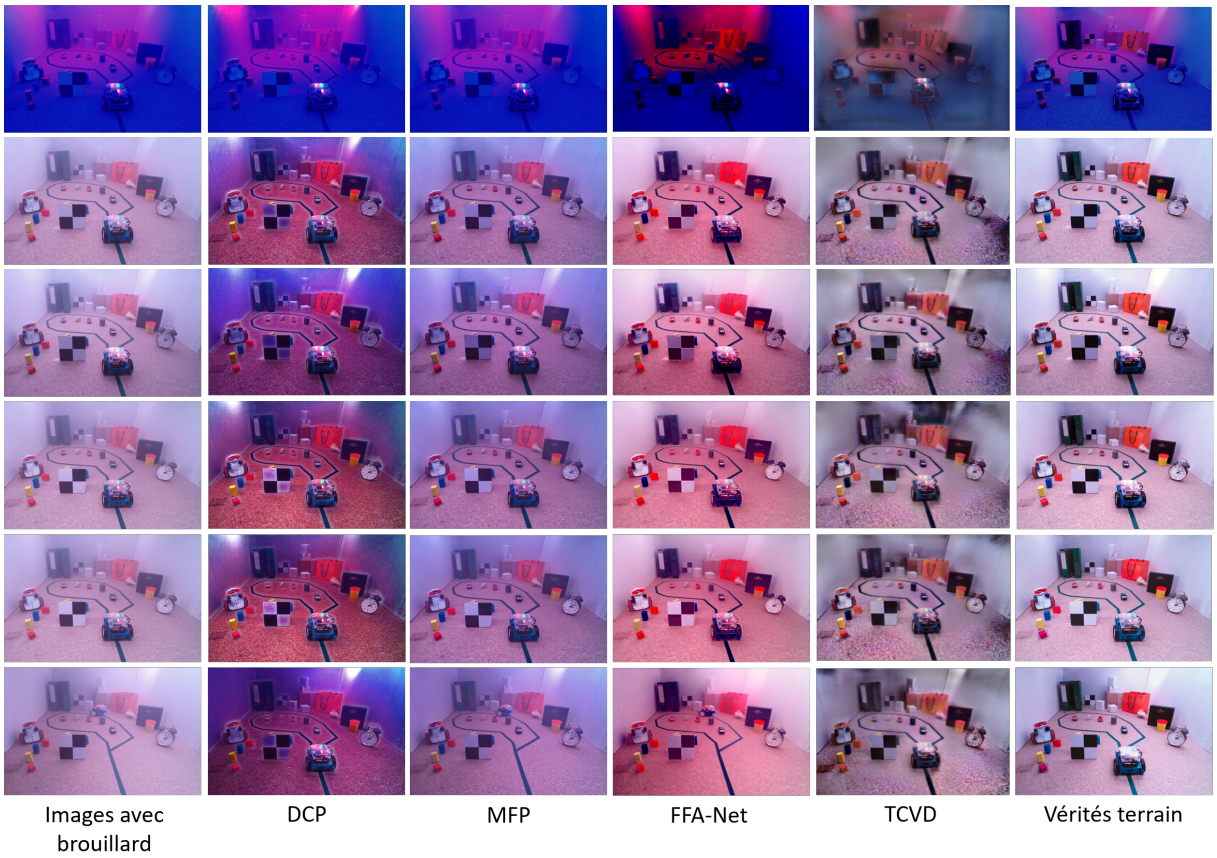


FIGURE 4.28 – Restauration de trames vidéo de la base de données VIREDA avec une densité de brouillard de 0.15.

thodes *MFP* et *FFA-Net* atténuent peu le brouillard dense de la figure 4.27, mais produisent des résultats plus satisfaisants lorsque le brouillard est plus léger. La méthode *TCVD*, destinée aux vidéos, atténue bien le brouillard mais laisse des artefacts sombres dans les zones blanches de l'image. Cet effet peut être dû aux zones ponctuelles de lumière introduit par l'éclairage. L'algorithme interprète peut-être ces zones de lumière comme étant du brouillard à supprimer, ce qui entraîne une sur-restauration dans ces zones. L'éclairage artificiel utilisé pour créer les vidéos de la base de données a produit des images légèrement coloré malgré l'usage de LED blanches. Cet éclairage induit un brouillard coloré qui semble perturber le processus de restauration en introduisant des artefacts et décalages colorés.

La première ligne de la figure 4.29 présente une séquence d'images que j'ai prise avec une caméra sur le toit du bâtiment par temps de brouillard. La seconde ligne correspond aux résultats obtenus après la restauration des images de la première ligne avec la méthode *TCVD*. Le brouillard est bien atténué au niveau de l'horizon, permettant de mieux voir les bâtiments, la végétation ainsi que les grues sur la troisième image. On peut remarquer de légères variations de couleur orangée à la limite entre le ciel et les bâtiments, conséquence potentielle de la présence de pollution.

4.4.2 Évaluation quantitative

Afin d'évaluer objectivement la méthode d'atténuation de brouillard dans des vidéos, celle-ci va être comparée à d'autres méthodes sur différentes bases de données. D'autres méthodes

FIGURE 4.29 – Restauration de trames vidéo avec du brouillard réel à l'aide de la méthode TCVD.

d'atténuation du brouillard dans des images et des vidéos de l'état de l'art vont faire partie de nos comparaisons et figurent dans les tableaux de cette section. Cependant, peu de codes associés aux méthodes d'atténuation du brouillard dans des vidéos sont disponible publiquement. En conséquence, pour l'évaluation sur la base de données REVIDE, j'ai repris les résultats de l'article de Zhang et al. (2022), créateurs de la base de données. Pour cette évaluation, j'ai ajouté une variante de l'algorithme *MFP*, appelée *MFP-vid*. Elle correspond à la méthode *MFP* en version dynamique appliquée aux vidéos. Pour la réaliser, un filtrage médian a été appliqué sur des séries de cinq trames avec l'objectif d'obtenir une atténuation des effets de scintillements et des artefacts visuels provoqués par une restauration traditionnelle trame par trame.

TABLE 4.2 – Évaluation quantitative sur la base de test REVIDE repris de l'article de Zhang et al. (2022). Les noms en bleu correspondent aux méthodes que j'ai réalisées. Les métriques en gras correspondent aux meilleurs résultats.

Métriques	DCP	**MFP**	**MFP-vid**	FFA-Net	VDN	EDVR	CG-IDN	**TCVD**
SSIM	0.7285	0.70	0.69	0.8133	0.8133	0.8707	0.8836	**0.95**
PSNR	11.03	17.59	17.54	16.65	16.64	21.22	**23.21**	20.97

Le tableau 4.2 présente des résultats avec les métriques SSIM et PSNR des modèles entraînés sur l'ensemble de données REVIDE et évalués sur la base de tests du même nom. Cette base de tests contient six vidéos de 36 à 58 trames chacune. Les résultats des méthodes *DCP* (He et al. (2011)), *FFA-Net* (Qin et al. (2019)), *VDN* (Ren et al. (2019)), *EDVR* (Wang et al. (2019b)) et *CG-IDN* (Zhang et al. (2022)) ont été repris de l'article de Zhang et al. (2022). D'après ce tableau, les méthodes *DCP*, *MFP*, *FFA-NET* et *VDN* obtiennent des résultats moins satisfaisants que les méthodes *EDVR*, *CG-IDN* et *TCVD*. Les méthodes *DCP*, *MFP* et *VDN* incluent une

étape d'estimation de la carte de transmission dans leur processus de restauration. Or, d'après Zhang et al. (2022), les cartes de transmission d'images réelles sont plus complexes à estimer que les cartes de transmission d'images avec du brouillard de synthèse et peuvent engendrer davantage d'erreurs. Notre méthode *TCVD* obtient une valeur de SSIM égale à 0.95. Ce résultat surpasse les autres mais est probablement l'effet d'un léger sur-apprentissage. En contrepartie, la valeur de PSNR obtenue est moins satisfaisante que celle des méthodes *EDVR* et *CG-IDN*.

Enfin, il est également important de prendre en compte le nombre de paramètres des algorithmes de *deep learning*. L'algorithme *CG-IDN* a au minimum 21,6 millions de paramètres alors que le nombre de paramètres des méthodes *FFA-Net* et *TCVD* est de l'ordre des 4 millions. Le nombre de paramètres pour les méthodes *EDVR* et *VDN* ne sont pas indiquées.

Le nombre de paramètres a un impact sur les performances finales, et souvent, plus son nombre est élevé et plus le modèle est exposé au risque de sur-apprentissage.

TABLE 4.3 – Évaluation quantitative sur la base de tests VIREDA.

métrique	densité	DCP	MFP	MFP-vid	FFA-Net	TCVD
SSIM	0.015	0.75	0.59	0.63	0.67	**0.76**
	0.05	0.60	**0.78**	0.71	0.67	**0.78**
	0.15	0.64	0.82	0.77	0.73	**0.84**
PSNR	0.015	10.11	**15.69**	13.38	14.26	11.81
	0.05	10,43	**15.78**	15.25	14.47	12.80
	0.15	11.50	16.40	**16.72**	16.26	13.82
FSIM	0.015	0.55	0.84	0,85	0.52	**0.90**
	0.05	0.62	**0.90**	0.89	0.53	**0.90**
	0.15	0.66	**0.93**	**0.93**	0.58	0.92

Le tableau 4.3 présente une comparaison des performances des algorithmes avec les métriques SSIM, PSNR et FSIM, sur la base de test VIREDA pour les trois densités de brouillard : 0.015, 0.05, 0.15. Les méthodes *DCP* et *MFP* conservent le même jeux de paramètres que dans le chapitre 3. La méthode *TCVD* a de meilleurs performances en terme de métrique SSIM que les autres méthodes. En contrepartie, la méthode *MFP* obtient de meilleurs résultats avec la métrique PSNR. Cet écart peut être dû à l'apparition d'artefacts dans les images de la bases VIREDA avec la méthode *TCVD*, qui se manifeste par zones noircies au fond de l'image (voir la figure 4.27 à la sous-section 4.4.1). De plus, si les cartes de transmission ne sont pas bien estimées, le brouillard est peu atténué et les images résultantes conservent une bonne qualité visuelle. Cette observation pourrait expliquer les résultats satisfaisants obtenus avec la métrique PSNR. La méthode *MFP-vid* ne semble pas significativement apporter d'amélioration à la méthode de base. Il serait intéressant de tester d'autres filtrages plutôt que le filtrage médian dans le futur.

Le tableau 4.4 présente les résultats obtenus avec différentes méthodes pour chaque éclairage et densité de brouillard. Il montre l'écart important de résultats entre l'éclairage 1 $ec1$, qui correspond à l'éclairage coloré, et les autres éclairages.

TABLE 4.4 – Résultats obtenus avec les métriques SSIM/PSNR pour les différents éclairages et une densité de 0.015 (images en taille réelle).

	SSIM/PSNR					
méthode	ec1	ec2	ec3	ec4	ec5	ec6
DCP	0.52/16.23	0.56/8.87	0.54/9.33	0.56/8.89	0.58/8.36	0.59/8.97
MFP	0.57/16.78	0.88/15.63	0.76/15.79	0.77/15.46	0.78/15.40	0.77/15.10
TCVD	0.72/12.64	0.77/12.10	0.77/12.20	0.75/11.20	0.76/11.54	0.79/11.85
FFA-NET	0.29/13.31	0.74/14.57	0.75/14.56	0.74/14.69	0.76/14.81	0.75/14.81

4.4.3 Consistance temporelle

Pour rappel, réaliser un traitement vidéo trame par trame peut entraîner l'apparition d'artefacts visuels, de variations d'intensités et de couleurs au sein d'une vidéo. Avoir recours à une méthode dédiée à la vidéo en exploitant la temporalité de la séquence vise à atténuer ces effets visuels qui peuvent être gênants. Dans cette sous-section, la consistance temporelle des vidéos restaurées avec notre méthode est évaluée à l'aide d'une métrique introduite dans l'équation 4.7. Il s'agit d'un calcul d'écart des moyennes des images consécutives. Ainsi, la valeur de ε sera d'autant plus élevée que la différence entre deux images consécutives est importante, par exemple à cause d'un changement d'intensité ou de couleur (voir la figure 4.30).

$$\varepsilon = \Sigma_{i=1}^{n-1} \frac{|\ \langle I_i \rangle - \langle I_{i-1} \rangle\ |}{n-1} \tag{4.7}$$

où $\langle I_i \rangle$ correspond à la moyenne de l'image I à la position i, et n au nombre de trames consécutives.

TABLE 4.5 – Calcul d'écart des moyennes entre des images consécutives (équation 4.7) avec les méthodes $TCVD$ et MFP sur la base de test VIREDA. Elle est décomposée en trois sous-bases avec trois densité de brouillard différentes, exprimées par des valeurs de contraste, du plus dense au moins dense : 0.015, 0.05, 0.15. La meilleure valeur est en gras.

0.015	TCVD	MFP	0.05	TCVD	MFP	0.15	TCVD	MFP
ε	0.0060	**0.0039**	ε	**0.0029**	0.0040	ε	**0.0022**	0.0034

Le tableau 4.5 présente les résultats du calcul de l'équation 4.7 sur la base de données VIREDA pour différentes densités de brouillard, déterminées par des valeurs de contraste. De plus, le calcul est réalisé sur les vidéos avec un éclairage de jour. La méthode $TCVD$ a de meilleurs résultats sur les bases de données avec les densités de brouillard correspondants aux valeurs de contrastes 0.05 et 0.15. Cependant, concernant la sous-base où le brouillard à un contraste de 0.015 (le plus dense), les performances sont moins élevées avec la méthode $TCVD$. Ces résultats peuvent s'expliquer par l'apparition de traces noires après application de la méthode de restauration sur les trames de la vidéo (voir la deuxième ligne de la figure 4.30). Ces traces apparaissent plus visibles et plus nombreuses après avoir appliqué la méthode de restauration sur les vidéos avec un brouillard dense.

FIGURE 4.30 – Restauration d'une vidéo de la base de données VIREDA avec les méthodes MFP (première ligne) et TCVD (deuxième ligne).

La figure 4.30 présente des trames d'une vidéo restaurée avec les deux méthodes MFP et $TCVD$. Les images de la première ligne, obtenues avec la méthode MFP (restauration d'images trame par trame), présentent des décalages de couleur dans la séquence. L'image centrale est en effet plus rosée. Le changement de couleur dans une séquence vidéo est un type d'inconsistance temporelle.

4.4.4 Brouillard de synthèse vs brouillard réel

En raison de la difficulté à créer des bases de données d'images et de vidéos de brouillard en conditions réelles avec des vérités terrain, les bases de données composées d'images et de séquences d'images construit avec un brouillard de synthèse se sont multipliées. L'utilisation de données de synthèse est valable pour toutes les conditions météorologiques comme la pluie. Pour créer des bases de données de synthèse, il faut élaborer des jeux de données qui serviront de vérités terrain ainsi que des cartes de profondeur associées, afin d'ajouter synthétiquement du brouillard ou de la pluie. Bien que la création de ce type de base de données semble facilement accessible, les types de conditions météorologiques reproduit synthétiquement ne modélisent pas efficacement la réalité. Il est en effet difficile, par exemple, de reproduire l'aspect hétérogène du brouillard. Ainsi, les algorithmes basés sur des techniques de *deep learning* appris sur des bases de données de synthèse, ne permettent pas de restaurer de manière satisfaisante les images acquises en conditions réelles.
La figure 4.31 présente un exemple d'images restaurées appartenant à trois bases de données de tests différentes et illustre l'observation précédente : la base de synthèse NYU-Depth V2, REVIDE et VIREDA. Le réseau de neurones de la méthode $TCVD$ a été appris sur la base d'apprentissage de NYU-Depth V2, sur lequel j'ai ajouté synthétiquement du brouillard grâce aux cartes de profondeurs. Elle est constituée de 199 vidéos dans la base d'apprentissage, et 103 dans la base de validation. Chaque vidéo contient un brouillard dont la densité a été déterminé aléatoirement pour chaque vidéo. La première ligne de la figure est une trame appartenant à une vidéo de la base de tests créée à partir de NYU-Depth V2. L'algorithme parvient à atténuer le brouillard de synthèse de manière satisfaisante, contrairement aux images appartenant aux bases de données REVIDE et VIREDA. Cet exemple montre que cette approche ne permet pas de généraliser aux images avec du brouillard réel.

Une autre approche consiste à mélanger les séquences vidéos avec les deux types de brouillard

FIGURE 4.31 – Résultats de la suppression du brouillard avec le réseau de la méthode TCVD appris sur la base de données NYU-Depth V2.

dans une base d'apprentissage.
La figure 4.32 présente les résultats obtenus après restauration des trames vidéos des bases de données REVIDE et VIREDA à partir du réseau appris avec différentes bases d'apprentissages. La colonne *Synthèse* (apprentissage avec la base de données NYU-Depth) correspond aux résultats de la figure 4.31. Les images de la colonne *Revide* + *Synthèse* sont les résultats obtenus après l'apprentissage de notre réseau sur une base d'apprentissage constituée de vidéos avec du brouillard de synthèse (52 vidéos de la base NYU-Depth) et du brouillard réel (42 vidéos de la base REVIDE). Les images de la colonne *Revide* sont obtenues avec la méthode dont la base d'apprentissage est constituée uniquement de vidéos de la base REVIDE, et utilisée dans l'évaluation 4.4.1. Les artefacts noirs présents dans la première image de la colonne *REVIDE* peut être le résultat d'une sur-restauration du contraste provoqué par l'assombrissement les zones blanches laissées par l'éclairage. Alors que dans la colonne *Synthèse*, le brouillard n'est pas atténué dans le fond du dispositif, nous avons décidé de tester le mélange des deux bases d'apprentissage. Nous constatons dans la première image de la colonne *Revide* + *Synthèse* l'absence des artefacts noirs probablement dus au léger brouillard résiduel au fond du dispositif.

Le tableau 4.6 présente les résultats obtenus des trois méthodes avec les métriques SSIM et PSNR pour une valeur de densité de brouillard (la plus élevée) égale à 0.015.
Les résultats montrent que les méthodes *Synthèse* et *Revide* + *Synthèse* surpassent la méthode *REVIDE* en termes de SSIM et PSNR, malgré le fait que cette méthode parvient mieux à atténuer le brouillard. Ces résultats montrent que les artefacts noirs présents dans les images

FIGURE 4.32 – Résultats obtenus sur les images des bases de données REVIDE et VIREDA à partir du réseau appris avec différentes bases d'apprentissages.

TABLE 4.6 – Résultats obtenus avec les métriques SSIM et PSNR pour les méthodes dont le réseau a été appris sur trois bases de données différentes.

Métrique	*Synthèse*	*REVIDE*	*Revide + Synthèse*
SSIM	**0.89**	0.76	0.84
PSNR	15.90	11.82	**17.01**

avec la méthode *REVIDE* ont bien un fort impact sur les performances des métriques SSIM et PSNR.

La figure 4.7 reprends le tableau 4.3 auquel ont été ajoutés les performances des méthodes *TCVD* et *FFA-Net* entraînées sur la base d'apprentissage *Revide + Synthèse*, appelées respectivement *TCVD*2 et *FFA-Net*2. La méthode *TCVD*2 surpasse les autres en termes de SSIM, PSNR, et FSIM. La différence de performance avec la métrique PSNR entre *TCVD* et *TCVD*2 est flagrante et témoigne de l'impact des artefacts sur les performances dans les images. Ces résultats sont à considérer en prenant bien en compte le fait que les réseaux des méthodes *FFA-Net* et *TCVD* ne sont pas appris sur la même base d'apprentissage que ceux de *TCVD*2 et *FFA-Net*2. Par contre, l'apprentissage de l'algorithme *FFA-Net*2 (avec une diminution de la taille du batch à 2 due aux limitations de la capacité mémoire du gpu) sur la base *Revide + Synthèse* n'améliore pas les performances de l'algorithme en termes de SSIM et PSNR. Néanmoins, les performances sont améliorées pour la métrique FSIM. Ces résultats montrent que la composition de la base d'apprentissage et l'architecture du réseau influent conjointement et de manière significative sur les performances des algorithmes.

Enfin, les deux types de brouillard présents nouvelle base sont très différents. Pour rappel, le brouillard de la base de données NYU-Depth est un brouillard de synthèse spatialement uniforme tandis que le brouillard des bases de données REVIDE et VIREDA a été obtenus avec une machine à brouillard. Cette différence au sein d'une même base de donnée semble provoquer des difficultés dans le processus d'apprentissage. J'ai en effet pu constater des difficultés de convergence ainsi qu'un sur-ajustement précoce. Néanmoins, les résultats obtenus avec cette méthode sont satisfaisants et mériteraient d'être plus amplement étudiés.

TABLE 4.7 – Évaluation quantitative sur la base de tests VIREDA.

métrique	densité	DCP	MFP	MFP-vid	FFA-Net	TCVD	FFA-Net2	TCVD2
SSIM	0.015	0.75	0.59	0.63	0.67	0.76	0,63	**0.84**
	0.05	0.60	0.78	0.71	0.67	0.78	0.69	**0.92**
	0.15	0.64	0.82	0.77	0.73	0.84	0.73	**0.92**
PSNR	0.015	10.11	15.69	13.38	14.26	11.81	11.38	**17.01**
	0.05	10,43	15.78	15.25	14.47	12.80	14.03	**20.30**
	0.15	11.50	16.40	16.72	16.26	13.82	15.34	**20.38**
FSIM	0.015	0.55	0.84	0,85	0.52	0.90	0.53	**0.91**
	0.05	0.62	0.90	0.89	0.53	0.90	0.59	**0.96**
	0.15	0.66	0.93	0.93	0.58	0.92	0.62	**0.96**

4.4.5 Difficultés rencontrées

4.4.5.1 Le sur-ajustement

Le sur-ajustement (ou *overfitting* en anglais) est une complication fréquente en *deep learning* survenant lorsque le modèle commence à mémoriser le bruit des données d'entraînement au lieu de poursuivre le processus d'apprentissage. Alors que l'objectif est d'obtenir des résultats au plus proche de la réalité, le phénomène de sur-ajustement peut induire une forte augmentation des marges d'erreurs. La mémorisation des données d'apprentissage entraîne des difficultés de généralisation, c'est-à-dire que la prédiction du modèle sur des nouvelles données seront de mauvaise qualité. Il existe différentes méthodes pour résoudre les problèmes de sur-ajustement :

1. **Ajouter des données dans la base d'apprentissage** : lorsque la quantité de données est insuffisante, le modèle génère des erreurs. Pour pallier ce problème, une technique d'augmentation des données, comme le recadrage, les rotations et les retournement d'images, permet d'étendre la base d'apprentissage.
2. **Régulariser le modèle** : il existe des techniques de régularisation permettant de simplifier le modèle en pénalisant les matrices de poids des nœuds du réseau. Un exemple bien connu de régularisation en *deep learning* est la technique de *dropout*. À chaque itération, cet algorithme sélectionne de manière aléatoire de certains nœuds et les supprime afin de rendre le modèle plus simple.
3. **Utiliser la technique du *Early Stopping* (Prechelt (1997))** : cette technique permet d'éviter le sur-ajustement des données en interrompant l'entraînement selon certains critères. Par exemple si la fonction de perte ne diminue plus au terme d'un nombre d'époques donné, le modèle risque le sur-ajustement. Il devient donc nécessaire d'interrompre le processus d'apprentissage afin d'assurer une généralisation correcte.

4.4.5.2 La combinaison de Batch/Layer-Normalisation et de *dropout*

La combinaison de ces deux techniques de régularisation dans un même modèle fait encore l'objet de nombreux débats. En effet, le périmètre d'utilisation de ces deux méthodes n'est pas clairement défini. Alors que certains n'hésitent pas à utiliser les deux techniques, d'autres

insistent sur le fait que l'utilisation de la batch-normalisation élimine le besoin d'utilisation du *dropout* car elle offre des avantages de régularisation similaires. C'est le cas de Li et al. (2019b). La plupart des architectures comme ResNet ou DenseNet n'utilisent pas de *dropout* dans leur modèle.

Toujours est-il que l'assemblage de ces deux techniques de régularisation a été infructueux dans le processus d'apprentissage de mon réseau. En effet, la marge d'erreurs augmentait très tôt dans le processus dans le processus d'apprentissage et provoquait le sur-ajustement précoce du modèle.

4.5 Ablation Study

Dans cette section, des études d'ablation sont réalisées dans le but d'analyser l'importance des modules temporels de la méthode proposée dans ce chapitre. Pour chaque module temporel retiré ou ajouté, le réseau est entraîné sur la base de données REVIDE avec les même hyperparamètres que ceux utilisés dans la section précédente.

4.5.1 Importance des blocs TpFormer dans la méthode de suppression de brouillard dans des vidéos

Pour illustrer cette analyse, les différents blocs **TpFormer** ont été assignés à une couleur distincte. La figure 4.33 présente le schéma de l'encodeur avec les différents blocs de couleurs. Le bloc bleu n'a pas pu être utilisé comme quatrième bloc temporel faute de performances de calcul. J'ai donc décidé, dans le cadre de cette section, de tester ce bloc seul à la fin de l'encodeur (voir la colonne B du tableau 4.8).

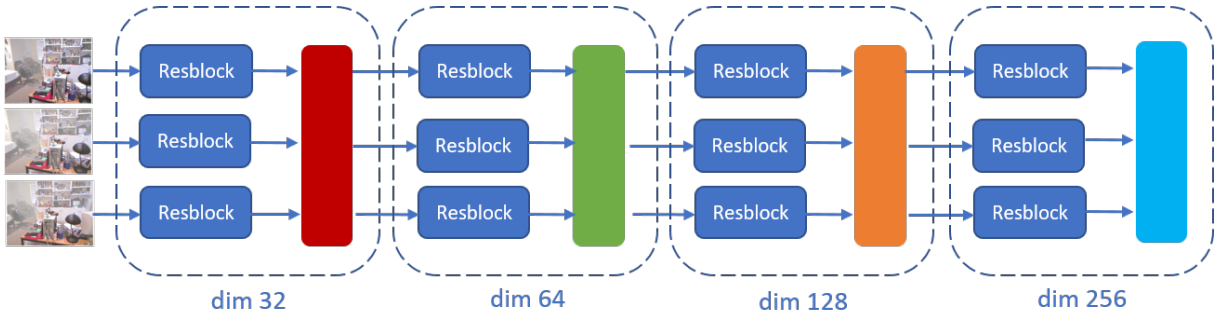


FIGURE 4.33 – Schéma de l'encodeur avec des blocs **TpFormer** de couleurs pour faciliter la compréhension de l'analyse.

Le tableau 4.8 présente les résultats des métriques SSIM et PSNR de la méthode avec les différents blocs **TpFormer**. Les lettres présentent dans la première ligne correspondent à la première lettre de la couleur des blocs (R pour rouge, V pour vert, O pour orange et B pour bleu). Les évaluations sont réalisées sur des images de taille 224×224.

Le tableau 4.8 montre que les meilleurs résultats sont obtenus avec l'utilisation de la configuration B, soit avec le bloc bleu seul. La ligne du tableau « Paramètres », correspond au nombre de paramètres du réseau correspondant. Le réseau composé uniquement du bloc B à de meilleures performances, en particulier avec la métrique PSNR. En contrepartie, le nombre de paramètres a presque triplé, et la différence de performances en termes de SSIM et PSNR reste assez faible. En effet, il est préférable de mettre en œuvre un réseau efficace avec le moins

TABLE 4.8 – Analyse des résultats de la méthode avec les différents blocs **TpFormer**. Les éléments de la première ligne du tableau correspondent à la première lettre de la couleur des blocs.

Métriques/Blocs	R+V+O (base)	V+O	O	B
SSIM	0.8968	**0.9023**	0.9016	**0.9035**
PSNR	21.20	**21.45**	21.07	**22.77**
Paramètres	4 513 763	4 387 907	4 037 891	11 259 651

de paramètres possibles. J'en déduis alors que la configuration avec uniquement le bloc B n'est pas la plus adaptée pour mon cas d'étude. De manière globale, les performances sont assez similaires quelles que soient les configurations. Finalement, la configuration avec les blocs $V+O$ semble être un bon compromis.

4.5.2 Architecture *Transformer* vs ConvLSTM

L'architecture *Transformer* est-elle vraiment adaptée à notre application de suppression de brouillard dans les vidéos, malgré sa popularité ? Les architectures *Transformer* et ConvLSTM ont chacune des avantages et des inconvénients rappelés dans l'introduction de ce chapitre. Dans cette sous-section, une analyse de la confrontation de ces deux méthodes est réalisée (voir le tableau 4.9). Les trois blocs **TpFormer** d'origine (R+V+O sur la figure 4.8) sont remplacés par des blocs **Att3D-ConvLSTM**, **Att3D** et **ConvLSTM**.

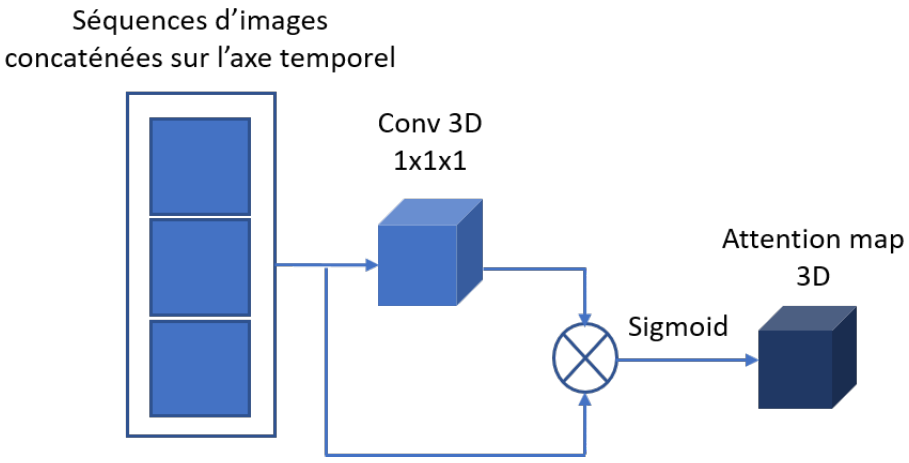


FIGURE 4.34 – Schéma du bloc Att3D.

Le bloc Att3D-ConvLSTM représenté à la figure 4.35 est composé de deux couches ConvLSTM2D (provenant du framework Keras), et d'un bloc d'attention temporel **Att3D**, détaillé à la figure 4.34. Ce bloc d'attention consiste à concaténer un triplet de cartes de caractéristiques (les *features* des images, $F(I_{i-1})$, $F(I_i)$, $F(I_{i+1})$) de la séquence comme entrée d'une convolution 3D afin de capturer l'information le long de l'axe temporel. Les trois cartes de caractéristiques ont une dimension 2D et ont la même taille, $\Re^{C\times H\times W}$. L'opération Sigmoid permet d'obtenir la carte d'attention correspondante.

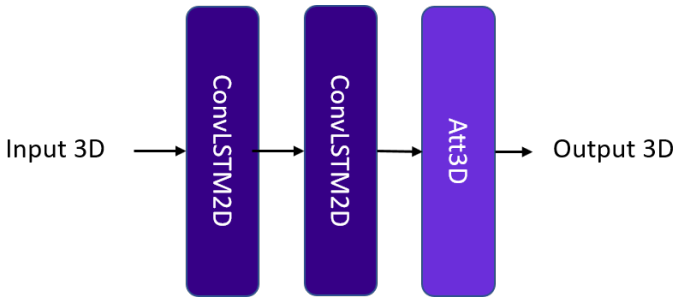


FIGURE 4.35 – Schéma du bloc Att3D-ConvLSTM.

TABLE 4.9 – Analyse des résultats de la méthode avec les différents blocs **TpFormer**, **Att3D-ConvLSTM**, **Att3D** et **ConvLSTM**

Métriques/Blocs	TpFormer	Att3D-ConvLSTM	Att3D	ConvLSTM
SSIM	**0.8968**	0.8840	0.8827	0.8790
PSNR	21.20	20.92	**21.49**	21.07
Paramètres	4 513 763	3 367 334	3 021 478	3 290 499

Le tableau 4.9 montre que l'utilisation des blocs TpFormer améliore légèrement les performances en terme de SSIM. Le nombre de paramètre est également plus élevé avec les blocs Tpformer qu'avec les blocs Att3D et ConvLSTM. Les résultats montrent que l'utilisation de Transformers dans cet algorithme n'améliore pas significativement les résultats.

4.6 Conclusion

Dans ce chapitre, les différentes étapes de la conception du dispositif permettant l'acquisition des données pour la base de données de tests VIREDA, ont été détaillées. Proposée en complément de la base de données REVIDE pour les besoins de la thèse, elle a été rendue accessible publiquement. Elle contient des vidéos d'une scène avec plusieurs densités et brouillard et plusieurs conditions d'éclairage différentes. Elle a permis d'évaluer l'algorithme $TCVD$, également présenté dans ce chapitre.

La figure 4.36 récapitule les méthodes existantes d'atténuation du brouillard dans des images et des vidéos. Les rectangles violets foncés correspondent aux contributions de cette thèse : les méthodes d'atténuation de brouillard MFP et $TCVD$, ainsi que la base de données VIREDA.

La méthode $TCVD$ (*Transformer-CNN for Video Defogging*), a été conçu avec la volonté d'exploiter l'information temporelle des trames adjacentes d'une vidéo. Pour cela, nous avons créé une architecture constituée de blocs temporels basés sur l'utilisation du *Transformer*, et de blocs spatiaux constitués de CNN. Dans la partie évaluation, $TCVD$ a été comparée à d'autres méthodes d'atténuation du brouillard destinées aux images, en raison du manque de ressources disponibles des méthodes existantes dédiées aux vidéos contenant du brouillard. L'évaluation menée a montré des résultats satisfaisants, mais encore perfectibles. D'un point de vue qualitatif, la méthode parvient à restaurer les trames vidéo, mais laisse des artefacts noirs au fond du dispositif dus probablement à une sur-restauration. La modification du contenu de la base d'apprentissage en ajoutant des vidéos avec du brouillard de synthèse a montré des résultats

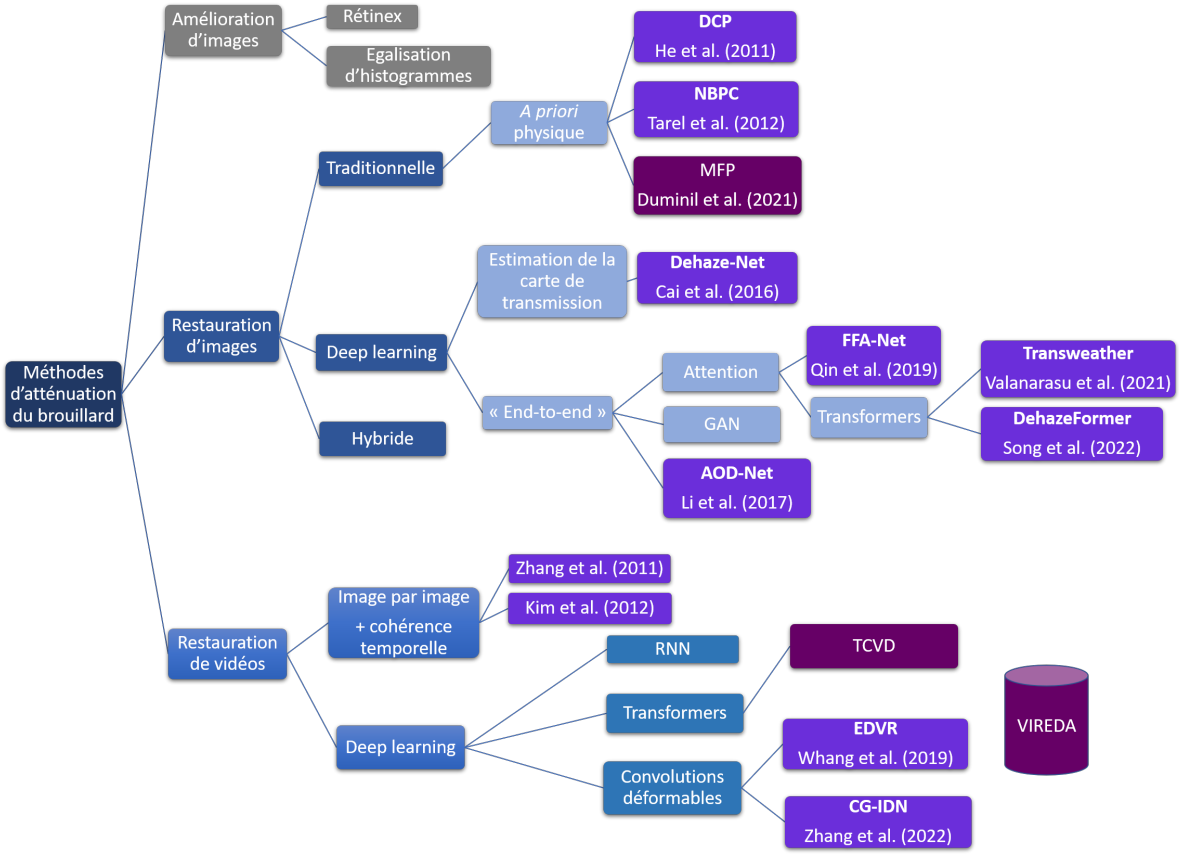


FIGURE 4.36 – Diagramme récapitulatif des méthodes de l'état de l'art.

prometteurs qui méritent d'être approfondis dans le futur.

Ce chapitre a fait l'objet d'un article de conférence, qui sera publié prochainement (ASPAI 2022).

Chapitre 5

Applications des algorithmes d'atténuation du brouillard pour le pré-traitement des images

5.1 Introduction

Les algorithmes de restauration d'images peuvent être utilisés comme pré-traitement pour diverses applications comme la photographie computationnelle et la reconnaissance visuelle. La photographie computationnelle, combinaison entre la photographie traditionnelle (prise de vue brute) et le post-traitement photographique vise à améliorer la qualité visuelle de l'image. La plupart des travaux en restauration d'images contenant du brouillard ont pour objectif d'acquérir des images restaurées de la meilleure qualité visuelle possible tout en minimisant les artefacts gênants dans la reconnaissance automatique d'objets. Ce n'est pas toujours le cas et il existe également des méthodes de restauration utilisées seulement pour des tâches de reconnaissance d'objets ne privilégiant pas la qualité visuelle afin d'obtenir la plus grande visibilité des objets possibles dans les images (Hautière et al. (2007), Liu et al. (2022a)). Malgré cela, VidalMata et al. (2021) ont récemment tenté d'allier ces deux applications, en proposant une méthode pour améliorer à la fois la qualité visuelle et l'interprétabilité des images pour la reconnaissance visuelle. Dans ce chapitre, des exemples des applications citées vont être présentés. Nous nous placerons dans le cas des ADAS pour la reconnaissance visuelle et la détections d'objet.

5.2 La détection d'objets en conditions météorologiques dégradées

5.2.1 L'algorithme YOLO

De plus en plus de véhicules sont équipés de systèmes avancés d'aides à la conduite (ADAS). Grâce à des capteurs embarqués, les applications sont multiples : détection et reconnaissance de signalisation routière et d'obstacles, ou encore prédiction du comportement des usagers de la route (véhicules et piétons). Cependant, en présence de mauvaises conditions météorologiques, la visibilité est réduite et les contrastes atténués ont pour conséquences de rater des détections.

Dans la figure 5.1, ces erreurs, induisent une mauvaise interprétation de l'environnement, qui peut augmenter les risques d'accidents de la circulation. Afin de diminuer ces risques, il est utile d'introduire des algorithmes de pré-traitement d'images et de vidéos. Le but de ces algorithmes est de restituer une image en atténuant le brouillard pour augmenter le contraste

FIGURE 5.1 – Détection d'objets par temps de brouillard

des images. Les algorithmes que j'ai mis en œuvre au cours de cette thèse peuvent être utilisés comme pré-traitement aux algorithmes de détection et de reconnaissance de véhicules, de signalisations ou de piétons. Les algorithmes de détection d'objet en vision par ordinateur ont pour objectif de détecter, reconnaître et classifier divers objets dans des images et des vidéos. La question est de savoir quels sont les objets à détecter et les localiser. Il existe plusieurs approches de détection d'objets comme Faster R-CNN (Ren et al. (2015)), Retina-Net (Lin et al. (2017)) ou encore YOLO (Redmon et al. (2016)). Dans la suite de ce chapitre nous utilisons une version de l'algorithme YOLO pour ses excellentes performances et sa capacité de prédiction d'objets en temps-réel. You Only Look Once (YOLO) est un algorithme de détection d'objets en temps réel qui ne cesse de gagner en popularité. L'algorithme utilise des réseaux de neurones convolutionnels pour prédire simultanément, en temps réel, les probabilités d'appartenance à une classe et les coordonnées des boîtes englobantes. Un des avantages de YOLO est la prédiction de l'ensemble des objets dans l'image en une seule exécution.

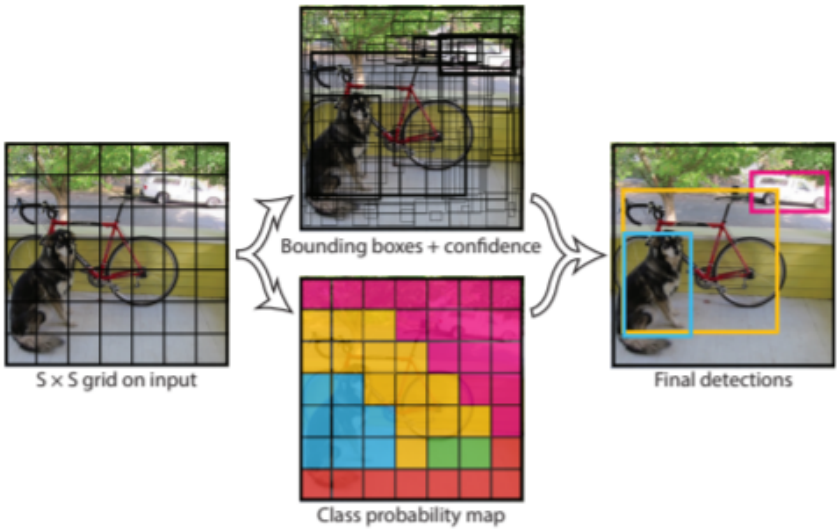


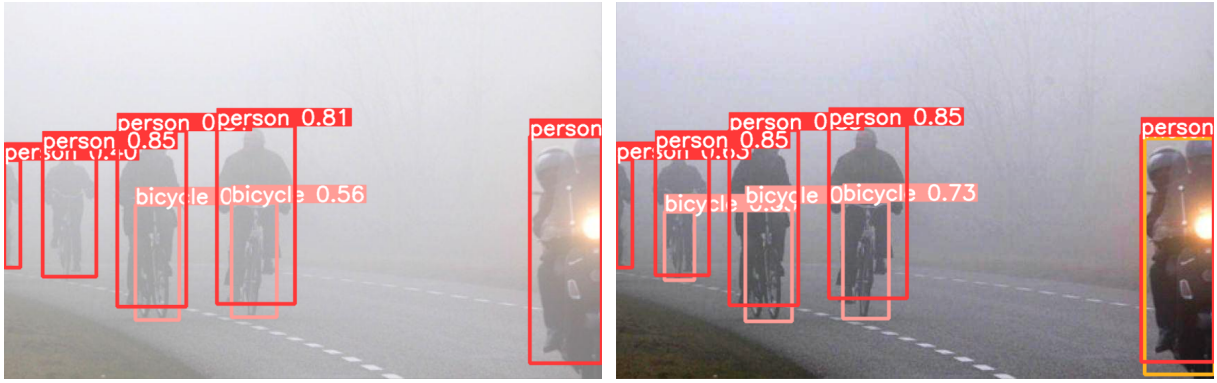
FIGURE 5.2 – Principe de fonctionnement du modèle YOLO, extrait de l'article de Redmon et al. (2016).

La figure 5.2 illustre le principe de fonctionnement de l'algorithme. L'image est, dans un premier temps, segmentée en une grille de dimensions S × S. Pour chaque imagette de la grille, les boîtes englobantes B, la confiance associée à ces boîtes, et les probabilités de présence de chaque classe C_i dans la boîte sont prédites. Dans l'image centrale du haut, plusieurs boîtes englobantes se superposent autour des objets. Pour obtenir le résultat de l'image de droite, avec une boîte englobante par objet, YOLO utilise la métrique d'évaluation *Intersection Over Union* (IOU) pour obtenir une boîte englobante qui entoure au mieux les objets. Pour cela, des boîtes englobantes « vérités terrain » sont nécessaires. Dans l'image finale, chaque objet est entouré par une boîte englobante de différentes couleurs indiquant leur appartenance aux trois classes les plus probables.

L'algorithme YOLO peut être utilisé dans plusieurs domaines d'application pour renforcer les systèmes de sécurité, la détection de faune, ou encore la conduite autonome, pour ne citer qu'eux en exemple.

5.2.2 La détection d'objets en conditions météorologiques dégradées

Des recherches ont été conduites sur la détection d'objet en conditions météorologiques dégradées. Une première approche consiste à appliquer un algorithme de suppression de brouillard avant la détection. La figure 5.3 montre le résultat d'une détection dans le cas d'une application routière. Pour cela, nous mettons en œuvre l'algorithme de restauration du chapitre (3). Dans le brouillard (figure 5.3a), YOLO a permis de détecter 5 personnes et 2 vélos avec une certaine précision. Après restauration, l'algorithme a pu détecter un vélo et une moto supplémentaire avec une meilleure précision. Parfois, l'utilisation du pré-traitement n'améliore pas la confiance dans la détection comme l'indique la valeur 0.85 pour la troisième personne dans l'image. Dans d'autres cas, comme le vélo présent au milieu de l'image, la précision s'améliore (0.56 avant restauration et 0.73 après restauration).



(a) (b)

FIGURE 5.3 – Détection avant et après atténuation du brouillard : (a) application de l'algorithme de détection sans restauration, (b) application de l'algorithme de détection après restauration.

Cependant, d'après Liu et al. (2022a), l'amélioration de la qualité de l'image avec des méthodes de restauration peut ne pas profiter aux performances de détection, si la méthode utilisée ne restaure pas correctement l'image. Ils proposent une méthode incluant un module de traitement d'image (DIP) pour prendre en compte les conditions météorologiques en entrée d'un détecteur YOLO. Huang et al. (2020) proposent de résoudre le problème de la détection

d'objets en présence de brouillard en introduisant un réseau double (DSNet) pour l'apprentissage conjointe de trois tâches : la restauration de la visibilité, la classification et la localisation des objets. Cette méthode produit de meilleurs résultats dans le brouillard que les méthodes classiques de détection. Des méthodes d'adaptation de domaine ont également été proposées pour permettre aux détecteurs de bien généraliser aux applications du monde réel (Zhang et al. (2021), Hnewa and Radha (2021)).

5.2.3 Évaluation

Pour évaluer l'algorithme MFP (chapitre 3) pour une application de détection dans un contexte d'ADAS, j'ai utilisé la version YOLOv5s entraîné sur la base de données COCO128, contenant les 128 premières images de la base de données COCO (Lin et al. (2014)). L'évaluation a été réalisée sur *Real-world Task-Driven Testing Set* (RTTS) de la base de données RESIDE, de Li et al. (2019a). Il existe différentes métriques pour évaluer les algorithmes de détection. La métrique *Mean Average Precision* (maP) est couramment utilisée pour évaluer la robustesse des algorithmes de détection et de segmentation d'objets comme Faster-R-CNN et Yolo.

La formule de la métrique maP est basée sur les sous-métriques suivantes : Matrice de confusion, IoU, *Recall* (R), *Precision* (P). La métrique *Recall* mesure le pourcentage de vrais positifs sur l'ensemble des prédictions positives, et *Precision* mesure le pourcentage de vrais positifs dans l'ensemble des prédictions (positives et négatives). La métrique *Average Precision* est donc définie comme l'aire sous la courbe precision/recall (courbe ROC). Dans la table 5.1 la métrique mAP@[.5,.95] correspond au calcul de maP moyen sur différents seuils de IoU entre 0,5 à 0,95 par pas de 0,05.

TABLE 5.1 – Évaluation quantitative sur la base RTTS composées de 4322 images de taille 640 $\times$640.

Méthodes	P	R	maP@[.5]	maP@[.5 :.95]
Sans restauration	0.568	0.436	0.495	0.288
Avec restauration (MFP)	0.554	0.448	0.504	0.294
Avec restauration (TCVD)	0.557	0.472	0.522	0.3

Le tableau 5.1 montre que les résultats obtenus avant et après restauration sont très proches, et ne semblent pas significatifs. Peut-être que la méthode MFP n'est pas assez performante pour cette application. L'utilisation de la méthode $TCVD$ permet d'obtenir des résultats un peu plus satisfaisants. Malgré cela, ajouter une étape de pré-traitement influe sur les performances temps réel à cause du temps de traitement de restauration. Dans le cas de nos deux méthodes, le temps de prédiction de l'image restaurée pour une taille d'image de 640×640 dépasse la seconde, ce qui n'est pas approprié pour le temps réel.

De plus, l'importance du traitement de restauration n'est pas d'obtenir le résultat de la meilleure qualité possible mais que la qualité de l'image soit suffisante pour que la scène soit interprétée sans ambiguïté en fonction de l'application concernée. En effet, si le traitement est destiné aux systèmes chargés d'interpréter les données, il n'est pas nécessaire d'obtenir une restauration d'une qualité visuelle parfaite. D'après la loi des 80/20 (aussi appelée loi Pareto), 20% des actions produisent 80% des résultats. La question est de savoir si les 80% restant

d'effort pour avoir une qualité d'image parfaite pour un gain en performance de 20%, est réellement nécessaire. De plus, l'utilisation de plusieurs pré-traitements pour améliorer la qualité visuelle, par exemple l'utilisation conjointe de méthodes de restauration de la visibilité et de super-résolution, peut impacter les performances temps réel.

D'un autre côté, si le traitement est destiné au conducteur, la qualité visuelle devrait être meilleure.

5.3 L'atténuation de brouillard en photographie

En ce qui concerne le brouillard, il existe deux approches en photographie computationnelle. Celles qui préfèrent tirer parti des différentes nuances de la brume et conserver l'aspect mystérieux des paysages, et ceux qui préfèrent l'atténuer au maximum. En effet, le brouillard atténue les contrastes et les couleurs et contribue à bruiter l'image. Les travaux réalisés sur l'atténuation de brouillard dans les images sont en général adaptés pour ce type d'application. En effet, l'évaluation qualitative des méthodes de l'état de l'art présentent des comparaisons entre les images restaurées avec pour objectif de présenter des résultats ayant la meilleure qualité visuelle possible.

En photographie, les logiciels de retouches, comme Adobe Lightroom, propose une fonctionnalité de correction de voile atmosphérique à l'aide d'un curseur. Déplacer le curseur vers la droite permet de diminuer la densité du voile atmosphérique, alors que le déplacement du curseur vers la gauche entraîne l'augmentation de sa densité. Le résultat n'apparaît pas toujours très naturel, il dépend surtout de la densité du brouillard. Plus il est dense, plus il est difficile d'obtenir une image sans brouillard aux couleurs naturelles (voir la figure 5.4c).

(a) (b) (c)

FIGURE 5.4 – Correction des images contenant du brouillard réel avec le logiciel Adobe Lightroom : (a) image en entrée, (b) ajout de brouillard supplémentaire, (c) atténuation du brouillard.

La figure 5.5 présente les images obtenues après avoir effectué un traitement d'amélioration du brouillard avec les méthodes MFP et TCVD. Bien que la méthode MFP peine à restaurer les éléments de l'images à cause du brouillard dense, la méthode TCVD produit un meilleur résultat en termes de restauration de la visibilité que MFP et Adobe Lightroom dans la figure 5.4c. Ces résultats illustrent que cette méthode peut être utilisée comme technique de post-traitement en photographie computationnelle.

(a) (b) (c)

FIGURE 5.5 – Atténuation du brouillard dans des images réelles avec nos méthodes : (a) Image en entrée, (b) Atténuation du brouillard avec MFP, (c) Atténuation du brouillard avec TCVD.

5.4 Atténuation des traînées de pluie

5.4.1 Méthodes existantes

Comme l'atténuation du brouillard dans les images et les vidéos en pré-traitement d'autres tâches, les algorithmes d'atténuation des traînées de pluie comme pré-traitement des algorithmes de détection et la reconnaissance d'objets sont également nécessaires pour améliorer leurs performances. Cette application a fait l'objet de nombreux travaux. Wang et al. (2019a) ont réalisé une étude sur les méthodes existantes de suppression de la pluie en s'appuyant sur une vidéo et une image unique. Les méthodes de suppression de la pluie dans les vidéos peuvent être divisées en trois catégories : les méthodes basées sur des propriétés physiques, celles basées sur des techniques de parcimonie et enfin celles avec des techniques de *deep learning*. Grâce à l'apport d'information temporelle, les méthodes d'atténuation de la pluie sur des vidéos sont plus faciles à appréhender, que les méthodes d'atténuation de la pluie dans des images uniques. Les méthodes basées sur les images peuvent, quant à elles, être divisées en trois catégories : les méthodes basées sur le filtrage, la modélisation de l'apparence des traînées de pluie et les méthodes basées sur le *deep learning*. Les méthodes d'atténuation de la pluie sont classées dans le schéma de la figure 5.6.

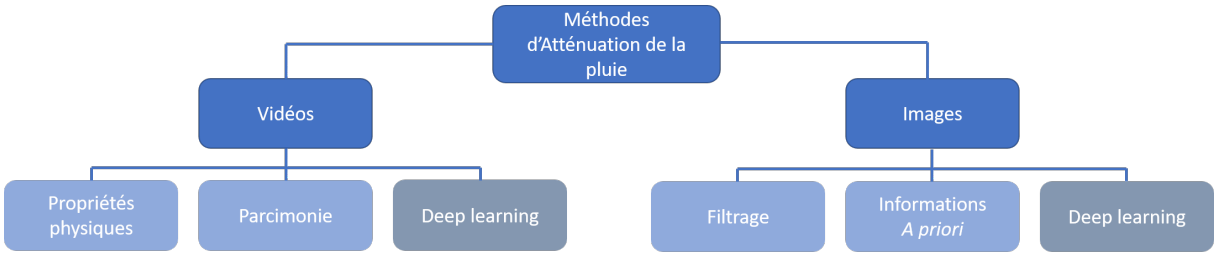


FIGURE 5.6 – Hiérarchisation des méthodes existantes de suppression de la pluie.

Parmi les méthodes existantes, quelques-unes ont été sélectionnées afin de réaliser une évaluation qualitative. Zhang and Patel (2018), ont proposé un algorithme basé sur un réseau neuronal convolutif multi-flux, sensible à la densité de la pluie, appelé DID-MDN. Il consiste à estimer la densité de la pluie et à supprimer en même temps les traînées de pluie. Ainsi, il permet au réseau de déterminer automatiquement les informations de densité, puis de supprimer efficacement les traînées de pluie correspondantes guidées par le label indiquant le niveau de densité de la pluie estimée. Ils ont également proposé une base de données composée de 12000

images construites avec de la pluie de synthèses et de vérités terrain, ainsi qu'une base de tests de 1200 images. Cette dernière est utilisée dans la partie évaluation, à la section 5.4.3.

5.4.2 Adaptation de la méthode proposée pour l'atténuation des traînées de pluie

Afin d'adapter notre méthode TCVD, dédiée à la vidéo, à une méthode « statique » d'atténuation de pluie dans les images, les blocs Tpformer, qui assurent la cohérence temporelle des séquences vidéo, ont été retirés. Ainsi, nous gardons uniquement l'architecture auto-encodeur inspiré de U-net, pour traiter l'information spatiale. La plus-value de cette architecture est sa capacité à s'adapter à de nombreuses tâches différentes. Le principe consiste à alimenter la base d'apprentissage avec un jeu de données spécifique.

Le réseau de neurones est identique à celui de la méthode présentée dans le chapitre précédent, à l'exception du nombre de filtres qui a été augmenté pour réaliser cette application. Les tailles des filtres utilisées sont (64, 128, 256, 512) au lieu de (32, 64, 128, 256) dans la méthode d'origine. Dans la suite de la section, cette méthode sera appelée *DA-UNet* pour *Deraining Algorithm based on U-Net architecture*.

5.4.3 Évaluation

Il est nécessaire d'utiliser une base d'apprentissage contenant des paires d'images composées d'une vérité terrain et d'une image associée avec de la pluie pour entraîner le réseau. Tout comme le brouillard, il n'est pas aisé de se procurer des bases avec des données réalistes avec et sans pluie. Il existe toutefois quelques bases de données de synthèse. La base de données de Zhang and Patel (2018), appelée *DID-MDN*, est composée de 4000 images par densité de pluie (légère, moyenne et forte), soit 12 000 images conçues pour la partie entraînement des réseaux de neurones. La base de tests en comporte 1200. J'ai réalisé l'apprentissage de mon réseau en sélectionnant 8400 images pour la base d'apprentissage et 3600 images pour la base de validation. Le réseau a obtenu les scores suivant sur la base de tests de Zhang and Patel (2018) : $SSIM = 0.92$ et $PSNR = 32.6$. Le tableau 5.2 montre les résultats obtenus sur la base de tests avec deux autres méthodes de suppression des traînées pluie. Les résultats montrent que $DA-UNet$ semble surpasser les deux méthodes en termes de PSNR et de SSIM sur une base de tests d'images de synthèse. Malgré ces chiffres, la méthode de Zhang and Patel (2018) généralise mieux aux images avec de la pluie réaliste (voir la figure 5.7).

TABLE 5.2 – Résultats quantitatifs sur la base de test de 1200 images de Zhang and Patel (2018) avec les métriques SSIM et PSNR. Les résultats présentés dans les deux premières colonnes sont extraits de l'article de Zhang and Patel (2018).

Métriques	Yang et al. (2017)	Zhang and Patel (2018)	DA-UNet
PSNR	26.75	27.95	**32.60**
SSIM	0.8901	0.9087	**0.9200**

La figure 5.7 montre le résultat de la méthode d'atténuation de la pluie appliquée à une image de la base de tests. L'algorithme a également été testé sur des images réalistes. La figures 5.8

(a) (b) (c)

FIGURE 5.7 – Images après application de l'algorithme d'atténuation de traînées de pluie sur une image de la base de tests : (a) image avec des traînées de pluie, (b) image vérité terrain, (c) image restaurée.

présente le résultat de l'atténuation des traînées de pluie avec ma méthode et avec celle de Zhang and Patel (2018). *DA-UNet* atténue les effets des traînées de pluie dans les images. L'autre méthode semble cependant mieux généraliser aux images de pluie réalistes, car elle obtient un résultat de meilleure qualité visuelle. Cependant, il reste des traînées de pluie légères.

Les résultats obtenus sur des images avec de la pluie réelle à partir d'une méthode entraînée sur une base de données de synthèse ne sont pas convaincants. Les données de synthèse sont généralement créées en ajoutant une superposition 2D de traînées de pluie, ce qui est loin de correspondre à la réalité. Les images avec de la pluie réaliste présentent en général un effet de brume dont la densité est proportionnelle à la distance à la caméra. La figure 5.9 illustre ce cas. L'algorithme de suppression de pluie utilisé dans cette section n'est pas en mesure de supprimer la brume. Pour remédier à ce problème, j'ai utilisé *MFP* présenté au chapitre 3. Les images 5.9(b) et 5.9(c) présentent les résultats après suppression de la pluie puis suppression de la brume respectivement. L'image restaurée 5.9(c) semble légèrement atténuer la brume au niveau des arbres.
La majorité des bases de données existantes de suppression de pluie ont été réalisées sans prendre en compte les propriétés physiques de la pluie en conditions réelles. L'apprentissage des réseaux pour une application de suppression de pluie sur des jeux de données de synthèse ne permet pas aux méthodes de bien généraliser aux images réalistes. Cela limite l'efficacité des techniques de suppression de la pluie sur de vraies images. Un exemple est présenté à la figure 5.7. Hu et al. (2021) ont pris en compte les effets de la pluie et ont créé la base de données RainCityscapes à partir de la base de données Cityscapes (Cordts et al. (2016)). Ils ont utilisé les paramètres de la caméra et les informations sur la profondeur des différentes scènes de cette base pour synthétiser la pluie et le brouillard (voir l'exemple à la figure 5.10).
RainCityscapes se rapproche un peu plus des conditions réalistes des scènes par temps de pluie. Cependant, Hu et al. (2021) ont souligné que l'ajout du brouillard et de la pluie est indépendant et uniformément réparti dans leurs images alors qu'en réalité, il existe une corrélation entre les effets visuels de la pluie et ceux du brouillard.
La figure 5.11 présente une image avec de la pluie réaliste restaurée avec *DA-UNet*. Le réseau a été appris avec une partie seulement de la base de données Raincityscapes, en raison des limites du système qui ne peut allouer plus de mémoire pour prendre en compte la totalité de la base de

FIGURE 5.8 – Résultats de l'atténuation des traînées de pluie sur des images du monde réel : (a) image en entrée, (b) image restaurée DA-UNet, (c) image restaurée avec la méthode de Zhang and Patel (2018).

données. L'image 5.11b obtenue est satisfaisante et supprime davantage de brume par rapport à l'utilisation de la méthode à la figure 5.9.

5.5 Conclusion

Ce chapitre a présenté les différentes possibilités d'utilisation des algorithmes mis en œuvre dans cette thèse. Les algorithmes d'atténuation du brouillard peuvent notamment être utilisés comme méthode de pré-traitement aux algorithmes de détection d'objet et de reconnaissance visuelle. En effet, améliorer la visibilité des objets dans les images peut favoriser l'interprétabilité de l'environnement et améliorer les performances de détection d'objets.

Nos méthodes d'atténuation du brouillard peuvent également être utilisées comme méthodes de post-traitement pour atténuer le brouillard en gardant une bonne qualité photographique.

L'architecture inspirée de U-net de la méthode $TCVD$ permet de l'adapter à d'autres types de conditions météorologiques avec une base de données appropriée. Dans ce chapitre, ce sont les traînées de pluie qui ont été étudiés, et les résultats se sont avérés satisfaisants. En effet, l'utilisation de la base de données Rain-Cityscapes, dont les images sont composées à la fois de brouillard et de pluie, a permis d'obtenir des images restaurées intéressantes. Cependant, il s'agit d'images avec du brouillard et des traînées de pluie de synthèse. Le prochain challenge serait de créer une base de données composées de scènes avec de la pluie et du brouillard réels.

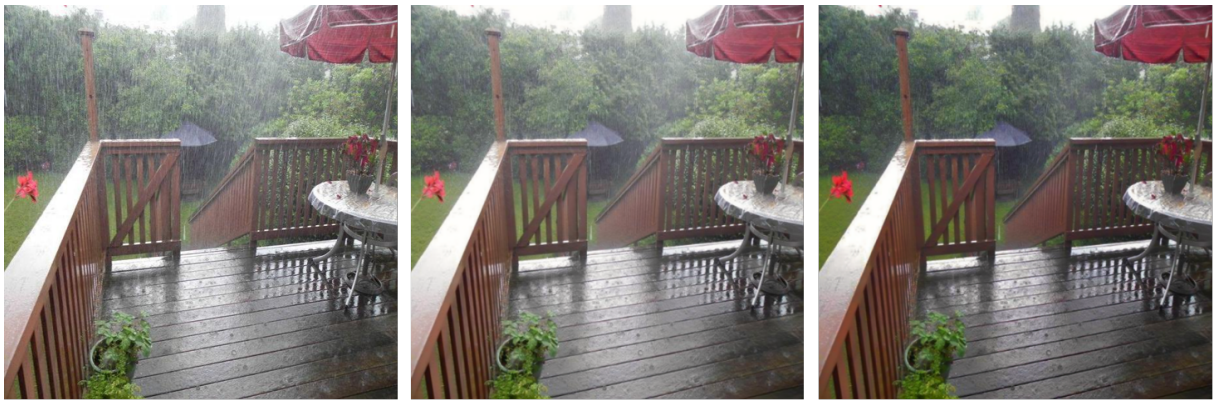


FIGURE 5.9 – Image de pluie réaliste avec un effet de brume provenant de l'article Zhang et al. (2017a) : (a) image en entrée, (b) image après suppression des traînées de pluie, (c) image après suppression de la brume avec MFP.

FIGURE 5.10 – Images de la base de données Cityscapes : (a) vérité terrain provenant de Cityscapes, (b) image avec de la pluie et du brouillard de synthèse provenant de RainCityscapes.

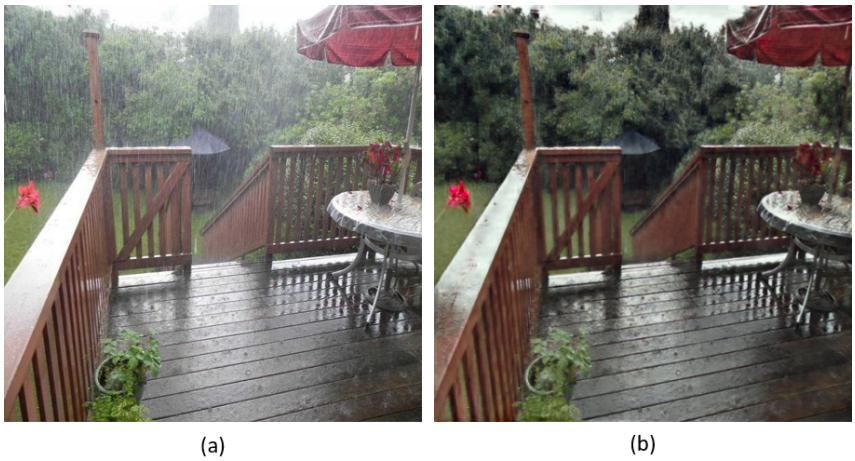


FIGURE 5.11 – Atténuation des effets de la pluie avec DA-UNet appris sur une partie de la base de données RainCityscapes : (a) image en entrée, (b) image restaurée.

Chapitre 6

Conclusion et perspectives

L'essor des systèmes d'aide à la conduite a permis des avancées significatives dans la prévention des accidents de la circulation. De plus en plus équipés de nouveaux capteurs pour la perception de l'environnement, ces systèmes sont désormais capables de gérer des tâches complexes, comme la détection des panneaux de circulation, des marquages au sol, d'autres véhicules ou des piétons. Cependant, les capteurs ne sont pas encore en mesure de gérer certaines situations dans des conditions météorologiques dégradées. En effet, par temps de brouillard ou de pluie forte, les capteurs peuvent fournir des données erronées aux systèmes, qui en conséquence, peuvent se déconnecter ou prendre des décisions inadaptées.

Dans cette thèse, des méthodes d'atténuation du brouillard ont été proposées afin de restaurer la visibilité des images et des vidéos acquises en conditions météorologiques dégradées, incluant principalement le brouillard, la brume et la poussière.

6.1 Contributions

Deux méthodes de restauration d'images et de vidéos acquises en conditions météorologiques dégradées ont été proposées. La première méthode, appelée MFP, est basée sur la loi de Koschimeder. Nous nous sommes inspirés de la méthode du *Dark Channel Prior*, une des premières dans le domaine d'atténuation du brouillard, pour réinterpréter un paramètre dans le calcul de restauration d'images selon nos hypothèses. Cette méthode est généralisable et permet d'atténuer différents types de particules venant perturber la visibilité dans les images, comme la brume, le brouillard et la poussière. Cette méthode permet également d'agir sur le brouillard de nuit et les particules sous-marines en appliquant la méthode sur chacun des canaux de couleur des images. Les évaluations menées ont montré des résultats satisfaisants sur différentes bases de données d'images. Elles ont également montré qu'il était possible de rivaliser avec des méthodes basées sur de techniques de *deep learning*. Pourtant, depuis l'essor du *deep learning*, une majorité de chercheurs en traitement d'image semble désormais considérer les méthodes plus traditionnelles comme marginales.

Les performances des algorithmes de traitement d'images basés sur des techniques de *deep learning* ne cessent de progresser ces dernières années. Cependant, une des contraintes majeures réside dans la conception ou l'acquisition d'une base de données d'apprentissage. Dans le cadre des méthodes d'atténuation de brouillard par apprentissage supervisé, les bases de données

doivent être composées de paires d'images ou de séquences d'images contenant du brouillard, et d'une vérité terrain sans brouillard. Or, il est particulièrement difficile d'obtenir des images de la même scène avec et sans brouillard dans des conditions de luminosité identiques. Pour résoudre ce problème, de nombreuses bases de données avec du brouillard de synthèse ont été créées. Cependant, ces bases de données de synthèses ne permettent pas une bonne généralisation aux images acquises en conditions réelles. Récemment, une base de données de vidéos avec du brouillard plus réaliste a été proposé pour l'apprentissage des algorithmes de *deep learning*. L'utilisation d'une machine à brouillard professionnelle permet d'obtenir un brouillard dont les effets visuels sont plus proches de la réalité que le brouillard de synthèse. J'ai créé une base de données de tests, appelée VIREDA, dans le cadre des travaux de thèse pour évaluer les performances des méthodes d'atténuation de brouillard dans des vidéos. Cette base est mise à la disposition de la communauté de recherche du domaine pour enrichir l'état de l'art. La base de données proposée a été réalisée sous différentes conditions d'éclairages et sous différentes densités de brouillard. La limite principale de ces bases de données est qu'elles ont été créées en intérieur.

Pour répondre au besoin d'atténuer les effets du brouillard dans le contexte des ADAS, nous avons décidé de nous orienter vers la réalisation d'algorithmes adaptés à la vidéo. Néanmoins, utiliser un algorithme de restauration dédié aux images et l'appliquer trame par trame à une vidéo ne permet pas d'atteindre les performances souhaitées. En effet, cette technique engendre des artefacts visuels dans la vidéo restaurée, comme des changements de couleur ou de luminosité au sein de la séquence d'images. Les performances des méthodes de *deep learning* appliquées à des vidéos nous ont confortés dans le choix de ces méthodes pour la mise en œuvre de l'algorithme d'atténuation de brouillard. De plus, la redondance temporelle des séquences d'images nous permet d'avoir une source d'information supplémentaire. J'ai dans un premier temps décidé de privilégier une méthode de « bout en bout » plutôt que de réaliser l'apprentissage des paramètres de la méthode MFP, car cette dernière introduit une étape d'estimation de la carte de transmission qui complexifie l'algorithme. De plus, il est plus difficile d'estimer les cartes de transmission d'images avec un brouillard réel.

Il existe plusieurs approches pour réaliser un algorithme de restauration de vidéos, dont celle des réseaux de neurones récurrents ou encore des *Transformers*. Finalement, nous avons mis en œuvre une méthode hybride multi-étages, appelée TCVD, qui associe deux architectures : les CNN et les *Transformers*. Cette méthode permet d'atténuer le brouillard dans des vidéos réalistes et rivalise avec les méthodes existantes d'atténuation du brouillard dans des vidéos. Cependant, le peu de code disponible publiquement des méthodes d'atténuation de brouillard dans des vidéos ne permet pas de réaliser une évaluation très poussée. L'étude d'ablation réalisée ensuite sur la méthode multi-étages a révélé que l'utilisation ou non de blocs temporels à différentes dimensions de l'encodeur ne conduit pas à montrer des différences significatives au niveau des performances.

6.2 Perspectives

L'algorithme MFP présenté au chapitre 3, a la particularité de pouvoir traiter différents types de brouillard, à la fois le brouillard dépendant de la profondeur et le brouillard spatialement uniforme. Si l'algorithme est utilisé en traitement par canal, il est capable d'atténuer le brouillard coloré, caractéristique des scènes de nuit, mais également la poussière et la pollution. Cependant, l'algorithme ne parvient pas correctement à supprimer le brouillard très dense et hétérogène, présent notamment dans les images des bases de données NTIRE. Il serait intéressant et

utile d'étendre les capacités de cet algorithme pour l'atténuation de ce type de brouillard.

L'utilisation d'un algorithme d'atténuation du brouillard image par image n'est pas le plus efficace en termes de performance pour des applications temps réel utilisées par les ADAS. Les scintillements et changements d'intensité provoqués par ce type de traitement peuvent être nuisibles à la fois pour l'interprétation de la scène les systèmes embarqués et pour le confort du conducteur. J'ai décidé alors d'orienter mes recherches vers des techniques d'atténuation du brouillard dans des vidéos. J'ai dû faire le choix entre une technique d'apprentissage des *a priori*, inspirée par la méthode MFP, et une technique d'apprentissage « de bout en bout ». J'ai privilégié le deuxième choix, car il offre davantage de possibilités de choix d'architecture. À titre de comparaison, il serait pertinant de mettre en œuvre la technique d'apprentissage des *a priori* à partir de MFP. Étant donné l'état de l'art actuel sur les techniques d'atténuation du brouillard dans les vidéos, le choix du traitement du brouillard image par image m'a permis dans un premier temps d'appréhender l'ensemble des techniques existantes. Cependant, si j'avais l'opportunité de recommencer ce travail, j'aurais davantage concentré mes recherches, et plus rapidement, sur une méthode temps réel d'atténuation du brouillard dans les vidéos.

L'architecture U-net permet de traiter des images de différentes résolutions. Bien que l'étape d'apprentissage impose d'avoir des images de petite taille (par exemple 224×224 ou 256×256), en raison des limites des performances GPU, les prédictions peuvent être réalisées sur des images de plus grande taille. Cependant, les Transformers font intervenir des types de couche (par exemple *embedding* ou *dense*) qui nécessitent une taille fixe. Les images prédites en sortie sont alors obligatoirement de la taille des images de la base d'apprentissage. Les images des séquences vidéo de REVIDE prédites en sortie, qui doivent être d'une taille 2708×1800, sont alors en basse résolution, redimensionnées à une taille d'image de 224×224. L'utilisation des blocs CONVLSTM à la place des *Transformers* permet la prédiction des images de plus grande résolution. Malgré cela, pour que nous soyons en mesure d'utiliser la méthode avec les *Transformers*, il serait nécessaire d'ajouter une étape de post-traitement pour augmenter la résolution des images en sortie du réseau qui utilise les blocs TpFormer.

Enfin, la méthode de restauration en vidéo mise en œuvre dans le chapitre 4 fournit des résultats satisfaisants dans son état actuel. Cependant, la méthode est encore perfectible et mériterait d'être davantage améliorée. Il serait intéressant d'explorer l'utilisation des blocs ConvLSTM.

La base de données VIREDA, créée et présentée au chapitre 4, a été très utile à l'évaluation de l'algorithme d'atténuation de vidéos. Les différents jeux d'éclairage ainsi que les densités disponibles ont permis de la diversifier. Cependant, une seule scène est disponible, ce qui rend la base trop petite pour l'utiliser en tant que base d'apprentissage. Il faudrait pour cela effectuer la même procédure d'acquisition vidéo réalisée, mais avec plusieurs scènes différentes (d'autres décors et d'autres objets mobiles, par exemple).

Pour évaluer la consistance temporelle au sein des vidéos, j'ai calculé la moyenne entre des images successives. Les changements de couleur et de luminosité entre les trames ont impacté la valeur utilisée ε. Plus le changement est important, plus sa valeur est élevée. Cette métrique donne une indication sur la consistance temporelle, mais il aurait été intéressant d'ajouter davantage de métriques temporelles pour obtenir une évaluation plus poussée. L'ajout d'une fonction temporelle dans le calcul du Loss pourrait également être envisagé.

Les performances GPU à ma disposition m'ont contrainte à redimensionner les images de la base de données pour l'apprentissage de l'algorithme. Ceci implique une faible résolution en sortie de l'algorithme. Or, acquérir des images de grandes résolutions est important aujourd'hui, tant pour la visualisation que pour la qualité des images, et ainsi garantir les performances des algorithmes de détection.

Nous avons vu dans le dernier chapitre que la pluie en conditions réelles laisse un aspect brumeux à l'horizon. Les algorithmes d'atténuation du brouillard peuvent donc être utilisés pour ce type de scènes. Cependant, ces algorithmes ne suppriment pas les traînées de pluie présentes au premier plan. L'utilisation de la base de données de synthèse Rain-Cityscape (brouillard + pluie) et d'une architecture auto-encodeur a permis d'obtenir des résultats satisfaisants. Les résultats obtenus avec les méthodes d'atténuation de brouillard sur les bases REVIDE et VIREDA ont montré que le brouillard produit par une machine à brouillard est similaire au brouillard réel. Il serait intéressant et utile de créer une base de données où il est possible de d'ajouter à la fois du brouillard et de la pluie.

Bibliographie

(1992). *International Meteorological Vocabulary*. World Meteorological Organization. 2nd ed. Geneva.

AFNOR (1998). Météorologie routière - recueil des données météorologiques et routiers, NF p99-320. https://www.boutique.afnor.org/fr-fr/norme/nf-p99320/meteorologie-routiere-recueil-des-donnees-meteorologiques-et-routiers/fa045352/15790.

Ancuti, C., Ancuti, C. O., and De Vleeschouwer, C. (2016a). D-HAZY : A dataset to evaluate quantitatively dehazing algorithms. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 2226–2230. ISSN : 2381-8549.

Ancuti, C., Ancuti, C. O., De Vleeschouwer, C., and Bovik, A. C. (2016b). Night-time dehazing by fusion. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 2256–2260. ISSN : 2381-8549.

Ancuti, C., Ancuti, C. O., De Vleeschouwer, C., and Bovik, A. C. (2020). Day and Night-Time Dehazing by Local Airlight Estimation. *IEEE Transactions on Image Processing*, 29 :6264–6275.

Ancuti, C. O., Ancuti, C., Timofte, R., and De Vleeschouwer, C. (2018). O-HAZE : A dehazing benchmark with real hazy and haze-free outdoor images. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 867–8678. ISSN : 2160-7516.

Ancuti, C. O., Ancuti, C., Timofte, R., Van Gool, L., Zhang, L., Yang, M.-H., Guo, T., Li, X., Cherukuri, V., Monga, V., Jiang, H., Yang, S., Liu, Y., Qu, X., Wan, P., Park, D., Chun, S. Y., Hong, M., Huang, J., Chen, Y., Chen, S., Wang, B., Michelini, P. N., Liu, H., Zhu, D., Liu, J., Santra, S., Mondal, R., Chanda, B., Morales, P., Klinghoffer, T., Quan, L. M., Kim, Y.-G., Liang, X., Li, R., Pan, J., Tang, J., Purohit, K., Suin, M., Rajagopalan, A., Schettini, R., Bianco, S., Piccoli, F., Cusano, C., Celona, L., Hwang, S., Ma, Y. S., Byun, H., Murala, S., Dudhane, A., Aulakh, H., Zheng, T., Zhang, T., Qin, W., Zhou, R., Wang, S., Tarel, J.-P., Wang, C., and Wu, J. (2019). NTIRE 2019 image dehazing challenge report. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2241–2253. IEEE.

Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lucic, M., and Schmid, C. (2021). ViViT : A Video Vision Transformer. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6816–6826, Montreal, QC, Canada. IEEE.

Bahdanau, D., Cho, K., and Bengio, Y. (2016). Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv preprint arXiv :1409.0473*.

Broyer, M. (2017). What the Fog ? https://www.behance.net/gallery/56423743/What-the-Fog.

Cai, B., Xu, X., Jia, K., Qing, C., and Tao, D. (2016). DehazeNet : An End-to-End System for Single Image Haze Removal. *IEEE Transactions on Image Processing*, 25(11) :5187–5198.

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6) :679–698.

Caraffa, L., Tarel, J.-P., and Charbonnier, P. (2015). The Guided Bilateral Filter : When the Joint/Cross Bilateral Filter Becomes Robust. *IEEE Transactions on Image Processing*, 24(4) :1199–1208.

Cavallo, V., Doré, J., and Colomb, M. (2000). Perception of vehicle distance in fog and fog light design. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 44(20) :3–377.

Chen, D., He, M., Fan, Q., Liao, J., Zhang, L., Hou, D., Yuan, L., and Hua, G. (2019). Gated Context Aggregation Network for Image Dehazing and Deraining. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*.

Chen, Z., Wang, Y., Yang, Y., and Liu, D. (2021). PSD : Principled synthetic-to-real dehazing guided by physical priors. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7176–7185. IEEE.

Cheng, J., Dong, L., and Lapata, M. (2016). Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

CIE (1987). International lighting vocabulary. https://cie.co.at/publications/international-lighting-vocabulary.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223.

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., and Wei, Y. (2017). Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773.

Dong, Y., Liu, Y., Zhang, H., Chen, S., and Qiao, Y. (2020). Fd-gan : Generative adversarial networks with fusion-discriminator for single image dehazing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10729–10736.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An Image is Worth 16x16 Words : Transformers for Image Recognition at Scale. *International Conference on Learning Representations*.

Duminil, A., Tarel, J.-P., and Brémond, R. (2021a). Single image atmospheric veil removal using new priors. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 1719–1723.

Duminil, A., Tarel, J.-P., and Brémond, R. (2021b). Single image atmospheric veil removal using new priors for better genericity. *Atmosphere*, 12(6) :772.

Ge, W., Lin, Y., Wang, Z., Wang, G., and Tan, S. (2021). An Improved U-Net Architecture for Image Dehazing. *IEICE Transactions on Information and Systems*, E104.D(12) :2218–2225.

Golts, A., Freedman, D., and Elad, M. (2020). Unsupervised single image dehazing using dark channel prior loss. *IEEE Transactions on Image Processing*, 29 :2692–2701.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Halmaoui, H. (2012). Restauration d'images par temps de brouillard et de pluie : applications aux aides à la conduite.

Hautière, N., Aubert, D., Dumont, E., and Tarel, J.-P. (2006). Validation expérimentale de méthodes dédiées à l'estimation embarquée de la visibilité atmosphérique. *Actes des journées scientifiques du LCPC*, page 6.

Hautière, N., Tarel, J.-P., and Aubert, D. (2007). In *Towards Fog-Free In-Vehicle Vision Systems through Contrast Restoration*, IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07), pages 1–8.

He, K., Sun, J., and Tang, X. (2009). Single image haze removal using dark channel prior. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1956–1963. IEEE.

He, K., Sun, J., and Tang, X. (2010). Guided image filtering. In *European conference on computer vision*, pages 1–14. Springer.

He, K., Sun, J., and Tang, X. (2011). Single Image Haze Removal Using Dark Channel Prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12) :2341–2353.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Hirschmuller, H. and Scharstein, D. (2007). Evaluation of cost functions for stereo matching. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE.

Hnewa, M. and Radha, H. (2021). Multiscale Domain Adaptive YOLO for Cross-Domain Object Detection. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 3323–3327.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-term Memory. *Neural computation*, 9 :1735–80.

Hu, X., Zhu, L., Wang, T., Fu, C.-W., and Heng, P.-A. (2021). Single-image real-time rain removal based on depth-guided non-local features. *IEEE Transactions on Image Processing*, pages 1759–1770.

Huang, S. C., Le, T. H., and Jaw, D. W. (2020). Dsnet : Joint semantic learning for object detection in inclement weather conditions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1.

Johnson, J., Alahi, A., and Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. *European conference on computer vision*, pages 694–711.

Kadambi, A., Bhandari, A., and Raskar, R. (2014). 3D Depth Cameras in Vision : Benefits and Limitations of the Hardware. In *Computer vision and machine learning with RGB-D sensors*, pages 3–26.

Kim, J.-H., Jang, W.-D., Park, Y., Lee, D.-H., Sim, J.-Y., and Kim, C.-S. (2012). Temporally x real-time video dehazing. In *2012 19th IEEE International Conference on Image Processing*, pages 969–972. ISSN : 2381-8549.

Koschmieder, H. (1924). Theorie der horizontalen sichtweite. *Beitrage zur Physik der freien Atmosphare*, pages 33–53.

Li, B., Peng, X., Wang, Z., Xu, J., and Feng, D. (2017). An All-in-One Network for Dehazing and Beyond. *arXiv :1707.06543 [cs]*.

Li, B., Peng, X., Wang, Z., Xu, J., and Feng, D. (2018a). End-to-end united video dehazing and detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).

Li, B., Ren, W., Fu, D., Tao, D., Feng, D., Zeng, W., and Wang, Z. (2019a). Benchmarking single-image dehazing and beyond. *IEEE Transactions on Image Processing*, 28(1) :492–505.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. (2018b). Visualizing the Loss Landscape of Neural Nets. *Advances in neural information processing systems*, 31.

Li, R. (2021). Progressive deep video dehazing without explicit alignment estimation. *arXiv :2107.07837 [cs]*.

Li, X., Chen, S., Hu, X., and Yang, J. (2019b). Understanding the disharmony between dropout and batch normalization by variance shift. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2682–2690.

Li, X., Hua, Z., and Li, J. (2022). Two-stage single image dehazing network using swin-transformer. *IET Image Processing*, page ipr2.12506.

Li, Y., Tan, R. T., and Brown, M. S. (2015). Nighttime Haze Removal with Glow and Multiple Light Colors. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 226–234, Santiago, Chile. IEEE.

Liang, J., Cao, J., Fan, Y., Zhang, K., Ranjan, R., Li, Y., Timofte, R., and Van Gool, L. (2022). VRT : A Video Restoration Transformer. *arXiv preprint arXiv :2201.12288*.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., and Dollár, P. (2014). Microsoft COCO : Common objects in context. In *European conference on computer vision*, pages 740–755.

Lin Zhang, Lei Zhang, Xuanqin Mou, and Zhang, D. (2011). FSIM : A Feature Similarity Index for Image Quality Assessment. *IEEE Transactions on Image Processing*, 20(8) :2378–2386.

Liu, W., Ren, G., Yu, R., Guo, S., Zhu, J., and Zhang, L. (2022a). Image-Adaptive YOLO for Object Detection in Adverse Weather Conditions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(2). Number : 2.

Liu, Y., Pan, J., Ren, J., and Su, Z. (2019). Learning deep priors for image dehazing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2492–2500.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. (2022b). Swin transformer : Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022.

Lou, W., Li, Y., Yang, G., Chen, C., Yang, H., and Yu, T. (2020). Integrating Haze Density Features for Fast Nighttime Image Dehazing. *IEEE Access*, 8 :113318–113330.

Middleton, W. E. K. (1952). *Vision through the atmosphere*. University of Toronto Press.

Mustafa, A., Mikhailiuk, A., Iliescu, D. A., Babbar, V., and Mantiuk, R. K. (2021). Training a task-specific image reconstruction loss. *In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2 :2319–2328.

Naka, K. I. and Rushton, W. a. H. (1966). S-potentials from colour units in the retina of fish (Cyprinidae). *The Journal of Physiology*, 185(3).

Nayar, S. and Narasimhan, S. (1999). Vision in bad weather. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 820–827, Kerkyra, Greece. IEEE.

Negru, M., Nedevschi, S., and Peter, R. I. (2015). Exponential Contrast Restoration in Fog Conditions for Driving Assistance. *IEEE Transactions on Intelligent Transportation Systems*, 16(4) :2257–2268.

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., and Walsh, J. (2020). Deep learning vs. traditional computer vision. In Arai, K. and Kapoor, S., editors, *Advances in Computer Vision*, volume 943, pages 128–144. Springer International Publishing.

Park, Y. and Kim, T.-H. (2018). Fast execution schemes for dark-channel-prior-based outdoor video dehazing. *IEEE Access*, 6 :10003–10014.

Prechelt, L. (1997). Early stopping - but when ? In *Neural Networks : Tricks of the trade*, pages 55–69.

Qin, X., Wang, Z., Bai, Y., Xie, X., and Jia, H. (2019). FFA-net : Feature fusion attention network for single image dehazing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11908–11915.

Qu, Y., Chen, Y., Huang, J., and Xie, Y. (2019). Enhanced pix2pix dehazing network. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8152–8160. IEEE.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once : Unified, real-time object detection. Proceedings of the IEEE conference on computer vision and pattern recognition, pages 779–788.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-CNN : Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 8.

Ren, W., Liu, S., Zhang, H., Pan, J., Cao, X., and Yang, M.-H. (2016). Single Image Dehazing via Multi-scale Convolutional Neural Networks. In Leibe, B., Matas, J., Sebe, N., and Welling, M., editors, *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science, pages 154–169, Cham. Springer International Publishing.

Ren, W., Pan, J., Zhang, H., Cao, X., and Yang, M.-H. (2020). Single image dehazing via multi-scale convolutional neural networks with holistic edges. *International Journal of Computer Vision*, 128(1) :240–259.

Ren, W., Zhang, J., Xu, X., Ma, L., Cao, X., Meng, G., and Liu, W. (2019). Deep video dehazing with semantic segmentation. *IEEE Transactions on Image Processing*, 28(4) :1895–1908.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net : Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X., and Westling, P. (2014). High-Resolution Stereo Datasets with Subpixel-Accurate Ground Truth. In Jiang, X., Hornegger, J., and Koch, R., editors, *Pattern Recognition*, Lecture Notes in Computer Science, pages 31–42, Cham. Springer International Publishing.

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-k., and Woo, W.-c. (2015). Convolutional LSTM network : A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28.

Silberman, N., Hoiem, D., Kohli, P., and Fergus, R. (2012). Indoor Segmentation and Support Inference from RGBD Images. In Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., and Schmid, C., editors, *Computer Vision – ECCV 2012*, Lecture Notes in Computer Science, pages 746–760, Berlin, Heidelberg. Springer.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *In International Conference on Learning Representations*.

Song, Y., He, Z., Qian, H., and Du, X. (2022). Vision transformers for single image dehazing. *arXiv :2204.03883 [cs]*.

Tan, R. T., Pettersson, N., and Petersson, L. (2007). Visibility Enhancement for Roads with Foggy or Hazy Scenes. In *2007 IEEE Intelligent Vehicles Symposium*, pages 19–24, Istanbul, Turkey. IEEE. ISSN : 1931-0587.

Tarel, J.-P., Hautiere, N., Cord, A., Gruyer, D., and Halmaoui, H. (2010). Improved visibility of road scene images under heterogeneous fog. In *2010 IEEE Intelligent Vehicles Symposium*, pages 478–485. IEEE.

Tarel, J.-P., Hautière, N., Caraffa, L., Cord, A., Halmaoui, H., and Gruyer, D. (2012). Vision Enhancement in Homogeneous and Heterogeneous Fog. *IEEE Intelligent Transportation Systems Magazine*, 4(2) :6–20.

Tarel, Jean-Philippe et Hautière, N. (2009). Fast visibility restoration from a single color or gray level image. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2201–2208, Kyoto. IEEE.

Tomasi, C. and Manduchi, R. (1998). Bilateral filtering for gray and color images. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 839–846.

Tremblay, M., Halder, S., De Charette, R., and Lalonde, J.-F. (2020). Rain rendering for evaluating and improving robustness to bad weather. *International Journal on Computer Vision (IJCV)*, pages 341–360.

Valanarasu, J. M. J., Yasarla, R., and Patel, V. M. (2021). TransWeather : Transformer-based restoration of images degraded by adverse weather conditions. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2353–2363.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention Is All You Need. *Advances in neural information processing systems*, 30.

VidalMata, R. G., Banerjee, S., RichardWebster, B., Albright, M., Davalos, P., McCloskey, S., Miller, B., Tambo, A., Ghosh, S., Nagesh, S., Yuan, Y., Hu, Y., Wu, J., Yang, W., Zhang, X., Liu, J., Wang, Z., Chen, H.-T., Huang, T.-W., Chin, W.-C., Li, Y.-C., Lababidi, M., Otto, C., and Scheirer, W. J. (2021). Bridging the Gap Between Computational Photography and Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12) :4272–4290.

Wang, H., Wu, Y., Li, M., Zhao, Q., and Meng, D. (2019a). A survey on rain removal from video and single image. *Science China Information Sciences*, 65.

Wang, T.-C., Liu, M.-Y., Zhu, J.-Y., Tao, A., Kautz, J., and Catanzaro, B. (2018). High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807.

Wang, X., Chan, K. C., Yu, K., Dong, C., and Loy, C. C. (2019b). EDVR : Video restoration with enhanced deformable convolutional networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1954–1963. IEEE.

Wang, Y., Song, W., Fortino, G., Qi, L.-Z., Zhang, W., and Liotta, A. (2019c). An Experimental-Based Review of Image Enhancement and Image Restoration Methods for Underwater Imaging. *IEEE Access*, 7.

Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. (2004). Image Quality Assessment : From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4) :600–612.

Wang, Z., Simoncelli, E., and Bovik, A. (2003). Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, volume 2, pages 1398–1402 Vol.2.

Wu, Y. and He, K. (2018). Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19.

Yang, D. and Sun, J. (2018). Proximal Dehaze-Net : A Prior Learning-Based Deep Network for Single Image Dehazing. In Ferrari, V., Hebert, M., Sminchisescu, C., and Weiss, Y., editors, *Computer Vision – ECCV 2018*, volume 11211, pages 729–746. Springer International Publishing, Cham.

Yang, D. and Sun, J. (2021). A model-driven deep dehazing approach by learning deep priors. *IEEE Access*, 9 :108542–108556.

Yang, W., Tan, R. T., Feng, J., Liu, J., Guo, Z., and Yan, S. (2017). Deep joint rain detection and removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1357–1366.

Yu, T., Song, K., Miao, P., Yang, G., Yang, H., and Chen, C. (2019). Nighttime single image dehazing via pixel-wise alpha blending. *IEEE Access*, 7 :114619–114630.

Zhang, H. and Patel, V. M. (2018). Density-aware single image de-raining using a multi-stream dense network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 695–704.

Zhang, H., Sindagi, V., and Patel, V. M. (2017a). Image de-raining using a conditional generative adversarial network. *IEEE transactions on circuits and systems for video technology*, 30 :3943–3956.

Zhang, J., Cao, Y., Fang, S., Kang, Y., and Chen, C. W. (2017b). Fast Haze Removal for Nighttime Image Using Maximum Reflectance Prior. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. ISSN : 1063-6919.

Zhang, J., Cao, Y., Zha, Z.-J., and Tao, D. (2020a). Nighttime dehazing with a synthetic benchmark. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2355–2363.

Zhang, J., Li, L., Zhang, Y., Yang, G., Cao, X., and Sun, J. (2011). Video dehazing with spatial and temporal coherence. *The Visual Computer*, 27 :749–757.

Zhang, S., Tuo, H., Hu, J., and Jing, Z. (2021). Domain adaptive yolo for one-stage cross-domain detection. In *Asian Conference on Machine Learning*, pages 785–797. PMLR.

Zhang, X., Dong, H., Pan, J., Zhu, C., Tai, Y., Wang, C., Li, J., Huang, F., and Wang, F. (2022). Learning to restore hazy video : A new real-world dataset and a new method. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9235–9244. IEEE.

Zhang, Y., Chen, Z., and Liu, S. (2020b). Video super resolution using temporal encoding ConvLSTM and multi-stage fusion. In *2020 IEEE International Conference on Visual Communications and Image Processing (VCIP)*, pages 298–301. ISSN : 2642-9357.

Zhang, Y., Ding, L., and Sharma, G. (2017c). HazeRD : An outdoor scene dataset and benchmark for single image dehazing. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3205–3209. ISSN : 2381-8549.

Zhao, D., Li, J., Li, H., and Xu, L. (2022). Complementary feature enhanced network with vision transformer for image dehazing. *arXiv :2109.07100 [cs]*.

Zhong, Z., Gao, Y., Zheng, Y., Zheng, B., and Sato, I. (2021). Efficient spatio-temporal recurrent neural network for video deblurring. *European Conference on Computer Vision*, pages 191–207.

Zhu, M., He, B., and Wu, Q. (2018). Single Image Dehazing Based on Dark Channel Prior and Energy Minimization. *IEEE Signal Processing Letters*, 25(2) :174–178.

Zhu, Z., Wei, H., Hu, G., Li, Y., Qi, G., and Mazur, N. (2021). A Novel Fast Single Image Dehazing Algorithm Based on Artificial Multiexposure Image Fusion. *IEEE Transactions on Instrumentation and Measurement*, 70 :1–23.