



HAL
open science

Restauration d'images par temps de brouillard et de pluie : applications aux aides à la conduite

Houssam Halmaoui

► **To cite this version:**

Houssam Halmaoui. Restauration d'images par temps de brouillard et de pluie : applications aux aides à la conduite. Traitement du signal et de l'image [eess.SP]. Université d'Evry-Val d'Essonne, 2012. Français. NNT: . tel-00830869

HAL Id: tel-00830869

<https://theses.hal.science/tel-00830869>

Submitted on 5 Jun 2013

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

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



IFSTTAR LEPSIS LIVIC



THÈSE

pour obtenir le titre de

Docteur

de l'université d'Évry-Val-d'Essonne
Spécialité : Traitement d'images

Présentée par

Houssam HALMAOUI

Restauration d'images par temps de brouillard et de pluie : applications aux aides à la conduite

Thèse dirigée par Didier AUBERT

préparée à l'IFSTTAR, projet ICADAC

soutenue le 30 novembre 2012

Jury :

<i>Rapporteurs :</i>	Fabrice MÉRIAUDEAU	-	Professeur IUT Le Creusot
	Frédéric CHAUSSE	-	Enseignant Chercheur LASMEA
<i>Directeur :</i>	Didier AUBERT	-	Dir. de recherche IFSTTAR
<i>Examineurs :</i>	Michel DEVY	-	Dir. de recherche LAAS/CNRS
	Samia BOUCHAFA	-	Professeur IBISC
	Nicolas HAUTIERE	-	Encadrant-Chercheur IFSTTAR
	Aurélien CORD	-	Encadrant-Chercheur IFSTTAR

Table des matières

1	Introduction	1
2	Etat de l'art - Restauration des images -	9
2.1	Cas général	9
2.1.1	Chaîne d'acquisition	9
2.1.2	Dégradation des images	11
2.1.3	Débruitage	12
2.1.3.1	Filtres classiques	13
2.1.3.2	Statistiques locales de l'image	13
2.1.3.3	Filtre de Wiener	15
2.1.3.4	Filtre Bi-latéral	15
2.1.3.5	Diffusion	16
2.1.3.6	Total Variation Model	19
2.1.3.7	Approches bayésiennes	20
2.1.3.8	Transformée en ondelettes	21
2.1.4	Rehaussement du contraste	22
2.1.4.1	Égalisation d'histogramme	22
2.1.4.2	Égalisation d'histogramme locale	23
2.1.4.3	Retinex	23
2.1.4.4	Multi Scale Retinex	25
2.1.5	Balance des blancs	25
2.1.5.1	Gray World	26
2.1.5.2	Max White	27
2.2	Images dégradées par le brouillard	27
2.2.1	Modèle optique du brouillard	28
2.2.1.1	Coefficient d'extinction atmosphérique	29
2.2.1.2	Diffusion et longueur d'onde	29
2.2.1.3	Atténuation	30
2.2.1.4	Le voile atmosphérique	31
2.2.2	Méthodes Interactives	33
2.2.2.1	Restauration en utilisant un filtre polarisant	33
2.2.2.2	Deux images acquises dans des conditions météoro- logiques différentes	36
2.2.2.3	Utilisation d'une seule image et d'informations four- nies par l'utilisateur	40
2.2.3	Méthodes automatiques utilisant une seule image	42
2.2.3.1	Hypothèse monde plan	43
2.2.3.2	Utilisation du modèle chromatique	45
2.2.3.3	Dark Channel	49

2.2.3.4	Utilisation de contraintes sur le voile atmosphérique	52
2.3	Images dégradées par les gouttes de pluie	53
2.3.1	Physique des gouttes de la pluie	54
2.3.2	Détection des gouttes de pluie	56
2.3.2.1	Méthode s'appuyant sur le modèle physique des gouttes	56
2.3.2.2	Détection des gouttes de pluie focalisées ayant un fort gradient	58
2.4	Conclusion	60
3	Outils et méthodes d'évaluation des restaurations	61
3.1	Base de données d'images	61
3.1.1	Images de synthèse	61
3.1.1.1	Logiciel SiVIC	62
3.1.1.2	Bases de données d'images routières de brouillard	63
3.1.1.3	Images de pluie	64
3.1.2	Ajout du brouillard sur des images réelles	65
3.2	Méthodes d'évaluation quantitative	66
3.2.1	Utilisation de la vérité terrain	66
3.2.2	Problème de pertes d'informations dans le brouillard	67
3.2.3	Indicateurs fondés sur les contrastes visibles à 5 %	69
3.2.4	Évaluation sur des ADAS	72
3.3	Vidéos ICADAC	72
3.4	Conclusion	72
4	Restauration d'images de brouillard	75
4.1	Approche proposée	75
4.1.1	Algorithme	76
4.1.1.1	Densité du brouillard	76
4.1.1.2	Intensité du ciel	76
4.1.1.3	Restauration locale du contraste	78
4.1.1.4	Ajustement global de l'intensité	81
4.1.1.5	Carte de transmittance	81
4.1.1.6	Exploitation de l'information couleur pour améliorer la carte de transmittance	81
4.1.1.7	Lissage de la carte de transmittance	82
4.1.1.8	Fusion avec approche monde plan	84
4.1.2	Optimisation des paramètres : Recuit simulé	85
4.1.3	Évaluation	89
4.1.3.1	Résultats qualitatifs	89
4.1.3.2	Résultats quantitatifs	91
4.2	Amélioration du rendu visuel	94
4.2.1	Accroissement de la dynamique	94
4.2.2	Lissage de la carte de transmittance par un filtre bilatéral	97

4.2.3	Couleur	100
4.3	Conclusion	100
5	Restauration d'images de pluie	103
5.1	Introduction	103
5.2	Généralités sur l'“inpainting”	104
5.3	État de l'art	105
5.3.1	“Inpainting” de régions contenant des structures	105
5.3.2	“Inpainting” rapide par diffusion isotropique	106
5.3.3	Synthèse de texture	108
5.3.4	“Inpainting” vidéo	111
5.4	Méthodes sélectionnées	112
5.4.1	Algorithme de Telea	113
5.4.2	Algorithme de Bornemann	115
5.5	Optimisation des paramètres sur des images routières	120
5.6	Conclusion	123
6	Applications	125
6.1	Pré-traitement : détection des panneaux	125
6.1.1	Algorithme de détection	126
6.1.2	Paramètres de la méthode de restauration	128
6.1.3	Base de données de panneaux	128
6.1.4	Résultats expérimentaux	130
6.1.4.1	Mesure des performances	130
6.1.4.2	Performances par temps clair	131
6.1.4.3	Impact de la densité du brouillard sur la perception des panneaux	131
6.1.4.4	Amélioration par restauration d'images	133
6.1.4.5	Amélioration de la gamme de fonctionnement des ADAS	134
6.1.5	Conclusion	135
6.2	Assistance : afficheur tête haute dans le brouillard	136
6.2.1	Temps de réaction	136
6.2.1.1	Loi de Piéron	137
6.2.1.2	Temps de réaction dans le brouillard	137
6.2.2	Modélisation du brouillard et mesure du temps de perception	138
6.2.2.1	Calibrage photométrique	140
6.2.2.2	Logiciel Presentation	140
6.2.2.3	Modélisation du brouillard	140
6.2.2.4	Résultats	141
6.2.3	Amélioration du temps de perception dans le brouillard	144
6.2.3.1	Modélisation d'images routières dans le brouillard	144
6.2.3.2	Déroulement de l'expérience	145
6.2.3.3	Résultats	146

6.2.4	Conclusion	148
6.3	Traitement : Détection de l'espace navigable	149
6.3.1	Estimation de la distance de visibilité	149
6.3.2	Discussion	150
6.3.3	Segmentation de la ROI fondée sur les cartes géodésique . . .	151
6.3.4	Détection de l'espace navigable	152
6.3.5	Conclusion	154
6.4	Conclusion	154
7	Conclusion et perspectives	157
A	“Inpainting”	161
A.1	Illustration de la méthode d'inpainting retenue par des exemples . .	161
A.2	Inpainting pour le débruitage	163
	Bibliographie	167

Introduction

Contexte

La sécurité routière n'a pas cessé de s'améliorer au cours des 20 dernières années. Cela est dû principalement aux progrès qu'a connus le secteur de l'industrie automobile, à l'amélioration des réseaux routiers et à l'utilisation de systèmes efficaces de gestion de trafic. Cependant, le nombre de tués sur la route reste assez élevé : environ 4000 par an en France et environ 40000 en Europe. D'après les études d'accidentologie française, plus de 90 % des accidents corporelles sont liés au conducteur (vitesse, fatigue, somnolence, distraction etc.) ou en association avec d'autres facteurs. L'objectif principal des systèmes d'aide à la conduite avancées ou ADAS (Advanced Driver Assistant System) est de réduire cette erreur humaine et donc d'améliorer la sécurité des passagers.

En termes de sécurité les ADAS se divisent en deux principaux groupes : les systèmes passifs qui tentent de minimiser la gravité d'un accident au moment où celui-ci se produit (prétensionneur, airbags passager ou piéton) et les systèmes actifs qui essaient d'éviter qu'un accident ait lieu (ACC, ESP, freinage d'urgence). Il existe aussi un autre groupe dit systèmes de sécurité tertiaires qui se déclenchent après l'accident (appel des secours, par exemple).

Grâce aux progrès technologique, l'augmentation des capacités de calcul et la baisse des prix des composants électroniques, les ADAS ont beaucoup évolué depuis leur introduction. En particulier, les nouveaux systèmes ne sont plus limités à l'habitacle du véhicule (capteurs proprioceptifs) et permettent de récolter des informations sur l'environnement du véhicule (capteurs extéroceptifs). Nous trouvons d'un côté des capteurs tels que le SONAR, le RADAR, le LASER et les caméras dans le domaine visible et infrarouge pour l'environnement proche et les systèmes de communication avec l'infrastructure ou avec d'autres véhicules pour l'environnement lointain. Le développement massif de ces systèmes a donné naissance à une nouvelle forme de véhicules autonomes (automatisés à 100 %), visant à rendre la tâche de conduite plus sûre et plus confortable (le conducteur est de moins en moins sollicité pour la prises de décisions). La figure 1.1 montre un exemple de ces véhicules. Ils ne sont pas encore commercialisés en Europe mais permettent d'évaluer les ADAS dans différentes conditions.

Dans le cadre de cette thèse, nous nous intéressons plus particulièrement aux ADAS utilisant une caméra placée derrière le pare-brise.

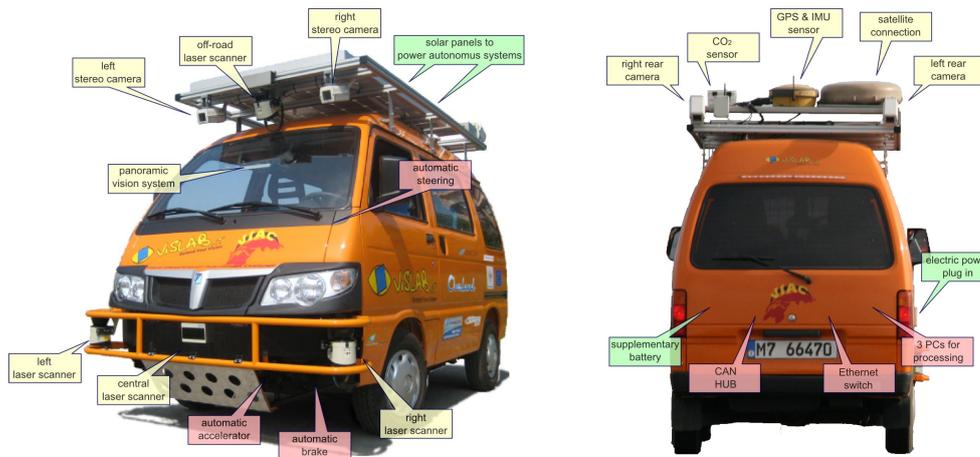


FIGURE 1.1 – Véhicule autonome utilisé dans le challenge VIAC (The Vislab Intercontinental Autonomous Challenge), dont le but était de parcourir en conduite automatique un trajet d'environ 13 000 km, partant d'Italie jusqu'à la Chine. Le véhicule utilise 7 caméras, 4 lasers, un GPS, une centrale inertielle et des moyens de communication inter-véhicules.

Caméra embarquée

La caméra est un capteur qui fournit beaucoup d'informations mais qui nécessite des traitements complexes.

L'intérêt d'utiliser une caméra est qu'elle nous renvoie une image similaire de la scène à ce que le conducteur perçoit, sachant qu'une grande quantité des informations utilisées par le conducteur sont visuelles. Nous allons donc doter le véhicule d'un œil (caméra) et l'ADAS va traiter les informations qu'il reçoit afin d'assister le conducteur. Cela peut s'envisager de la même façon qu'un passager, ou co-pilote, qui pourrait lui indiquer, par exemple, la présence d'un danger qu'il n'avait pas perçu. La vision par ordinateur permet d'extraire différentes informations du signal caméra pour réaliser de multiples applications en utilisant un seul capteur et donc de réduire le coût et l'encombrement de ces systèmes. Il est même possible d'aller plus loin : donner l'ordre à un automate de prendre la commande de certaines fonctionnalités du véhicule.

D'autre part, le fait d'embarquer des algorithmes complexes de traitement d'images devient tout à fait possible de nos jours grâce aux nombreux calculateurs puissants présents sur le marché (FPGA, DSP, GPU...). Ce n'était pas le cas il y a une dizaine d'années à cause du temps de traitement prohibitif et des capacités de mémoire que nécessitent la plupart des méthodes avancées de vision par ordinateur.

De nombreuses applications ADAS utilisant une caméra ont vu le jour : détection et reconnaissance des signalisations verticales et horizontales, détection des piétons, détection et suivi de véhicules... Elles permettent d'avertir le conducteur en cas de danger (sortie de voie involontaire, dépassement de la vitesse limite...). La figure 1.2 montre des exemples de ces systèmes.



FIGURE 1.2 – En haut, à gauche, système de reconnaissance des panneaux à l'aide d'une caméra et d'un GPS ; projection sur le pare-brise et/ou sur le système de navigation. En haut, à droite, affichage tête haute : informations sur la route ainsi que des instructions et des avertissements. En bas, détection de piétons par caméra infrarouge.

Influence de la météo

Conducteur

D'après les études d'accidentologie, 37 % des accidents en Europe se produisent dans le cas d'une visibilité réduite (brouillard, nuit...). Les accidents dus au brouillard ne sont pas très nombreux, mais lorsqu'ils se produisent leur gravité est plus importante et le risque de carambolage augmente.

Le brouillard a pour effet de réduire la distance de visibilité. Dans ce cas, alors qu'on s'attend à ce que le conducteur réduise sa vitesse, différentes études prouvent le contraire. D'après [Cavallo 2000], à cause des effets visuels du brouillard, un phénomène de surestimation des distances dans le brouillard est susceptible de contribuer à la réduction des intervalles entre véhicules : surestimation de 25 à 50 % dans le cas d'un brouillard dense et autour de 6 % dans le cas d'un brouillard de densité moyenne. À cause de cette surestimation, la vitesse de conduite légale peut être dépassée. Plusieurs expériences ont été réalisées pour comparer la perception des distances et des vitesses par le conducteur dans le cas d'un temps clair et dans le cas du brouillard. [Sumner 1977] montrent que l'ajustement de vitesse effectué dans le brouillard en fonction de la distance de visibilité est plus de 50 % du temps insuffisant pour permettre au conducteur d'effectuer un freinage à une distance sécurisée. De la même façon, [Snowden 1998] ont démontré qu'il existe une sous-estimation de la vitesse qui entraîne un dépassement de

la vitesse réglementaire et que plus le brouillard est dense plus la vitesse augmente. [Tenkink 1988] montre qu'il peut y avoir un franchissement de ligne à cause de la vision latérale qui est dégradée. [Barham 2001] montre que le conducteur a tendance à suivre de trop près le véhicule de devant afin de ne pas le perdre de vue.

À l'inverse du brouillard, la pluie est une situation météorologique dynamique composée de particules plus larges qui peuvent être vues individuellement. Les effets visuels générés par la pluie sont plus complexes puisque la distribution et la forme des gouttes varient de manière aléatoire, spatialement et temporellement.

Outre la diminution de l'adhérence, qui entraîne une augmentation de la distance de freinage, en présence de pluie, la visibilité du conducteur est réduite à cause de plusieurs facteurs : gouttes sur le pare-brise, passage des essuies glace, rideaux d'eau projetées par les véhicules de devant et une perte d'énergie de la lumière des phares et donc une vision de la chaussée réduite. Tous ces facteurs rendent la tâche de conduite difficile et le conducteur devient moins concentré sur la vision périphérique. De plus, les effets peuvent s'aggraver la nuit. Dans cette thèse nous nous intéressons aux gouttes présentes sur le pare-brise qui peuvent gêner la vision du conducteur et altérer les performances des ADAS en masquant des objets de la scène.

Notons que les conditions dans lesquelles un système de vision rencontre des difficultés sont, la plupart du temps, les mêmes dans lesquelles un humain a du mal à bien percevoir son environnement : géométrie de la route, obstacles, météo dégradées, illuminations. . .

ADAS/Caméra

D'une manière générale, les signaux renvoyés par la caméra peuvent être perturbés soit à cause du capteur (optique, électronique. . .) soit à cause de l'environnement (obscurité, météo, mouvements rapide. . .).

Le problème avec les ADAS utilisant une caméra est qu'ils sont le plus souvent conçus pour fonctionner dans le cas d'une météo favorable et leurs performances peuvent chuter de manière drastique dans le brouillard ou la pluie. En effet, comme la plupart des méthodes de traitement d'images utilisent des descripteurs qui dépendent de la couleur et du contraste de l'image et que ces derniers sont atténués par le brouillard, les résultats obtenus après le traitement sont rarement satisfaisants. De la même façon, en présence des gouttes un algorithme d'appariement, par exemple, ne va pas fournir de bons résultats à cause des occultations.

Dans des situations telles que le brouillard ou la pluie, la visibilité du conducteur est très réduite. Alors que c'est le moment où il a le plus besoin d'être assisté, les ADAS voient aussi leurs performances se dégrader et seront donc moins efficaces. Pire encore, ils risquent de fournir de mauvaises informations au conducteur qui prendra de mauvaises décisions et pourra se trouver en situation de danger.

Une solution simple pour éviter ce genre de problèmes est de désactiver le sys-

tème dans de telles situations. Cela pourra se faire automatiquement en utilisant des capteurs de brouillard ou de pluie. Notons, que les chercheurs participant au projet VIAC, cité plus haut, ont signalé que les rares fois où les systèmes de vision ont rencontré des problèmes étaient en cas de météo dégradée. Ils étaient donc contraints d'arrêter le système et de repasser en conduite manuelle.

Une autre solution est de modifier les paramètres des algorithmes afin qu'ils s'adaptent à de telles situations. Sauf qu'il faut le faire pour chaque système, ce qui accroît leur coût de développement et leur complexité.

La solution que nous proposons dans cette thèse est de détecter ces perturbations et de restaurer le signal à la sortie du capteur avant de le fournir aux ADAS.

Restauration brouillard et pluie

Détection

La première étape pour surmonter notre problème est d'avoir un système capable d'identifier et de quantifier les conditions météorologiques. Il est alors possible, par exemple, de conseiller au conducteur une vitesse en fonction du niveau de visibilité, d'activer automatiquement les feux de brouillard ou de désactiver l'ADAS utilisant une caméra dans le cas d'une visibilité très réduite qui pourra gêner son bon fonctionnement. La détection, dans notre cas, va permettre d'activer l'étape de la restauration en cas de brouillard ou de pluie. Dans ce qui suit nous nous appuyons sur des systèmes de détection existants.

Restauration

Suivant la densité du brouillard, la luminance et la distance de l'objet observé, le degré d'atténuation des couleurs et des contrastes varie. Tant que l'atténuation n'atteint pas un certain niveau, il est toujours possible de restituer les informations atténuées dans l'image par la diffusion atmosphérique. De la même façon, les gouttes

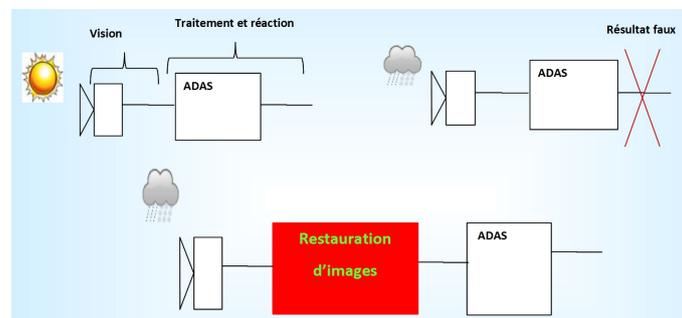


FIGURE 1.3 – En haut à gauche, fonctionnement nominal d'un ADAS par temps clair. À droite, fonctionnement dégradé à cause d'une visibilité réduite. En bas, système proposé dans cette thèse, fondé sur un module de restauration.

de pluie pourront être restaurées à partir des informations disponibles dans l'image tant que les gouttes ne couvrent pas une surface trop significative de l'image.

Nous allons proposer, dans cette thèse, des solutions qui, après la détection des conditions météorologiques défavorables, en particulier le brouillard et la pluie, restaurent les images à la sortie du capteur avant de les fournir aux ADAS leur permettant ainsi d'étendre leur gamme de fonctionnement. La figure 1.3 illustre le principe d'association d'un modèle de restauration au ADAS.

Plan

Nous allons montrer dans le chapitre 2, intitulé état de l'art, différentes approches existantes qui permettent d'éliminer le bruit dans les images ou d'améliorer le contraste et la couleur. Nous aborderons ensuite les deux problèmes de restauration d'images dégradées par le brouillard ou la pluie. Pour chacun de ces deux phénomènes, nous expliquerons les modèles optiques correspondants et discuterons des différentes approches existantes permettant de restaurer les images de brouillard ainsi que les méthodes permettant la détection des gouttes de pluie afin d'effectuer la restauration.

Dans le chapitre 3 Nous allons présenter les différentes bases d'images de brouillard et de pluie créées pour évaluer les méthodes de restauration : images de synthèses, images réelles avec brouillard de synthèse et images de brouillard réelles. Les différentes méthodes utilisées seront présentées : utilisation de la vérité terrain en se comparant à l'image restaurée idéale et détection des contours visible qui nous donnent des indicateurs sur le gain en terme de visibilité.

La méthode proposée pour la restauration d'images de brouillard est présentée dans le chapitre 4. Les différentes étapes de l'algorithme seront présentées : caractérisation du brouillard, estimation de l'intensité du ciel, restauration du contraste et de la couleur, lissage et fusion avec une approche permettant de mieux restaurer le plan de la route. Nous verrons ensuite l'algorithme d'optimisation, le recuit simulé, utilisé pour trouver la meilleure configuration des paramètres de notre algorithme. Nous allons ensuite montrer les résultats d'évaluation qualitative et quantitative. Enfin, nous allons présenter les améliorations proposées pour obtenir des images restaurées avec un meilleur contraste global et une image plus lisse.

Dans le chapitre 5, nous allons aborder le cas d'images dégradées par les gouttes de pluie. Nous commencerons par présenter les différents types d'approche "inpainting" qui pourront être utilisées pour résoudre le problème : diffusion, synthèse de texture et approches temporelle. Parmi ces méthodes un type d'approche à retenu notre attention, les EDP de diffusion. Nous allons présenter deux méthodes des plus évoluées dans cette catégorie. La méthode qui donne les meilleurs résultats à été retenue et ses paramètres optimisés sur une base d'images routières en utilisant le recuit simulé.

Dans le chapitre 6, nous allons présenter trois applications ADAS, appartenant à 3 familles différentes, pour notre algorithme de restauration d'images de brouillard.

La première application est la détection des panneaux en utilisant comme pré-traitement la restauration. Nous présentons dans cette partie la méthode de détection, la base de d'images de panneaux permettant l'évaluation et les résultats en terme de performances du système proposé. La deuxième application concerne la mise en place d'une expérience visant à mesurer le temps de perception d'un objet dans le brouillard par un panel de sujets, afin d'évaluer les performances d'un systèmes d'assistance HUD dans le brouillard. Nous allons montrer comment l'expérience a été mise en place et les résultats obtenues en terme de gain de temps de détection. La dernière application ADAS est la détection de l'espace navigable. Nous allons présenté la méthode de détection et les résultats obtenus après optimisation de la méthode sur une base d'images réelles de routes.

Enfin, nous allons conclure cette thèse et présenterons les différentes perspectives auxquelles nous avons pensé en terme d'amélioration des méthodes et d'autres applications ADAS ou de vision en général.

Etat de l'art - Restauration des images -

Sommaire

2.1 Cas général	9
2.1.1 Chaîne d'acquisition	9
2.1.2 Dégradation des images	11
2.1.3 Débruitage	12
2.1.4 Rehaussement du contraste	22
2.1.5 Balance des blancs	25
2.2 Images dégradées par le brouillard	27
2.2.1 Modèle optique du brouillard	28
2.2.2 Méthodes Interactives	33
2.2.3 Méthodes automatiques utilisant une seule image	42
2.3 Images dégradées par les gouttes de pluie	53
2.3.1 Physique des gouttes de la pluie	54
2.3.2 Détection des gouttes de pluie	56
2.4 Conclusion	60

2.1 Cas général

Une image dégradée peut avoir un impact négatif sur son interprétation par l'œil humain ainsi que sur les performances d'un système de vision par ordinateur. Dans ce chapitre, nous allons montrer les phénomènes entrant en jeu dans les différentes étapes de la chaîne d'acquisition d'une scène et qui peuvent altérer la qualité de l'image capturée. Nous décrirons par la suite les méthodes utilisées pour améliorer la qualité de l'image.

2.1.1 Chaîne d'acquisition

L'image récupérée à la sortie d'un capteur image correspond à la projection d'une scène tridimensionnelle (scène réelle) sur un plan bidimensionnel (plan image). Le sténopé (pinhole) est le modèle idéal de formation d'images : absence d'aberrations géométriques car il n'utilise pas de lentille et possède une netteté de 0 à l'infini. Ce modèle, comme le montre la figure 2.1, permet de récupérer une image après

le passage de la lumière, émise ou renvoyée par un objet, par un orifice de très petite taille. La taille du trou doit être d'au moins 1/100 ème de la taille de l'objet photographié pour que chaque point de la scène ne corresponde qu'à un seul point dans l'image produite.

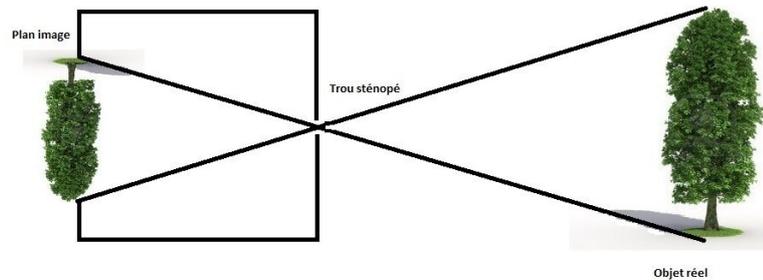


FIGURE 2.1 – Modèle sténopé

Dans les caméras classiques le trou du sténopé est remplacé par une lentille qui, grâce à une plus grande surface et à sa propriété de focalisation de la lumière, permet de récupérer une image plus lumineuse et d'avoir une meilleure netteté. Les éléments qui composent une caméra basique sont les suivants : une ou plusieurs lentilles, le plan image (film, CCD, CMOS), un dispositif de focalisation qui permet de déplacer la lentille par rapport au plan image, le diaphragme qui contrôle la quantité de lumière reçue par la lentille et l'obturateur qui contrôle le temps d'ouverture du diaphragme.

La qualité de l'image observée va dépendre de l'optique du capteur, de l'électronique (amplification, quantification et échantillonnage) ainsi que de l'environnement dans lequel la lumière va se propager avant d'atteindre l'objectif et des conditions d'éclairage (Figure 2.2). Nous allons montrer, dans la suite, les différents types de dégradations d'images.

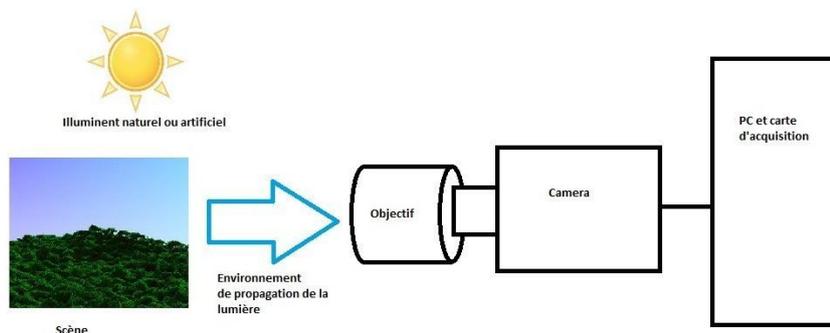


FIGURE 2.2 – Chaîne d'acquisition d'une image

2.1.2 Dégradation des images

L'affectation des images par du bruit est un problème inévitable. Afin de pouvoir analyser et effectuer des traitements sur les images acquises de la manière la plus fiable possible, en plus d'un réglage minutieux des paramètres du capteur (focale et ouverture), les défauts dus aux différents éléments de la chaîne d'acquisition de l'image (voir figure 2.2) doivent être corrigés le mieux possible.

Nous citons ci-dessous les différents défauts à prendre en compte.

Problèmes liés à l'instrumentation :

- Aberrations sphérique : flou
- Aberrations chromatiques : franges colorées
- Astigmatisme : flou
- Distorsions : à cause des imperfections de la lentille l'image est très déformée (plus aux bords qu'au centre).
- Vignettage : assombrissement des bords
- Bruit photonique
- Bruit thermique
- Bruit de lecture
- Bruits d'amplification et de quantification

Problèmes liés à l'environnement :

- Éclairage de la scène faible ou in-homogène
- Géométrie de la scène : occultations
- Flou de bougé
- Présence de particules atmosphérique : poussière, brouillard, pollution...

Problèmes liés aux traitements des images :

- Génération d'artefacts

Parmi les phénomènes cités ci-dessus, nous distinguons ceux qui sont aléatoires et ceux qui sont déterministes. Par exemple, le bruit photonique est un phénomène aléatoire (poissonien) à cause du caractère aléatoire de la lumière. La distorsion est un phénomène déterministe qui peut être corrigé en effectuant un calibrage de la caméra. Le flou dû à un mouvement plan de la caméra peut être corrigé en connaissant les déplacements horizontaux et verticaux de la caméra. Le flou dû à l'optique est aussi un phénomène déterministe. Une mesure de la PSF (réponse impulsionnelle du système optique) et une déconvolution de l'image observée par la PSF permettent d'éliminer ce type de dégradation.

Notons que dans cette liste, les problèmes qui nous intéressent dans cette thèse sont essentiellement le brouillard et la pluie. Cependant, dans la plupart des méthodes de restauration, nous avons besoin d'éliminer le bruit et les artefacts générés ou amplifiés par la méthode utilisée. Pour cela, nous avons été amenés à étudier différentes méthodes de débruitage, d'amélioration de la couleur ou des contrastes. Nous avons ensuite sélectionné celles qui pourront nous intéresser et nous les avons adapté à notre problématique ADAS.

Restauration et débruitage

Pour montrer la différence entre la restauration et le débruitage prenons comme exemple le cas d'une image I_0 affectée par un bruit b additif en plus d'un flou optique H .

Nous pouvons distinguer le débruitage d'images qui consiste à retrouver l'image I_0 sans bruit à partir du modèle :

$$I = I_0 + b \quad (2.1)$$

et la restauration d'images qui consiste à récupérer I_0 à partir du modèle de formation du flou tout en prenant en compte le bruit :

$$I = H * I_0 + b \quad (2.2)$$

La restauration permet donc de retrouver l'image source en essayant de prendre en compte les différents phénomènes qui entrent en jeu dans le processus de formation d'images et qui peuvent altérer la qualité de celle-ci. Pour cela une connaissance précise de ces modèles est donc nécessaire pour restaurer l'image : modèle de Koschmieder dans le cas du brouillard par exemple (Voir section 2.2) ou modèle de formation de la pluie (Voir section 2.3).

Dans certains cas, le problème peut être mal posé (au sens d'Hadamard) : le problème n'admet pas de solution, en admet plusieurs ou il n'existe pas de solution stable. Dans ce cas, il devient nécessaire d'introduire des contraintes (a priori) permettant de régulariser la solution. Pour cela, des méthodes statistiques de minimisation d'énergie sont souvent utilisées : Les champs de Markov (MRF), par exemple, appliqués dans un contexte d'estimation bayésienne en combinaison avec un algorithme d'optimisation sont très connus pour résoudre ce genre de problèmes mais sont très coûteux en temps de calcul. Cependant, suivant le type de problème, nous trouvons des méthodes déterministes (trouvent un minimum local qui est proche du minimum global) qui permettent d'obtenir des résultats équivalents plus rapidement.

Dans ce qui suit, nous allons montrer quelques méthodes existantes permettant d'améliorer la qualité d'images bruitées, puis nous nous intéresserons aux méthodes de restauration d'images dégradées par les conditions météorologiques.

2.1.3 Débruitage

Le bruit peut apparaître sous différentes formes : aléatoire, centré, additif, multiplicatif ou convolutif. La figure 2.3 montre une image à laquelle nous avons ajouté différents types de bruit : gaussien additif, multiplicatif et impulsionnel.

Dans cette section, nous allons faire un panorama de méthodes destinées au débruitage des images.



FIGURE 2.3 – En haut à gauche, image originale ; en haut à droite, bruit gaussien additif de moyenne nulle et variance 0,025 ; en bas à gauche, bruit multiplicatif de moyenne nulle et variance 0,05 ; en bas à droite, bruit poivre et sel couvrant 20 % de l'image

2.1.3.1 Filtres classiques

En général, les filtres d'images peuvent être soit linéaires (moyenne, convolution) ou non linéaires (médiane).

Lorsque le bruit est un signal à hautes fréquences, un filtrage passe bas permet de le réduire mais ajoute du flou et atténue les contours. En particulier, ils atténuent le bruit dans les zones uniformes, en revanche, au niveau des zones texturées ou proche des contours, les effets sont visibles. Plus la taille du filtre est grande plus la dégradation de l'image est importante. La figure 2.4 montre le résultat obtenu par les filtres moyenne et médian pour différents types de bruits.

Nous pouvons utiliser les statistiques locales de l'image afin d'améliorer ce genre de filtres comme nous allons le voir dans la suite.

2.1.3.2 Statistiques locales de l'image

Afin de réduire le flou dans les zones texturées, nous pouvons utiliser la variance de l'intensité de chaque pixel par rapport à son voisinage (comme dans [Loizou 2005]) afin de détecter si ce pixel est bruité : nous comparons la différence entre la valeur d'un pixel et la moyenne (ou la médiane) de ses voisins et si cette différence est plus grande que l'écart type des voisins alors nous remplaçons la valeur du pixel par la moyenne (ou la médiane) de ses voisins, sinon nous gardons sa valeur initiale.

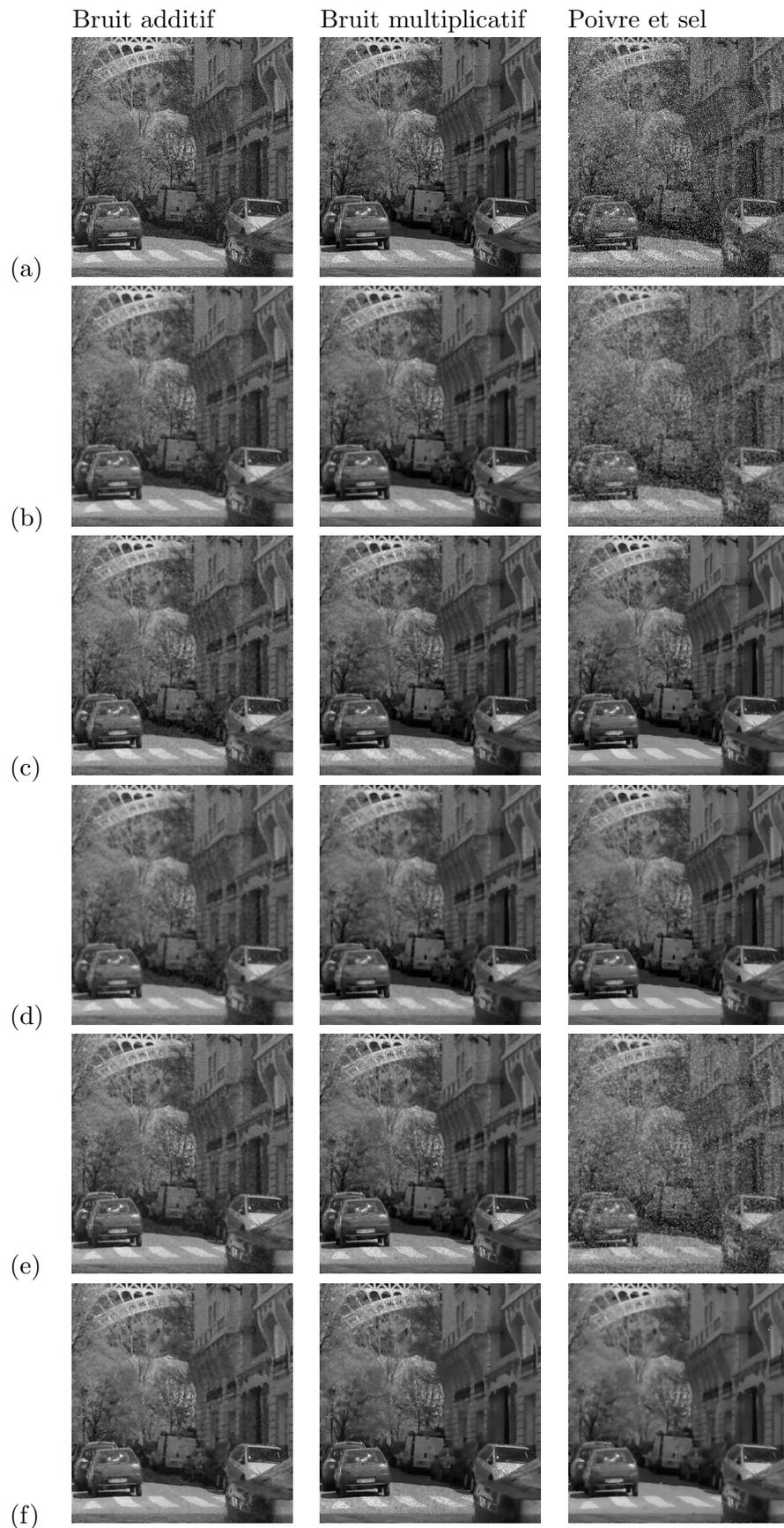


FIGURE 2.4 – Débruitage : Images avec du bruit (a), Moyenne (b), Médian (c), Statistiques locales(moyenne)(d), Wiener(e), Filtre bilatéral (f).

Comme le montre la figure 2.4, ce filtre donne des meilleurs résultats qu'une simple moyenne surtout dans le cas d'un bruit poivre et sel.

Toutefois, pour des grandes quantités de bruit cet algorithme atteint ses limites. Dans la suite, nous allons montrer des méthodes de débruitage qui sont plus robustes.

2.1.3.3 Filtre de Wiener

Considérons le cas d'un bruit additif b de variance σ , le filtre de Wiener [Ghael 1997] correspond à l'estimateur linéaire optimal G de I_0 tel que $\hat{I}_0 = GI$ et qui minimise l'espérance de l'erreur quadratique moyenne $\varepsilon = E \| I_0 - \hat{I}_0 \|^2$. Nous obtenons G en cherchant sa valeur tel que $\frac{\partial \varepsilon}{\partial G} = 0$, ce qui nous donne :

$$G_w = R_{I_0}(R_{I_0} + \sigma^2 I_d)^{-1} \quad (2.3)$$

avec $R_{I_0} = E[I_0 I_0^\perp]$ et I_d la matrice identité.

La figure 2.4 montre le résultat obtenu par ce filtre. Dans le cas de bruit additif ou multiplicatif, nous obtenons une image contenant moins de flou par rapport aux filtres précédents.

En plus du débruitage, ce filtre est aussi utilisé dans le traitement d'images pour le défloutage (restauration) d'images [Legendijk 1999].

Pour certains types de bruits, comme un bruit poivre et sel, ce filtre devient inefficace. Dans la suite, nous allons présenter un autre filtre, qui est plus robuste à différents types de bruit.

2.1.3.4 Filtre Bi-latéral

Il s'agit d'un filtre linéaire dont le principe est de remplacer la valeur d'un pixel par une moyenne des valeurs voisines pondérées par les distances spatiales (application d'une gaussienne) et les distances des niveaux de gris des voisins [Tomasi 1998, Fleishman 2003]. La valeur d'un pixel se trouvant à la position (i, j) et ayant une intensité $I(i, j)$ est remplacée par $\hat{I}(i, j)$ selon l'équation suivante :

$$\hat{I}(i, j) = \frac{1}{N} \sum_{m, n \in \Omega} w_g(m, n) w_{ng}(m, n) I(m, n) \quad (2.4)$$

avec Ω un voisinage autour de (i, j) , w_g les poids de la gaussienne et

$$w_{ng} = \frac{1}{|I(m, n) - I(i, j)|^2 + \lambda} \quad (2.5)$$

le poids des distance en niveaux de gris avec λ est un coefficient de régularisation permettant de contrôler la force du lissage. Notons que d'autres poids w_{ng} peuvent être utilisés (exponentielle, distance L_1 ...).

N est un coefficient de normalisation tel que :

$$N = \sum_{m, n \in \Omega} w_g(m, n) w_{ng}(m, n) \quad (2.6)$$

La figure 2.4 montre les résultats obtenus par ce filtre. Nous constatons que dans les différents éléments de l'image (arbre, bâtiments, voitures . . .) le bruit a été bien réduit tout en préservant les contours. Dans le cas d'un bruit poivre et sel, nous avons utilisé une distance L_1 pour les poids w_{ng} qui donne de meilleurs résultats pour ce type de bruit. Le problème avec les filtres locaux est le choix de la taille de la fenêtre : une grande taille élimine bien le bruit sur les régions homogènes mais dégrade les contours. Une fenêtre de petite taille rend le filtre moins efficace sur des bruits intenses mais altère moins les détails de l'image dans les régions inhomogènes.

Comme nous allons le voir au chapitre 4, le filtre bilatéral peut aussi être utilisé afin d'éliminer les effets blocs liées à un pré-traitement par exemple. Celui-ci s'est montré robuste à différentes variations d'intensité dans l'image et à plusieurs types de bruits et d'artefacts.

Le fait de prendre en compte les différences de niveaux de gris permet de contrôler la diffusion le long des contours à fort gradient. De la même façon que les EDP (diffusion), que nous allons détailler dans la section suivante, le filtre bilatéral permet d'éliminer le bruit tout en préservant les contours. L'auteur [Barash 2002] a montré qu'il existe une équivalence entre le filtre bilatéral et la diffusion.

2.1.3.5 Diffusion

Ce type de filtrage est inspiré de la physique (propagation des ondes et equation de la chaleur). Il s'agit de propager l'information contenue dans les pixels vers les zones voisines suivant l'équation de diffusion de la chaleur (Équations aux dérivées partielles : EDP).

La diffusion, en physique, a pour but d'équilibrer les différences de concentrations (niveaux de gris, dans le cas d'images) sans aucun changement de la masse. La loi de Fick décrit la propriété d'équilibre :

$$j = -D\nabla I \quad (2.7)$$

avec j le flux, ∇I le gradient de I (l'image bruitée dans notre cas) et D le tenseur de diffusion.

Lorsque $D = g$ est un scalaire, appelé la diffusivité, il s'agit de diffusion isotropique. Dans ce cas, j et ∇I sont parallèles. Sinon, il s'agit de diffusion anisotropique.

L'équation de continuité décrivant la conservation de la masse est la suivante :

$$\frac{\partial I}{\partial t} = -div j \quad (2.8)$$

avec t le temps.

À partir des équations 2.7 et 2.8, nous déduisons l'équation de diffusion :

$$\frac{\partial I}{\partial t} = div(D\nabla I) \quad (2.9)$$

Nous pouvons réécrire cette équation sous la forme :

$$\frac{\partial I}{\partial t} = D\Delta(I) \quad (2.10)$$

avec $\Delta I = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ l'opérateur laplacien.

Pour discrétiser cette équation, nous pouvons écrire la dérivé de I par rapport au temps à l'aide d'un développement de Taylor du premier ordre :

$$\frac{\partial I}{\partial t} = \frac{I_{t+1} - I_t}{\Delta t} \quad (2.11)$$

et l'opérateur laplacien à l'aide d'un développement de Taylor du deuxième ordre :

$$\frac{\partial^2 I}{\partial x^2} = \frac{I_{x+1} - 2I_x + I_{x-1}}{\Delta x^2} \quad (2.12)$$

Ainsi l'équation 2.10 devient sous une forme discrète :

$$I_{t+1} = I_t + D \frac{\Delta t}{\Delta x^2} (I_{x+1} - 2I_x + I_{x-1}) + D \frac{\Delta t}{\Delta y^2} (I_{y+1} - 2I_y + I_{y-1}) \quad (2.13)$$

Notons que des développement de Taylor d'ordre supérieur peuvent être utilisés. Suivant la forme du tenseur de diffusion D nous distinguons 3 types de diffusion :

Diffusion isotropique linéaire : il s'agit d'une diffusion isotropique globale telle que la diffusivité est constante. C'est le cas d'une convolution de l'image initiale par une gaussienne qui est une solution pour l'équation de diffusion. Ce type de diffusion est optimal dans les zones uniformes mais floute les contours et les textures et les rend difficilement identifiable. La figure 2.5 montre le résultat obtenu par ce type de diffusion pour les trois types de bruits présentés précédemment.



FIGURE 2.5 – Débruitage par diffusion linéaire : application d'une gaussienne

Diffusion isotropique non linéaire : Les auteurs Perona et Malik [Perona 1990] ont été les premiers à introduire ce modèle. L'idée est de réduire la diffusivité dans les régions ayant une grande probabilité d'être des contours. Cette probabilité peut être mesurée par $|\nabla I|^2$. L'équation de diffusion devient dans ce cas :

$$\partial_t I = \text{div}(g(|\nabla I|^2)\nabla I) \quad (2.14)$$

avec

$$g(|\nabla I|^2) = \frac{1}{1 + \frac{|\nabla I|^2}{\lambda^2}} \quad (2.15)$$

La figure 2.6 montre les résultats obtenus par cette méthode pour les 3 images bruitées présentées auparavant.



FIGURE 2.6 – Débruitage par diffusion non linéaire : modèle de Perona et Malik

L'utilisation d'une régularisation (spatiale, temporelle ou spatiale et temporelle) permet de stabiliser le processus. Dans [Catté 1992], les auteurs proposent une régularisation spatiale en remplaçant la diffusivité $g(|\nabla I|^2)$ par une version lissée par une gaussienne $g(|\nabla I_\sigma|^2)$ avec :

$$I_\sigma = K_\sigma * I \quad (2.16)$$

K_σ étant le noyau de la gaussienne. Ce qui nous donne :

$$\partial_t I = \text{div}(g(|\nabla I_\sigma|^2)\nabla I) \quad (2.17)$$

D'un point de vue pratique, la régularisation spatiale limite l'influence des bruits ayant des échelles inférieures à σ .

L'auteur [Whitaker 1993] a proposé de choisir σ comme fonction décroissante du temps.

Diffusion anisotropique non linéaire : Il s'agit du cas où le tenseur de diffusion est local. Dans certains cas, par exemple pour prendre en compte les discontinuités, il est plus intéressant de ne pas suivre strictement le gradient pour augmenter l'influence des régions qui nous intéressent (idée de Graham [Graham 1962]).

L'auteur [Weickert 1998] propose une régularisation du modèle de Perona et Malik en prenant en compte la direction du gradient en plus de son module. Cela permet la diffusion le long de contours plutôt qu'au travers de ceux-ci. Les vecteurs propres du tenseur nous donnent les directions parallèles et orthogonale au gradient :

$$v1 \parallel \nabla I_\sigma \quad (2.18)$$

et

$$v2 \perp \nabla I_\sigma \quad (2.19)$$

Wickert a proposé de choisir les valeurs propres du tenseur tel que :

$$\lambda_1(\nabla I_\sigma) = g(|\nabla I_\sigma|^2) \quad (2.20)$$

et

$$\lambda_2(\nabla I_\sigma) = 1 \quad (2.21)$$

Dans le cas d'images couleurs un tenseur commun aux 3 canaux permet d'éviter le problème de création de contours.

En plus du débruitage, les EDP ont plusieurs applications [Weickert 1998] dans le domaine du traitement d'image : shape from shading, modèle de contour actif, flot optique, stéréo-vision, modification de l'histogramme, segmentation de texture, analyse de séquence d'images, détection de contours, filtrage d'images médicales. . .

Une des applications des EDP de diffusion qui nous intéressent dans notre étude est l'"inpainting" et qui va nous servir afin de restaurer les images dégradées par les gouttes de pluie dans le chapitre 5.

Nous allons présenter maintenant une autre méthode de débruitage qui est la "Total Variation Model".

2.1.3.6 Total Variation Model

À partir des caractéristiques locales de l'image, nous mesurons une régularité globale. Cette régularité porte le nom de "variation totale" et se calcule comme la somme des gradients locaux de l'image $TV = \sum |\nabla I_0|$.

Dans le cas d'un bruit additif b l'image I observée s'écrit $I = I_0 + b$. Le débruitage de l'image s'effectue en cherchant l'image I_0 qui minimise la fonction :

$$\lambda TV + \sum (I - I_0)^2 \quad (2.22)$$

La TV est utilisée comme un terme de régularisation qui permet de pénaliser les grandes variations et d'autoriser les discontinuités le long des contours suffisamment réguliers. Le facteur λ contrôle la force de débruitage : plus il est grand plus la variation totale de l'image résultante sera petite. Le deuxième terme de cette équation est le terme d'attache aux données.

Le défaut de ce type de lissage est que les textures peuvent être considérées comme du bruit et être effacées. Nous observons aussi la création d'effets escalier.

Cette méthode, en plus du débruitage [Rudin 1992], peut aussi être utilisée pour restaurer les images dégradées par du flou [Chambolle 2010]. Par exemple, dans le cas d'une utilisation d'une méthode de défloutage basée sur la l'inversion de la fonction de transfert, nous allons générer du bruit, comme le montre la figure 2.7. La "Total Variation Model" est, dans ce cas, mieux appropriée.

Notons que la méthode "Total Variation Model" peut être interprétée comme un cas particulier des approches bayésiennes que nous allons aborder dans la suite.



FIGURE 2.7 – De gauche à droite : image affectée par du flou optique, restauration par inversion du noyau et restauration par TVM

2.1.3.7 Approches bayésiennes

Considérant une image I_0 à laquelle vient s'ajouter un bruit aléatoire b , l'image observée I s'écrit alors : $I = I_0 + b$. La formule de Bayes exprime la distribution a posteriori d'avoir I_0 connaissant I :

$$P(I_0/I) = \frac{P(I/I_0)P(I_0)}{P(I)} \quad (2.23)$$

Le débruitage consiste à maximiser cette probabilité (voir [Hunt 1977]). La probabilité $P(I)$ est connue et elle joue le rôle d'une constante de normalisation, $P(I/I_0)$ est la vraisemblance et est déterminée à partir du modèle de formation des données et $P(I_0)$ est le terme de régularisation (a priori).

L'estimation ML (maximum de vraisemblance) consiste à chercher la valeur de I_0 qui maximise la vraisemblance :

$$\operatorname{argmax}_{I_0} \left(\prod_i P(I_i/I_0) \right) \quad (2.24)$$

Pour des raisons de simplicité il est préférable d'estimer le log de ce produit de probabilités :

$$L = \sum \log P(I_i/I_0) \quad (2.25)$$

Généralement la maximisation se fait en cherchant I_0 qui satisfait $\partial L / \partial I_0 = 0$.

La densité de probabilité $P(I/I_0)$ dans le cas d'un bruit gaussien b , tel que $b = I - I_0$, de moyenne nulle et d'écart type σ est :

$$e^{-\frac{1}{2\sigma^2} \sum_{i,j} |I_{i,j} - I_{0i,j}|^2} \quad (2.26)$$

L'estimation MAP (Maximum a posteriori) a l'avantage de pouvoir prendre en compte l'a priori $P(I)$. La MAP consiste à trouver le I_0 qui maximise $P(I_0/I)$, ce qui donne en appliquant la fonction logarithmique à la formule de Bayes :

$$\operatorname{argmax}_{I_0} \left(\sum \log P(I_i/I_0) + \log P(I_0) \right) \quad (2.27)$$

Les champs de Markov sont souvent utilisés pour déterminer la probabilité a priori $P(I_0)$. Pour cela l'image est considérée comme un arrangement d'atomes [Geman 1984, Charbonnier 1994] se trouvant sur plusieurs états énergétiques : les niveaux de gris. L'état de chaque atome dépend seulement de ses voisins. Le théorème d'Hammersley-Clifford donne l'expression de cette probabilité :

$$P(I_0) = \frac{1}{Z} * \exp(-J(I_0)) \quad (2.28)$$

avec Z une constante de normalisation, $J(I_0)$ est la somme de fonctions potentielles calculée sur des cliques (voisinages). L'a priori correspond au choix des cliques et des fonctions potentielles (opérateur différentiel par exemple).

Le problème d'estimation MAP peut s'écrire sous une forme équivalente qui consiste à résoudre le problème de minimisation suivant :

$$\operatorname{argmin}_{I_0} \lambda J(I_0) + \frac{1}{2\sigma^2} \sum_{i,j} |I_{i,j} - I_{0i,j}|^2 \quad (2.29)$$

avec λ une fonction permettant de pondérer les deux termes.

La fonction $J(I_0)$, correspondant à l'a priori, synthétise l'idée que nous avons du signal que nous voulons récupérer. Notons que la méthode "variation totale" est une estimation MAP tel que $J(I_0) = TV(I_0)$.

Pour déterminer le MAP, il existe deux familles de méthodes :

- Les algorithmes déterministes qui sont rapides, mais risquent de converger vers un minimum local éloigné du minimum global. Nous citons par exemple l'algorithme de descente du gradient et ses variantes.
- Les algorithmes stochastiques qui sont lents mais assurent la convergence vers un minimum global. Nous citons dans cette catégorie l'algorithme du recuit qui accepte une nouvelle configuration si l'énergie diminue ou avec une probabilité $\frac{1}{Z} \exp(-\beta J(I_0))$ pour sortir des minima locaux, avec Z une constante de normalisation et $\beta = \frac{1}{KT}$, K étant la constante de Boltzmann et T la température.

Dans la suite nous allons présenter une autre méthode qui a connu un grand succès dans le domaine de débruitage d'images : la transformée en ondelette discrète.

2.1.3.8 Transformée en ondelettes

Les ondelettes ont été introduites au début des années 80, afin de surmonter un problème lié à la transformée de Fourier qui ne permet pas de localiser les fréquences du signal dans le temps.

Considérons une image I contenant un bruit additif b : $I = I_0 + b$, la transformée en ondelette discrète ou DWT (Discret Wavelet Transform) d'un signal I s'écrit sous la forme :

$$d = WI \quad (2.30)$$

avec d le dictionnaire et W la transformée d'ondelettes (voir [Boubchir 2005]).

En appliquant cette transformée W au modèle de formation du bruit nous obtenons les coefficients de la DWT suivants :

$$c_{mn} = a_{mn} + b_{mn} \quad (2.31)$$

$$d_{mn}^{jo} = s_{mn}^{jo} + b_{mn} \quad (2.32)$$

avec m, n la position spatiale du pixel,

a_{mn} : coefficient d'approximation de la DWT de I_0 ,

c_{mn} : coefficient d'approximation de la DWT de I ,

s_{mn}^{jo} : coefficient de détails de la DWT de I_0 , à l'échelle j et l'orientation o ,

d_{mn}^{jo} : coefficient de détails de la DWT de I , à l'échelle j et l'orientation, o .

Le débruitage consiste à garder les coefficients c_{mn} et d_{mn}^{jo} ayant une valeur importante, en considérant que les faibles valeurs correspondent au bruit, ensuite d'inverser l'équation 2.30 pour récupérer l'image sans bruit : $I = W^{-1}d$.

Pour récupérer les coefficients qui nous intéressent, il faudra trouver un seuil pour détecter les coefficients qui correspondent au bruit, pour cela une grande diversité de méthodes existe (seuillage fort, doux...).

Notons qu'il existe d'autres méthodes appartenant à la même famille que les ondelettes (représentation temps-fréquence) telles que les curvelet ou les contourlet.

Outre les problèmes liées au bruit, l'image peut souffrir de perte des contrastes. Dans ce qui suit, nous allons montrer quelques méthodes utilisées pour améliorer le contraste des images.

2.1.4 Rehaussement du contraste

Un éclairage inhomogène ou très faible peut induire des pertes de contraste dans l'image. Dans le cas où l'information du contraste est toujours disponible, il existe plusieurs méthodes permettant son rehaussement de façon globale ou locale. Le rehaussement du contraste est très utile pour de nombreuses applications de vision par ordinateur telles que la segmentation d'image ou la reconnaissance de formes.

2.1.4.1 Égalisation d'histogramme

Le but de ce type d'approches est de modifier l'histogramme de l'image en assignant des nouvelles valeurs aux pixels de l'image d'entrée. L'histogramme des images ayant un faible contraste occupe une petite portion de la plage des intensités. Le but de l'égalisation est d'étaler l'histogramme sur une plus grande plage [Hummel 1977, Acharya 2005].

Pour cela, à partir de l'histogramme de l'image, nous calculons l'histogramme cumulatif et l'appliquons (après normalisation) à l'image afin d'étaler son histogramme uniformément sur toute la plage de dynamique.

La figure 2.8 illustre le principe d'égalisation d'histogramme. La figure 2.9 montre le résultat obtenu par cette méthode.

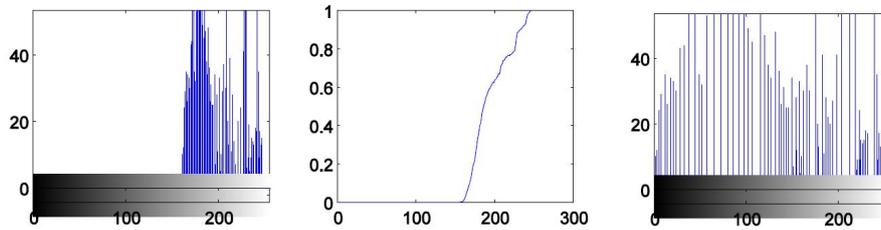


FIGURE 2.8 – Histogramme original, histogramme cumulatif normalisé entre 0 et 1, histogramme de l'image résultat.

Nous pouvons aussi utiliser d'autres fonctions (logarithmique, exponentielle, puissance ou autres) afin d'obtenir un histogramme ayant une certaine forme.

L'égalisation d'histogramme donne souvent des meilleurs résultats lorsqu'elle est appliquée localement.

2.1.4.2 Égalisation d'histogramme locale

Dans de nombreux cas, l'histogramme de l'image couvre une large dynamique. Dans ce cas une égalisation d'histogramme locale [Pizer 1987, Kim 2001] est nécessaire pour faire ressortir les contrastes des différentes parties de l'image.

Pour cela l'image est parcourue avec une fenêtre de petite taille et le principe d'égalisation décrit ci-dessus est appliqué sur chaque fenêtre séparément. Ensuite, afin d'éliminer les effets blocs générés, dus à la différence des histogrammes entre blocs voisins, une interpolation bi-linéaire est utilisée.

Cette méthode porte le nom de CLAHE (Contrast Limited Adaptive Histogram Equalization) [Zuiderveld 1994]. La figure 2.9 montre les résultats obtenus pour différentes tailles de fenêtres.

Le défaut de ce type de méthode est la sur-amélioration des contrastes : elle fait ressortir de faux détails. À cause du caractère local de la méthode, celle-ci requiert plus de temps de traitement qu'une égalisation globale.

Dans le chapitre 4, nous allons présenter une nouvelle méthode de restauration d'images altérées par le brouillard, inspirée de CLAHE et que nous avons adapté en y introduisant des contraintes déduites du modèle optique du brouillard.

Dans la suite, nous abordons une autre méthode classique de restauration locale du contraste.

2.1.4.3 Retinex

La Retinex [Jobson 1995] est la combinaison des mots Rétine et cortex. La méthode se fonde sur le constat que le système visuel humain perçoit le contraste et la couleur d'un objet relativement de la même façon dans des conditions d'illumination différentes. Ce n'est pas le cas pour les capteurs caméra car la valeur d'intensité d'un pixel dépend fortement du flux de photon. L'objectif est de construire, à partir d'une image donnée, une nouvelle image éclairée par une lumière blanche et constante.



FIGURE 2.9 – Première ligne : image originale et résultat après l'égalisation de l'histogramme. Deuxième ligne : résultats obtenues avec CLAHE pour différentes tailles de fenêtre

La Retinex (ou SSR : single scale retinex) à une échelle revient à appliquer une opération non linéaire sur l'image d'entrée à l'échelle logarithmique :

$$R = \log(I) - \log(F * I) \quad (2.33)$$

avec I : l'image d'entrée, F : un noyau gaussien tel que

$$F = k * \exp(-(x^2 + y^2)/\sigma) \quad (2.34)$$

avec k une constante de normalisation choisie telle que l'aire de la gaussienne soit de 1 :

$$\int \int F(x, y) dx dy = 1 \quad (2.35)$$

σ représente l'écart type qui permet de contrôler la quantité des détails retenus : une petite valeur permet une meilleure restitution des détails mais introduit des artefacts.

D'autres noyaux peuvent être utilisés :

- Land : $F = 1/r^2$ ou $F = 1/(1 + r^2/c_1^2)$ avec $r = \sqrt{x^2 + y^2}$ et $c_1 = 50 \text{ pix}$
- Moore : $F = \exp(-|r|/c_2)$ et $c_2 = 72 \text{ pix}$
- Hulbert : $F = \exp(-r^2/c_3^2)$ et $c_3 = 60 \text{ pix}$

Dans notre cas, nous avons utilisé un noyau gaussien. La figure 2.11 montre le résultat obtenu pour différentes tailles du noyau.

Nous remarquons que pour une petite taille de σ nous récupérons les détails fins, en revanche l'image souffre de pertes de couleurs. Avec une plus grande taille nous améliorons les couleurs mais observons des saturations et des pertes de détails. Nous utiliserons dans la suite la moyenne des résultats obtenus aux 3 échelles pour une meilleure restauration des détails et des couleurs.

2.1.4.4 Multi Scale Retinex

La MSR (Multi Scale Retinex) [Jobson 1997] est, comme son nom l'indique, une combinaison de plusieurs SSR (généralement 3) effectuées à différentes échelles (différentes tailles de gaussiennes) :

$$\sum_{n \in [1, N]} W_n R_{SSR_n} \quad (2.36)$$

avec N le nombre d'échelles.

Expérimentalement, il a été démontré qu'une pondération uniforme $W_n = 1/N$ donne de bons résultats.

La dernière étape de l'algorithme est la normalisation : ramener le résultat dans l'intervalle de définition de l'image à l'aide d'une opération affine : $R' = G \times R - O$ avec G le gain et O le décalage.

La figure 2.11 montre les résultats obtenus avec des tailles de voisinage de 15, 80 et 250 pixels.

L'algorithme du SSR est simple et automatique mais requiert un rapport signal sur bruit important pour obtenir un résultat satisfaisant.

D'un autre coté, afin d'améliorer le temps de traitement et pouvoir traiter des images de grande taille plus rapidement, il est usuel de remplacer la convolution dans le domaine spatiale par une multiplication dans le domaine fréquentiel.

Dans la suite, nous abordons un autre problème qui est la balance des blancs.

2.1.5 Balance des blancs

Dans certain cas, les zones blanches dans la scène apparaissent colorées dans les images. Généralement invisible par l'œil humain, qui est capable de faire la correction automatiquement, cela entraîne des couleurs extrêmement saturées après l'application d'une méthode de restauration. Ce problème est particulièrement sensible dans les conditions de brouillard, si le ciel n'est pas blanc lors de la restauration. Dans cette partie, nous allons décrire les méthodes que nous avons trouvées intéressantes pour améliorer les images avant l'application de méthodes de restauration.



FIGURE 2.10 – De gauche à droite et de haut en bas : image originale, Retinex avec un noyau $\sigma = 15$, Retinex avec un noyau $\sigma = 80$, Retinex avec un noyau $\sigma = 250$, Retinex multi-échelles [15 80 250] et résultat final après normalisation

2.1.5.1 Gray World

Le “Gray World” proposé par [Buchsbaum 1980] est l’une des plus anciennes méthodes utilisées pour effectuer une balance des blancs. Elle considère que les intensités moyennes des trois canaux RGB doivent être égales. Le principe est de garder le canal vert inchangé et de multiplier les canaux rouge et bleu, respectivement par les gains $R_g = Moyenne(G)/Moyenne(R)$ et $B_g = Moyenne(G)/Moyenne(B)$. L’auteur [Lam 2005] a proposé une combinaison de cette méthode avec la Retinex.



FIGURE 2.11 – Image originale à gauche et résultat de la balance des blancs à droite.

2.1.5.2 Max White

Cette approche suppose que le maximum d'intensité sur chaque canal correspond au blanc. Le gain de chaque canal dans le cas d'image 8 bit se calcule d'une manière simple : $G_C = 255/\max(C)$ avec $C = \{R, G, B\}$.

Dans le cas d'images de brouillard, comme nous allons le voir dans cette thèse, une balance des blancs peut être utilisée pour obtenir des meilleurs résultats de restauration.

Il existe d'autres méthodes permettant d'améliorer les couleurs de l'image telles que l'"*histogram matching*" [Zhang 1990] qui utilise deux images dont l'une, qui n'est pas dégradée, sert de référence pour améliorer les couleurs de l'autre. Des méthodes utilisant une combinaison des méthodes présentées dans cette partie sont aussi utilisées.

2.2 Images dégradées par le brouillard

Le brouillard est un amas de particules d'eau suspendues dans l'atmosphère qui se forment par condensation de la vapeur d'eau se trouvant dans l'air. Pour cela 2 conditions doivent être réunies : un air humide et une basse température au niveau du sol.

Lorsque la concentration de l'air en vapeur d'eau dépasse un certain seuil qui dépend de la température (plus l'air est chaud, plus le seuil est élevé), les molécules d'eau se regroupent pour former des gouttelettes d'eau liquide. On parle de saturation de l'air en vapeur d'eau. La chute de température au niveau du sol provoque la formation du brouillard.

Le principe de formation du brouillard est toujours le même, mais les étapes du processus de sa formation varient en fonction de l'environnement et des conditions météorologiques (brouillard d'advection, de rayonnement, d'évaporation...).

Nous distinguons aussi, suivant le type de particules, le brouillard naturel, le brouillard de pollution (particules de sulfates par exemple) et le brouillard de poussières.

En fonction de la distance de visibilité, qui correspond à la distance à laquelle un objet noir observé sur fond du ciel est visible avec un contraste de 5 % (CIE), on parle soit de brouillard lorsque la visibilité est inférieure à 1000m soit de brume dans le cas contraire. Le brouillard et la brume se forment de la même façon et il suffit que l'humidité augmente pour voir la brume se transformer en brouillard.

La chute de visibilité dépend de plusieurs facteurs : la concentration, la taille et la composition chimique des particules, l'humidité relative, l'angle de pénétration de la lumière...

Mesure de la visibilité

L'appareil permettant de mesurer la visibilité s'appelle un visibilimètre (diffusomètre, transmissomètre...). En général la mesure de visibilité s'effectue d'abord par observation : la plus grande distance à laquelle un objet noir sur fond diffusant

est visible pour l'observateur. Elle s'évalue en mesurant la POM (porté optique météorologique) qui correspond à la longueur du trajet parcourue par un rayon lumineux (parallèle, émanant d'une lampe à incandescence, à une température de couleur de 2 700 K) pour que son intensité soit réduite de 95 % (CIE).

Nous pouvons citer deux appareils utilisés dans les aéroports et les stations météo et qui sont :

- Le transmissiomètre qui mesure la transmittance de l'atmosphère, à partir d'un émetteur et d'un récepteur distants d'une dizaine de mètres. Il n'est pas envisageable dans le cas des ADAS à cause de son caractère encombrant.
- Le diffusiomètre qui mesure le coefficient d'extinction atmosphérique. Il a l'avantage d'être mobile (émetteur et récepteur collocalisés) mais ne prend pas en compte plusieurs phénomènes : opacité de l'atmosphère, lumière ambiante, phare de voiture, propriétés transmissives du pare-brise...

L'auteur [Pomerleau 1997] a été un des premiers à proposer d'utiliser une caméra embarquée dans un véhicule afin d'estimer une distance de visibilité relative. Le système détecte les marquages sur la route, et utilise l'hypothèse qu'ils possèdent un contraste faible en haut de l'image et un contraste plus fort en bas de l'image. En utilisant un calibrage géométrique de la caméra, il estime la distance de visibilité. L'inconvénient du système est qu'il peut fournir des faux résultats à cause de la variabilité des scènes et de leur éclairage. Une solution est d'utiliser une trentaine d'image successives (filtre temporel) pour augmenter la précision.

Nous allons montrer dans la suite de ce chapitre des méthodes plus performantes et plus récentes utilisant le modèle optique du brouillard pour estimer la visibilité atmosphérique par caméra embarquée ainsi que différentes approches permettant d'améliorer la qualité de l'image par temps de brouillard.

Dans ce qui suit, après avoir expliqué le modèle de formation des images de brouillard, nous montrerons les méthodes existantes permettant d'améliorer la visibilité de ce type d'images.

2.2.1 Modèle optique du brouillard

Nous appelons optique atmosphérique la branche qui s'intéresse à l'interaction de la lumière avec l'atmosphère. Elle s'intéresse aux phénomènes de diffusion, d'absorption et d'émission.

La diffusion est le phénomène qui nous intéresse dans le cadre de cette étude. Elle dépend de multiples facteurs : taille, forme et propriétés matériels de la particule et leur concentration dans l'atmosphère. La lumière incidente avant d'atteindre l'observateur va subir de multiples diffusions causées par les particules qui se trouvent sur son chemin.

2.2.1.1 Coefficient d'extinction atmosphérique

Le coefficient d'extinction atmosphérique est la proportion de flux lumineux qu'un faisceau de rayons lumineux parallèles émanant d'une source à incandescence à une température de couleur de 2700 K perd en parcourant dans l'atmosphère une longueur égale à l'unité de distance. Le coefficient d'extinction atmosphérique dépend de l'absorption et de la diffusion d'un flux lumineux pénétrant dans l'atmosphère. Dans l'atmosphère, l'effet d'absorption est le plus souvent négligeable et le coefficient d'extinction est considéré comme égal au coefficient de diffusion. Ce n'est toutefois plus le cas en présence de neige, de brouillard se congelant (brouillard givrant), ou de particules de pollution en suspension. Les coefficients d'extinction atmosphérique mesurés varient entre $2 \cdot 10^{-5}$ et $2 \cdot 10^{-1} \text{ m}^{-1}$, ce qui implique, que pour mesurer ce coefficient à l'aide d'un appareil, il doit avoir une sensibilité très importante.

2.2.1.2 Diffusion et longueur d'onde

La diffusion dépend généralement de la longueur d'onde. La couleur bleue du ciel et la couleur bleuâtre d'une brume au niveau des montagnes sont des exemples illustrant cette dépendance. Dans ces deux cas, la diffusion de la longueur d'onde bleu est plus importante que les autres longueurs d'ondes. Toutefois le brouillard et une brume dense vont diffuser toutes les longueurs d'onde plus ou moins de la même façon.

La relation entre β , le coefficient d'extinction atmosphérique, et λ , la longueur d'onde de la lumière incidente, est donnée par la loi de Rayleigh : $\beta \propto \frac{1}{\lambda^\gamma}$, avec γ un coefficient qui dépend de la taille de la particule et qui varie entre 0 et 4. Plus la taille est petite par rapport à la longueur d'onde plus γ est grand.

Dans le cas d'un air pur, la taille des particules est petite ($10^{-4} \mu\text{m}$) et donc $\gamma = 4$, dans ce cas les longueurs d'onde les plus petites (bleu) dominent (car β est inversement proportionnel à λ) ce qui explique la couleur du ciel. Dans le cas du brouillard, la taille des particules est d'environ $1 \mu\text{m}$, par conséquent γ tend vers 0 et donc toutes les longueurs d'onde seront diffusées de la même façon.

Le tableau 2.2.1.2 montre les tailles de différentes particules [Narasimhan 2002]. Les tailles des particules de nuage, du brouillard et de la pluie sont beaucoup plus grandes que celles de l'air ou de la brume. Pour les plus grandes particules l'extinction est constante dans les domaines ultraviolet, visible et proche infrarouge.

Météo	Type de la particule	Taille(μm)
Air	Molécule	10^{-4}
Brume	Aérosol	$10^{-2} - 1$
Brouillard	gouttelette d'eau	$1 - 10$
Nuages	gouttelette d'eau	$1 - 10$
Pluie	gouttes d'eau	$10^2 - 10^4$

Dans le cas du brouillard diurne, nous observons deux effets de la diffusion :

l'atténuation et le voile atmosphérique [Nayar 1999]. Nous allons expliquer dans la suite ces deux manifestations.

2.2.1.3 Atténuation

Soit un volume contenant des particules illuminées par un flux lumineux $E(\lambda)$ comme le montre la figure 2.12.

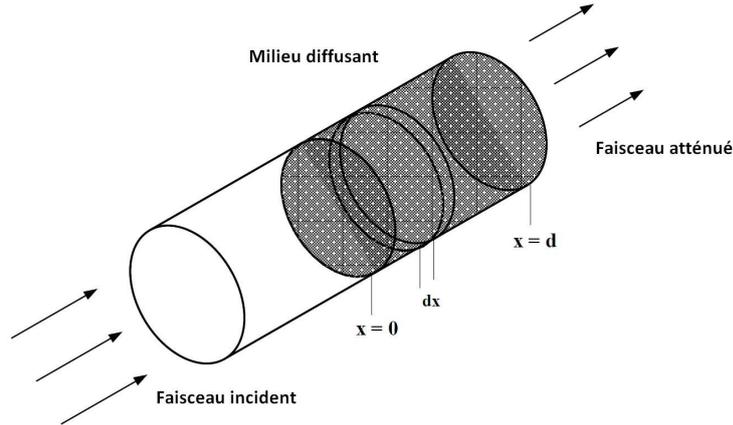


FIGURE 2.12 – Atténuation directe

L'intensité de la lumière $I(\theta, \lambda)$ diffusée dans la direction θ par une partie infinitésimale de ce volume, d'épaisseur dx s'écrit :

$$I(\theta, \lambda) = \beta(\theta, \lambda)E(\lambda)dx \quad (2.37)$$

avec $\beta(\theta, \lambda)$ le coefficient de diffusion angulaire.

Le flux total diffusé dans toutes les directions par ce petit volume est alors :

$$\phi(\lambda) = \beta(\lambda)E(\lambda)dx \quad (2.38)$$

avec $\beta(\lambda)$ le coefficient de diffusion total (dans toutes les directions).

La variation relative de $E(\lambda)$ à la position x s'écrit :

$$\frac{dE(x, \lambda)}{E(x, \lambda)} = -\beta(\lambda)dx \quad (2.39)$$

En intégrant des deux côtés entre $x = 0$ et $x = d$, nous trouvons la loi exponentielle de Bouguer's découverte en 1729 :

$$E(d, \lambda) = E_0(\lambda)\exp(-\beta d) \quad (2.40)$$

Cette équation montre que lorsque la densité du brouillard et la distance de l'objet observé augmentent, la luminance de l'objet s'atténue exponentiellement.

2.2.1.4 Le voile atmosphérique

Un autre phénomène appelé le voile atmosphérique ou “Airlight” fait que l’atmosphère se comporte comme une source de lumière. Il est provoqué par la diffusion de la lumière environnante par les particules atmosphériques. La lumière environnante peut avoir plusieurs sources : la lumière directe du soleil, la lumière diffuse du ciel et la lumière réfléctée par le sol. Alors que l’atténuation directe diminue lorsque la distance augmente, le voile atmosphérique augmente avec la distance.

Considérons le schéma sur la figure 2.13. Nous supposons que l’illumination de l’environnement est constante et que son intensité, son spectre et sa direction sont inconnus.

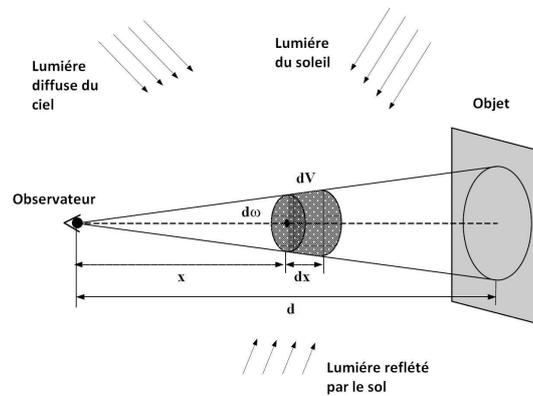


FIGURE 2.13 – Voile atmosphérique

Après la diffusion de la lumière environnante par un volume infinitésimal dV en direction de l’observateur, son intensité devient :

$$dI(x, \lambda) = dV k \beta(\lambda) = dw x^2 dx k \beta(\lambda) \quad (2.41)$$

Avec dw l’angle solide de l’objet par rapport à l’observateur, x la distance de l’objet à l’observateur, $\beta(\lambda)$ le coefficient de diffusion total, k une constante qui dépend de la nature exacte de l’illumination et des particules atmosphériques.

Le volume dV se comporte comme une source de lumière. D’après l’équation 2.40 nous avons :

$$dE(x, \lambda) = dE_0(\lambda) \exp(-\beta(\lambda)x). \quad (2.42)$$

La loi de l’inverse du carré de la distance des faisceaux divergent d’une source ponctuelle nous donne : $E_0 = \frac{I}{x^2}$. À partir de cette formule et de l’équation 2.42, nous trouvons :

$$dE(x, \lambda) = \frac{dI(x, \lambda) \exp(-\beta(\lambda)x)}{x^2} \quad (2.43)$$

D’un autre coté nous avons le rayonnement de dV qui s’écrit :

$$dL(x, \lambda) = \frac{dE(x, \lambda)}{dw} \quad (2.44)$$

À partir des équations 2.44 et 2.43, nous obtenons :

$$dL(x, \lambda) = \frac{dI(x, \lambda) \exp(-\beta(\lambda)x)}{dwx^2} \quad (2.45)$$

à partir des équations 2.41 et 2.45, nous trouvons :

$$dL(x, \lambda) = k\beta(\lambda) \exp(-\beta(\lambda)x) dx \quad (2.46)$$

En intégrant cette équation entre $x = 0$ et $x = d$, nous trouvons le rayonnement total du trajet optique entre l'objet et l'observateur :

$$L(d, \lambda) = k(1 - \exp(-\beta(\lambda)d)) \quad (2.47)$$

Lorsque l'objet se trouve à une distance infinie (à l'horizon) nous avons :

$$L_h(\infty, \lambda) = k \quad (2.48)$$

En remplaçant k dans l'équation 2.47, nous obtenons l'expression du voile atmosphérique :

$$L(d, \lambda) = L_h(\infty, \lambda)(1 - \exp(-\beta(\lambda)d)) \quad (2.49)$$

L'avantage de cette équation est que nous n'avons pas besoin de connaître k puisque la valeur de l'intensité lumineuse à l'horizon L_h nous suffit pour estimer le voile atmosphérique. Notons que dans le cas d'un brouillard dense et des objets proches ou bien dans le cas d'un brouillard léger et des objets lointains, la luminosité des objets est due en grand partie au voile atmosphérique.

Enfin la distribution spectrale reçue par un observateur est la somme de la transmission directe et du voile atmosphérique :

$$E(d, \lambda) = E_{dt}(d, \lambda) + E_a(d, \lambda) \quad (2.50)$$

Cette équation correspond au modèle de Koschmieder découvert en 1924 [Middleton 1983]. Ce modèle va nous servir tout au long de cette thèse pour restaurer les images dégradées par le brouillard.

Modèle de formation des images dans le brouillard

En considérant que la caméra possède une réponse linéaire, nous pouvons appliquer le modèle de Koschmieder directement aux intensités des pixels de l'image. En considérant I l'image observée, A le voile atmosphérique, I_∞ l'intensité du ciel, T la transmittance et I_0 l'image en absence de brouillard, le modèle de Koschmieder devient :

$$I = I_0T + A \quad (2.51)$$

avec $T = e^{-\beta d}$ et $A = I_\infty(1 - e^{-\beta d})$

Nous allons maintenant présenter différentes méthodes (interactives et automatiques) de l'état de l'art utilisant le modèle de Koschmieder pour effectuer la restauration d'images de brouillard.

2.2.2 Méthodes Interactives

2.2.2.1 Restauration en utilisant un filtre polarisant

La lumière peut être décrite comme une onde électromagnétique telle que les champs électrique \vec{E} et magnétique \vec{B} sont orthogonaux. La direction de \vec{E} caractérise la polarisation de la lumière incidente. Les filtres polarisant absorbent les vibrations lumineuses ne laissant passer la lumière que dans une certaine direction.

Les filtres polarisant ont été utilisés depuis longtemps pour atténuer le voile atmosphérique dans les images : le polariseur est orienté de façon à obtenir le meilleur contraste de la scène photographiée.

Toutefois nous obtenons des bons résultats avec cette méthode seulement dans le cas où la diffusion est très faible (particules de petites tailles comme l'air naturel), comme le montre la figure 2.14. Dans le cas du brouillard, un polariseur ne suffit pas pour retrouver les couleurs et les contrastes de la scène, voir les images de la première ligne sur la figure 2.16.



FIGURE 2.14 – À gauche, image acquise sans polariseur ; à droite image acquise avec polariseur

Nous allons définir quelques notions avant de présenter une méthode de restauration d'images de brouillard utilisant un filtre polarisant [Schechner 2001].

Degré de polarisation

Le plan d'incidence est le plan formé par le rayon de la lumière incidente sur une particule atmosphérique et la ligne entre cette particule et la caméra (voir figure 2.15). Le voile atmosphérique est partagé en deux composantes : une parallèle au plan d'incidence A_{\parallel} et une perpendiculaire à ce plan A_{\perp} .

La figure 2.15 illustre le principe d'utilisation du polarisant. La composante du voile parallèle au plan d'incidence est transmise de la meilleure façon lorsque l'angle α du filtre est égal à θ_{\parallel} .

Le degré de polarisation du voile atmosphérique est défini par le taux :

$$P = \frac{A_{\perp} - A_{\parallel}}{A_{\perp} + A_{\parallel}} = \frac{A_{\perp} - A_{\parallel}}{A} \quad (2.52)$$

et il dépend de la taille des particules, de leur densité et de la direction d'observation.

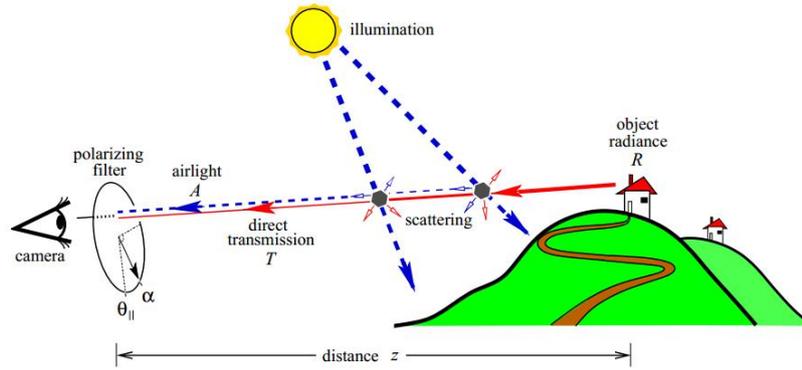


FIGURE 2.15 – Phénomènes d’atténuation et du voile atmosphérique. La composante du voile parallèle au plan d’incidence est polarisée de la meilleure façon lorsque $\alpha = \theta_{\parallel}$

Lorsque la lumière incidente fait un angle de 90° par rapport à la direction d’observation, le voile atmosphérique est totalement polarisé $P = 1$. En orientant donc le filtre parallèlement au plan d’incidence de la lumière nous éliminons le voile atmosphérique de l’image capturée.

Dépolarisation

Le degré de polarisation diminue (dépolarisation) lorsque l’angle entre la lumière incidente et la direction d’observation s’éloigne de 90° ou lorsque la lumière incidente subit de multiples diffusions créant chacune un plan d’incidence différent (changement d’orientation). La dépolarisation dépend aussi de la longueur d’onde.

Lorsque la taille et la densité des particules sont importantes (brouillard, par exemple) la lumière subit de multiples diffusions et est donc dépolarisée.

Le changement de polarisation de la transmission directe est négligeable par rapport à celui du voile atmosphérique (sauf pour les surfaces spéculaires proches de l’observateur).

Restauration

L’auteur [Schechner 2001] a proposé une méthode de restauration utilisant deux images et qui suppose que le voile atmosphérique est partiellement polarisé et que la polarisation de la transmission directe est négligeable. Les variations résultantes de la rotation du filtre polarisant sont dues principalement au voile atmosphérique.

Deux images de la scène sont obtenues en orientant le filtre parallèlement (θ_{\parallel}) et ensuite perpendiculairement (θ_{\perp}) par rapport au plan d’incidence. L’image observée sans utiliser le filtre polarisant est la somme de ces deux images.

$$I = I_{\parallel} + I_{\perp} \quad (2.53)$$

avec

$$I_{\parallel} = T/2 + A_{\parallel} \quad (2.54)$$

et

$$I_{\perp} = T/2 + A_{\perp} \quad (2.55)$$

T étant la transmission directe,

$$A_{\parallel} = A(1 - P)/2 \quad (2.56)$$

et

$$A_{\perp} = A(1 - P)/2 \quad (2.57)$$

Nous pouvons donc, en connaissant P , estimer le voile atmosphérique de n'importe quel point de la scène :

$$A = \frac{I_{\perp} - I_{\parallel}}{P} \quad (2.58)$$

Nous pouvons ensuite déduire la transmission directe :

$$T = I - A \quad (2.59)$$

Pour effectuer la restauration, nous avons donc besoin de connaître le degré de polarisation P et l'intensité du ciel A .

L'intensité du ciel est estimée de manière interactive.

Pour estimer P nous considérons une zone du ciel. À partir de cette zone nous pouvons déduire :

$$P = \frac{A_{\infty}^{\perp} - A_{\infty}^{\parallel}}{A_{\infty}^{\perp} + A_{\infty}^{\parallel}} \quad (2.60)$$

La figure 2.16 montre les résultats de restauration par cette méthode.

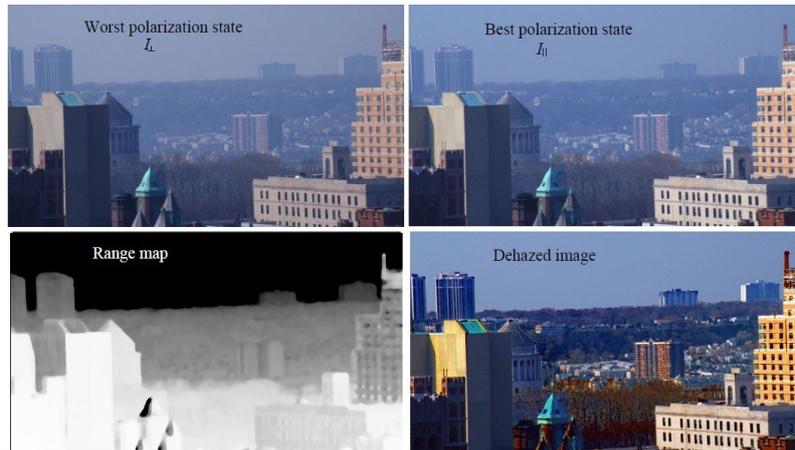


FIGURE 2.16 – En haut, images polarisées I_{\parallel} et I_{\perp} ; en bas, carte de profondeur et image restaurée

Une autre méthode de restauration, que nous allons présenter dans ce qui suit, utilise deux images de la même scène correspondant à des densités de brouillard différentes.

2.2.2.2 Deux images acquises dans des conditions météorologiques différentes

La méthode de restauration proposée par [Narasimhan 2003b] suppose que les conditions météorologiques (taille et densité des particules) ne changent pas spatialement dans le champ de vision du capteur (quelques kilomètres) et que les propriétés de diffusion atmosphériques sont fonction de la longueur d'onde de la lumière.

Le changement d'intensité dans l'image est utilisé comme contrainte pour retrouver la structure de la scène (variations de profondeur) en utilisant deux images prises dans des conditions météorologiques diurnes différentes et inconnues. Ensuite le contraste des images est restauré.

Le modèle décrivant l'éclairement total reçu par un capteur est le suivant :

$$E(d, \lambda) = E_{dt}(d, \lambda) + E_a(d, \lambda) \quad (2.61)$$

avec d la distance séparant l'objet et le capteur et λ la longueur d'onde.

Le premier terme de l'équation correspond à l'éclairement énergétique atténué par les particules et reçu par le capteur :

$$E_{dt}(d, \lambda) = \frac{E_\infty(\lambda)\rho(\lambda)\exp(-\beta(\lambda)d)}{d^2} \quad (2.62)$$

avec β le coefficient d'extinction atmosphérique et ρ une fonction qui décrit les propriétés de réflectance d'un point de la scène.

Le deuxième terme est le voile atmosphérique :

$$E_a(d, \lambda) = E_\infty(\lambda)(1 - \exp(-\beta(\lambda)d)) \quad (2.63)$$

Dans le cas d'une image capturée par une caméra monochrome l'équation 2.61 devient :

$$E = I_\infty \rho \exp(-\beta d) + I_\infty (1 - \exp(-\beta d)) \quad (2.64)$$

avec I_∞ l'intensité du ciel et ρ la radiance normalisée d'un point de la scène.

Afin de restaurer le contraste de l'image, la première étape consiste à segmenter l'image en régions ayant des profondeurs égales, sans nécessairement que cette dernière soit connue.

Détection des variations de profondeur

Considérons une petite partie de l'image où tous les pixels sont à la même profondeur. Soit l'intensité moyenne :

$$\bar{E} = I_\infty \bar{\rho} \exp(-\beta d) + I_\infty (1 - \exp(-\beta d)) \quad (2.65)$$

et l'écart type

$$\sigma_E = I_\infty \exp(-\beta d) \sqrt{\frac{1}{n} \sum_{i=1}^n (\rho_i - \bar{\rho})^2} \quad (2.66)$$

À partir de l'équation 2.64 nous obtenons, après un simple calcul :

$$\frac{E_i - \bar{E}}{\sigma_E} = \frac{\rho_i - \bar{\rho}}{\sqrt{\frac{1}{n} \sum_{i=1}^n (\rho_i - \bar{\rho})^2}} \quad (2.67)$$

avec E_i la valeur d'un pixel dans la région considérée. Cette équation est indépendante de β et I_∞ .

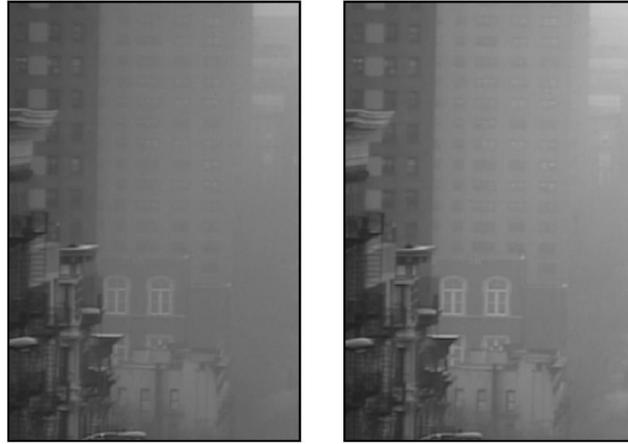


FIGURE 2.17 – Même scène capturée sous deux brouillards différents

Nous considérons deux images de la même scène prises sous des conditions météorologiques différentes, comme sur la figure 2.17. Pour des régions ayant la même profondeur, les deux courbes $\frac{E_i - \bar{E}}{\sigma_E}$ (équation 2.67) sont identiques (figure 2.18). Pour des régions contenant des variations de profondeur, les deux courbes sont différentes (figure 2.19).

En utilisant la SSD (somme des différences au carrés) nous pouvons déterminer le degré de ressemblance des courbes (2.67) de la même région dans les des deux images et ainsi déterminer si la région analysée contient des variations de profondeur ou non.

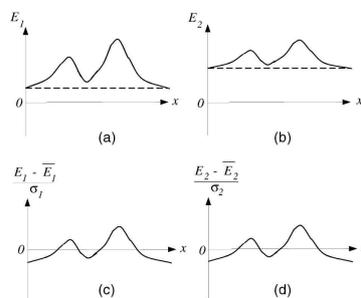


FIGURE 2.18 – Courbes dans une région ayant la même profondeur

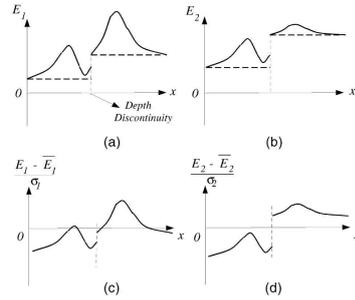


FIGURE 2.19 – Courbes dans une region ayant des profondeurs différentes

Pour détecter les variations de profondeur dans la scène, les contours de l'une des deux images sont d'abord détectés à l'aide d'un filtre de Canny. Pour chaque pixel appartenant à ses contours, la SSD est appliquée sur un voisinage de 15×15 pour comparer les deux images. Lorsque la valeur de la SSD est grande, le pixel est considéré comme variation de profondeur, sinon il est classé comme variation de luminosité. La figure 2.20 montre le résultat de la détection.



FIGURE 2.20 – Détection des variations de profondeur et de luminosité. En blanc, les contours correspondant à une variation de profondeur. En noir, les contours correspondant à une variation de luminosité.

Notons que cette méthode est très sensible au bruit. Une autre approche que nous allons présenter maintenant donne de meilleurs résultats.

Estimation de la profondeur de la scène

Considérons maintenant un pixel ayant les valeurs d'intensités E_1 et E_2 dans chacune des images. En éliminant ρ , qui est indépendant des conditions météorologiques, de l'équation 2.64, nous obtenons :

$$E_2 = \left[\frac{I_{\infty 2}}{I_{\infty 1}} \exp(-(\beta_2 - \beta_1)d) \right] E_1 + [I_{\infty 2}(1 - \exp(-(\beta_2 - \beta_1)d))] \quad (2.68)$$

Cette équation montre que pour deux points ayant la même profondeur, le tracé E_1 en fonction de E_2 est une ligne (figure 2.21).

En divisant l'image en blocs, nous faisons une régression des lignes de points (E_2, E_1) dans chaque bloc. Si la régression est correcte (points alignés le long d'une

ligne), les pixels sont considérés à la même profondeur.

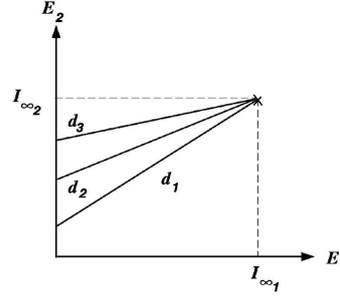


FIGURE 2.21 – Chaque ligne correspond à une profondeur et l'intersection de toutes les lignes nous donne les intensités du ciel des deux images

Dans le cas où $E_1 = I_{\infty 1}$, nous obtenons $E_2 = I_{\infty 2}$ ce qui signifie que l'intersection des lignes correspondant à des profondeurs différentes est le point $I_{\infty 1}$ et $I_{\infty 2}$: les intensités du ciel dans les deux images. Pour estimer l'intensité du ciel nous avons donc besoin d'utiliser au moins deux blocs.

À partir de l'équation 2.68 nous obtenons, à un facteur près, la carte de profondeur de la scène :

$$(\beta_1 - \beta_2)d = -\ln\left(\frac{I_{\infty 2} - E_2}{I_{\infty 1} - E_1}\right) - \ln\left(\frac{I_{\infty 1}}{I_{\infty 2}}\right) \quad (2.69)$$

Restauration du contraste de régions ayant la même profondeur

Soit un ensemble de points j ayant la même profondeur qu'un pixel i . L'équation 2.64 nous donne :

$$\rho_i = 1 - \left(\sum_j 1 - \sum_j \rho_j \right) \frac{I_{\infty} - E_i}{\sum_j (I_{\infty} - E_j)} \quad (2.70)$$

En connaissant l'intensité du ciel et en ayant accès à une segmentation en profondeur de la scène, nous appliquons l'équation 2.70 en chaque pixel pour effectuer la restauration.

Restauration à partir de la structure de la scène

Notons que les deux méthodes de détection des profondeurs fonctionnent bien dans les zones texturées, alors qu'elles ne sont pas efficaces lors de variations graduelles de profondeur (par exemple les montagnes ou une vue oblique de la route). Nous allons présenter une méthode qui permet de prendre en compte ses lentes variations de profondeurs.

Nous supposons qu'il existe une zone E_{zero} dont la transmission directe est égale à zéro du fait soit de sa radiance qui est nulle, soit de son grand éloignement de la caméra. Cette zone peut être sélectionnée manuellement ou détectée avec la méthode

de [Narasimhan 2000]. E_{zero} n'apparaît jamais comme ayant une intensité de zéro à cause du voile atmosphérique qui vient s'ajouter à son intensité. La profondeur optique de cette zone est donc d'après l'équation 2.69 :

$$\beta d_{zero} = -\ln(1 - E_{zero}/I_{\infty}) \quad (2.71)$$

La profondeur optique de n'importe quel point de la scène s'obtient avec :

$$\beta d_i = (\beta d_{zero}) \left(\frac{d_i}{d_{zero}} \right) \quad (2.72)$$

Le deuxième terme $\frac{d_i}{d_{zero}}$ se calcule à partir de l'équation 2.69. La restauration de ρ est ensuite effectuée avec l'équation 2.64. La figure 2.22 montre le résultat de restauration en utilisant deux images réelles de la même scène prises dans des conditions météorologiques différentes.

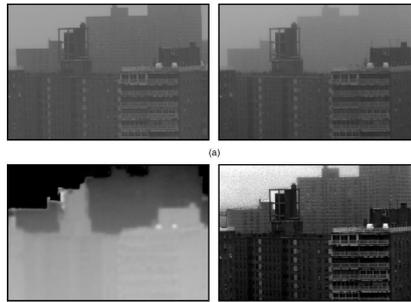


FIGURE 2.22 – En haut, images sous deux conditions de brouillard différentes; en bas, carte de profondeur et image restaurée

Rappelons que la caméra doit avoir une position fixe. Dans le cas où il existe des objets en mouvement dans la scène, la méthode ne restaurera pas leur contraste puisque leur profondeur est inconnue. Dans ce cas, il faudra séparer ses objets du fond et leurs assigner la profondeur minimale des pixels voisins.

Une autre méthode de restauration, que nous allons présenter dans la suite, utilise une seule image mais nécessite l'interaction avec l'utilisateur.

2.2.2.3 Utilisation d'une seule image et d'informations fournies par l'utilisateur

Nous avons présenté des méthodes de restauration utilisant de multiples images prises sous différentes conditions météorologiques ou prises en modifiant l'optique du capteur. Dans certains cas, il n'est pas possible d'acquérir de multiples images. Dans [Narasimhan 2003c] l'auteur propose deux méthodes interactives de restauration utilisant une seule image.

Algorithme -1- : transfert de la couleur

Dans le cas d'une image couleur, le modèle de Koschmieder s'écrit : $\mathbf{E} = p\mathbf{D} + q\mathbf{A}$. Les caractères en gras représentent les vecteurs couleurs. \mathbf{E} correspond à la couleur d'un point d'une scène de brouillard, \mathbf{D} la direction de la couleur d'un point de la scène en absence de brouillard, \mathbf{A} la direction de la couleur du voile atmosphérique, $p = R - \beta d$ et $q = E_\infty(1 - \exp(-\beta d))$. La figure 2.23 illustre ce modèle.

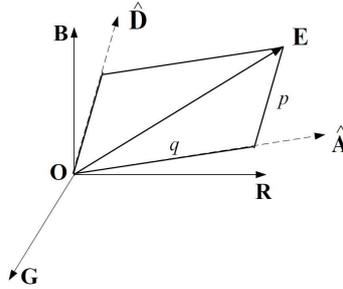


FIGURE 2.23 – Modèle dichromatique de diffusion de la lumière

Des objets de la scène ayant la même radiance (route ou arbres par exemple) et se trouvant à des distances différentes vont avoir des intensités différentes mais vont garder la même direction de couleur. L'algorithme proposé utilise la couleur des points de la scène les plus proches de la caméra (qui sont les moins corrompus par le brouillard) afin de corriger la couleur des points les plus distants. L'utilisateur doit sélectionner une zone de l'image qui soit la moins dégradée par le brouillard (régions proches) et ensuite sélectionner une région qui ressemble le plus à la couleur du ciel. Pour chaque pixels E_i de l'image, nous cherchons dans la région supposée la moins affectée, les pixels qui lui ressemblent le plus. La recherche est limitée aux pixels de la région la moins touchée satisfaisant la propriété de planarité dichromatique : $E_i \cdot (D \times A) = 0$. À partir de cet ensemble nous sélectionnons le vecteur D_i qui fait le plus grand angle avec le voile atmosphérique : $\min(D \cdot A)$ et nous remplaçons la valeur de E_i par D_i pour corriger les couleurs dégradées de l'image (figure 2.24).

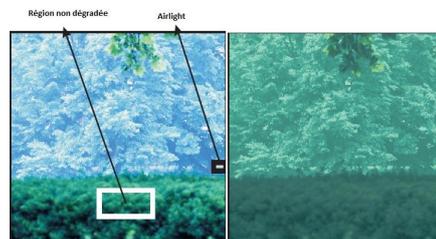


FIGURE 2.24 – À gauche, image originale avec la zone correspondant au voile atmosphérique et la zone ayant une bonne qualité couleur ; à droite, image restaurée

Nous remarquons sur la figure 2.24 que les zones les plus éloignées ont des couleurs moins saturées que dans l'image originale.

Le problème avec cet algorithme est que les couleurs présentes dans l'image n'ont pas toutes leur couleur correspondante dans la région sélectionnée par l'utilisateur. En outre, la méthode ne s'applique qu'à des images couleurs.

Algorithme -2- profondeur heuristique

Cette deuxième méthode se fonde sur un calcul heuristique de la profondeur de la scène [Narasimhan 2003a]. L'utilisateur sélectionne une région du ciel E_∞ et un emplacement approximatif du point de fuite dans le sens croissant de la profondeur. Nous déduisons alors une profondeur heuristique de la scène à l'aide de l'équation :

$$d = d_{min} + \alpha(d_{max} - d_{min}) \quad (2.73)$$

avec α la distance normalisée (0 et 1) d'un pixel par rapport au point de fuite et d_{min} , d_{max} les distances approximatives minimales et maximales fournies par l'utilisateur. La figure 2.25 montre un exemple de carte de profondeur obtenue.

La restauration est ensuite effectuée en inversant le modèle de Koschmieder. Pour cela, il faut le coefficient β . En supposant β constant spatialement, l'auteur propose de le faire varier manuellement jusqu'à obtenir une image contenant le moins de brouillard.

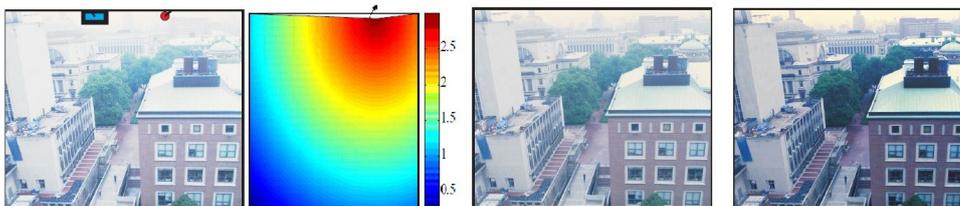


FIGURE 2.25 – De gauche à droite : image originale, profondeur heuristique et les images restaurées avec deux β différents.

Le défaut de cette méthode est qu'elle n'est pas très appropriée pour des scènes avec de fortes variations de profondeur.

Dans le cas de caméras embarquées dans le véhicule, les méthodes que nous venons de présenter dans cette section ne sont pas appropriées. Dans la suite, nous allons montrer des méthodes utilisant une seule image et complètement automatiques qui conviennent mieux aux applications d'aide à la conduite.

2.2.3 Méthodes automatiques utilisant une seule image

Nous avons vu dans la partie précédente des méthodes de restauration d'images de brouillard utilisant le modèle de Koschmieder et plusieurs images prises dans des conditions différentes ou des informations fournies par l'utilisateur afin d'estimer les différentes inconnues du modèle. Dans ce qui suit nous allons voir des méthodes utilisant une seule image et des contraintes sur la scène pour effectuer la restauration et qui sont plus adaptées à notre problématique ADAS.

2.2.3.1 Hypothèse monde plan

L'auteur [Hautière 2007] a proposé une méthode utilisant une seule image et le modèle optique du brouillard afin d'estimer le coefficient d'extinction atmosphérique. Ensuite, la profondeur de la scène est estimée en supposant la route plane. La restauration est effectuée en inversant le modèle de Koschmieder connaissant β et d . Selon le type d'application ADAS l'auteur a proposé trois variantes de l'algorithme. Nous allons commencer par montrer comment les conditions météorologiques sont détectées.

Profondeur du monde plan

Considérons le schéma sur la figure 2.26. La caméra est positionnée à une hauteur H du sol et fait un angle θ avec l'axe horizontale. Les coordonnées d'un pixel dans le plan image sont notées (u, v) .

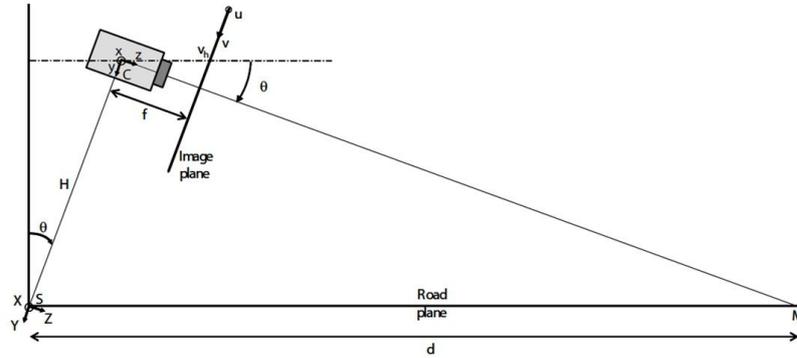


FIGURE 2.26 – Configuration de la caméra

En faisant l'hypothèse que la route est plane (jusqu'à l'horizon) nous pouvons, connaissant les paramètres intrinsèques de la caméra (focale f et taille des pixels t_p), associer à chaque ligne v de l'image une distance d :

$$d = \frac{\lambda}{v - v_h} \quad (2.74)$$

Avec $\lambda = \frac{H\alpha}{\cos^2\theta}$, $\alpha = \frac{f}{t_p}$ et v_h étant la ligne d'horizon dans l'image.

Estimation des conditions météorologiques

À partir de l'équation 2.74 et de l'équation de Koschmieder (2.51), nous obtenons :

$$I = R - (R - A_\infty) \left(1 - e^{-\beta \frac{\lambda}{v - v_h}} \right) \quad (2.75)$$

En cherchant le point d'inflexion de I par rapport aux lignes v de l'image : $\frac{d^2 I}{dv^2} = 0$, nous déduisons le coefficient d'extinction atmosphérique $\beta = \frac{2(v_i - v_h)}{\lambda}$ avec v_i la position du point d'inflexion.

Afin d'estimer v_i nous sélectionnons une bande verticale de l'image, nous calculons la médiane des intensités de chaque ligne de cette bande et détectons le point d'inflexion. La bande considérée doit contenir une zone homogène et le ciel. La méthode de croissance de région est utilisée afin de détecter les zones ayant un minimum de variation de gradient comme le montre la figure 2.27, la bande est ensuite sélectionnée dans cette zone.

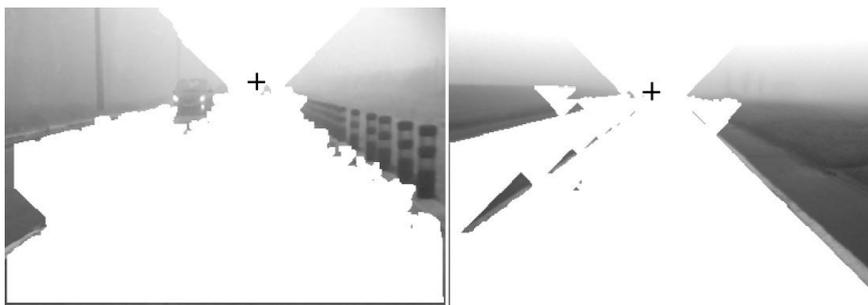


FIGURE 2.27 – Résultat de la croissance de région

La position de la ligne d'horizon v_h peut être estimée en utilisant une centrale inertielle qui nous fournit l'angle que fait le véhicule avec la route, avec une méthode de traitement d'images ou en faisant un calibrage a priori de la caméra.

La figure 2.28 montre la détection de la ligne d'horizon, de la bande verticale et du point d'inflexion pour deux images de brouillard.

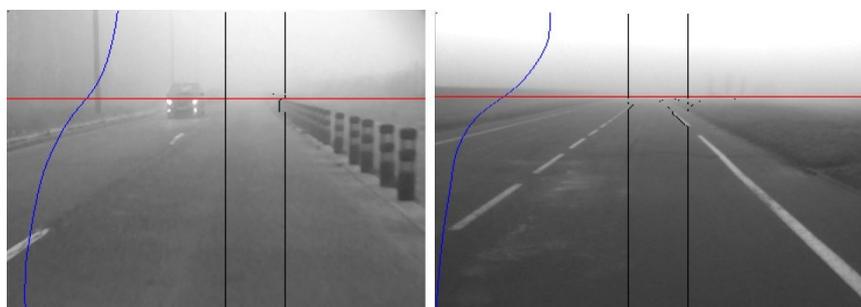


FIGURE 2.28 – Détection du point d'inflexion.

L'intensité du ciel est estimée à partir des valeurs de I et de sa dérivée $\frac{dI}{dv}$ lorsque $v = v_i$: $I_\infty = I_i + \frac{(v_i - v_h)}{2} \frac{dI}{dv} |_{v=v_i}$.

Connaissant I_∞ , β et d nous appliquons le modèle de Koschmieder inverse pour effectuer la restauration. Toutefois, la profondeur des pixels n'appartenant pas au plan de la route est faussée et ils seront donc mal restaurés.

Dans la section 2.2.2.3, nous avons vu que nous pouvons estimer une profondeur heuristique de la scène en utilisant des informations fournies par l'utilisateur : le point de fuite dans le sens croissant de la profondeur ainsi qu'une région du ciel, que nous détectons ici automatiquement :

- Les pixels appartenant au ciel sont sélectionnés comme étant ceux qui se trouvent au-dessus de la ligne d’horizon et qui ont une intensité supérieure à celle de la ligne d’horizon.
- Le point de fuite est estimé comme étant au centre de la surface obtenue par la croissance des régions, sur la ligne d’horizon.

Nous pouvons donc estimer une carte de profondeur heuristique des objets verticaux. La figure 2.29 montre les cartes de profondeur obtenues et le résultat de la restauration avec ces cartes de profondeur.

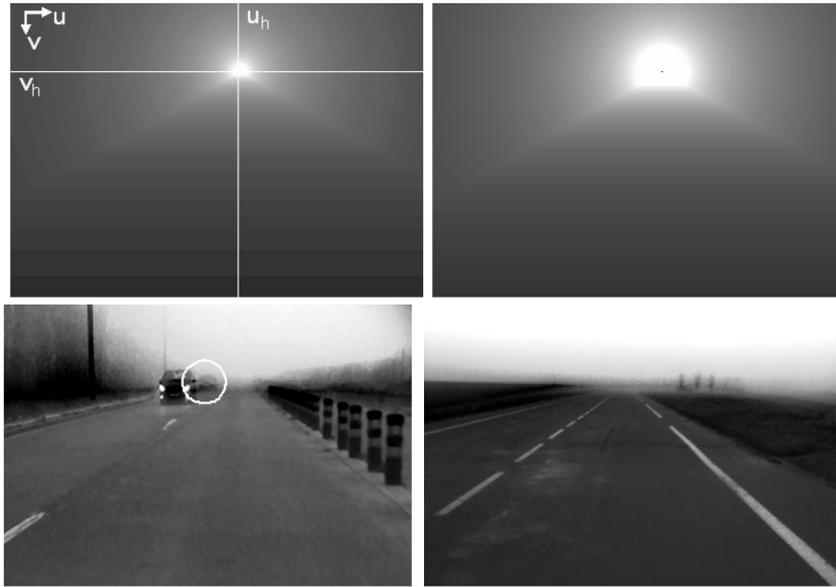


FIGURE 2.29 – En haut, cartes de profondeur heuristiques ; en bas, images restaurées

Pour améliorer la qualité de restauration au niveau des objets verticaux, l’auteur [Hautière 2007] définit un critère de qualité qui mesure la corrélation entre les pixels restaurés et leurs correspondant dans l’image originale. La profondeur attribuée aux pixels n’appartenant pas à la route est celle qui maximise ce critère.

Une autre méthode automatique utilisant le modèle chromatique de la lumière est présentée dans la suite.

2.2.3.2 Utilisation du modèle chromatique

L’idée est de transformer l’équation de Koschmieder sous la forme d’un modèle chromatique [Tan 2007] :

$$\sigma_c = \frac{E_c}{E_r + E_g + E_b} \quad (2.76)$$

avec σ_c la chromaticité de l’image et $c = \{r, g, b\}$

Lorsque la transmission est nulle ($d \rightarrow \infty$), nous obtenons la chromaticité de la lumière :

$$\Gamma_c = \frac{I_\infty^c}{I_\infty^r + I_\infty^g + I_\infty^b} \quad (2.77)$$

Lorsque le voile atmosphérique est absent, nous obtenons la chromaticité de l'objet :

$$\Lambda_c = \frac{I_\infty^c \rho_c}{I_\infty^r \rho_r + I_\infty^g \rho_g + I_\infty^b \rho_b} \quad (2.78)$$

Nous pouvons donc réécrire l'équation de Koschmieder sous cette forme :

$$E(x) = B(x)\Lambda(x) + F(x)\Gamma \quad (2.79)$$

avec $B(x) = (I_\infty^r \rho_r(x) + I_\infty^g \rho_g(x) + I_\infty^b \rho_b(x))e^{-\beta d(x)}$
 et $F(x) = (I_\infty^r + I_\infty^g + I_\infty^b)(1 - e^{-\beta d(x)})$

L'équation 2.79 à la même forme que le modèle de réflexion dichromatique. L'idée de base de la méthode est d'utiliser la corrélation linéaire entre la chromaticité de l'image et celle de la lumière [Tan 2008b] :

$$\sigma_c = p_c \frac{1}{\Sigma E_i} + \Gamma_c \quad (2.80)$$

tel que $\Sigma E_i = E_r + E_g + E_b$
 et $p_c = B(\Lambda_c - \Gamma_c)$

La distribution des points $(\sigma_c, \frac{1}{\Sigma E_i})$ décrit des lignes qui ont différentes pentes p_c mais la même ordonnée à l'origine Γ_c (figure 2.30). Si la pente était constante, une transformée de Hough permettrait de détecter l'intersection Γ_c . Dans le cas plus général, il est possible de compter les intersections des points comme proposé par [Tan 2008b] (figure 2.31). Cette méthode est appliquée sur les 20 % des pixels ayant les intensités les plus hautes dans l'image afin de n'avoir que des zones de brouillard.

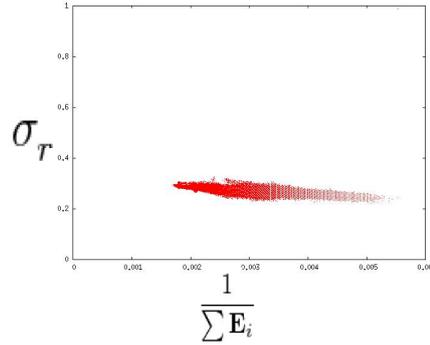
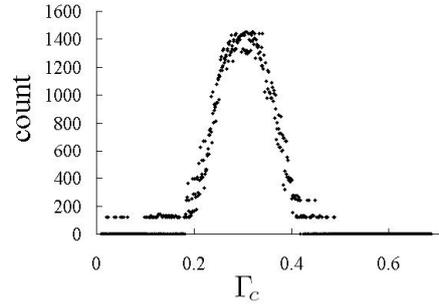


FIGURE 2.30 – Distribution des valeurs $(\sigma_c, \frac{1}{\Sigma E_i})$

En divisant l'image par Γ_c nous obtenons une image normalisée dans laquelle la lumière de l'environnement est blanche (Figure 2.32).

L'auteur propose ensuite d'estimer F comme la luminance de l'image dans l'espace couleur YIQ à laquelle nous appliquons une gaussienne pour une meilleure approximation.

FIGURE 2.31 – Estimation de Γ_c par décompte des intersections

Pour estimer $I_\infty^r + I_\infty^g + I_\infty^b$ nous considérons que la lumière de l'environnement est la plus grande intensité dans l'image. Nous pouvons donc déduire, à partir de la formule de F , le terme $(1 - e^{-\beta d})$ qui correspond au voile atmosphérique et effectuer la restauration en inversant le modèle de Koschmieder.

La figure 2.32 montre le résultat obtenu avec cette méthode :

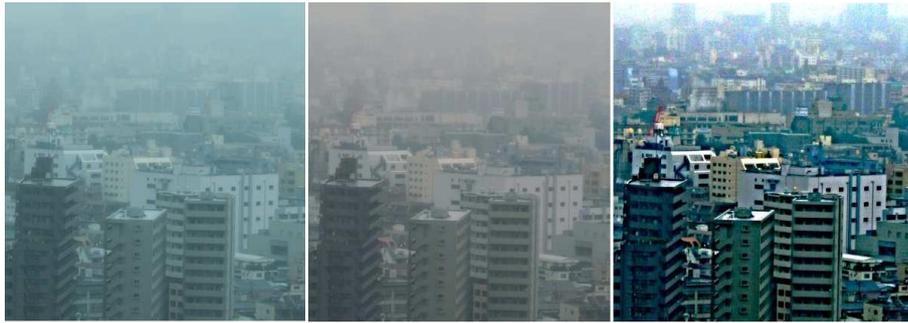


FIGURE 2.32 – De gauche à droite, image originale, image avec la lumière de l'environnement blanche et image restaurée

Autre approche :

Une autre méthode de restauration fondée sur le modèle chromatique [Tan 2008a], utilise le fait que l'image sans brouillard possède un plus grand nombre de contours qu'une image avec brouillard à cause de la perte du contraste due à la diffusion. Pour améliorer la visibilité la contrainte suivante est donc utilisée :

$$C_{edges} \left(\frac{B\Lambda}{e^{-\beta d}} \right) > C_{edges}(\mathbf{p}) \quad (2.81)$$

tel que le terme $\frac{B\Lambda}{e^{-\beta d}}$ correspond à l'image sans brouillard, \mathbf{p} une petite fenêtre de l'image, supposée contenir des objets ayant la même profondeur et $C_{edges}(I)$ le nombre de contours de I :

$$C_{edges}(I) = \Sigma | \nabla I_c(x) | \quad (2.82)$$

Une autre contrainte utilisée, exprimant que l'intensité maximale est celle du ciel, découle du modèle de Koschmieder :

$$0 \leq \frac{B\Lambda}{e^{-\beta d}} \leq L_\infty^c \quad (2.83)$$

La figure 2.33 montre la distribution de $C_{edges}(\frac{B\Lambda}{e^{-\beta d}})$ en fonction de F pour une zone découpée dans l'image. Nous remarquons que le nombre de contours atteint le maximum avant de décroître du fait de la seconde contrainte.

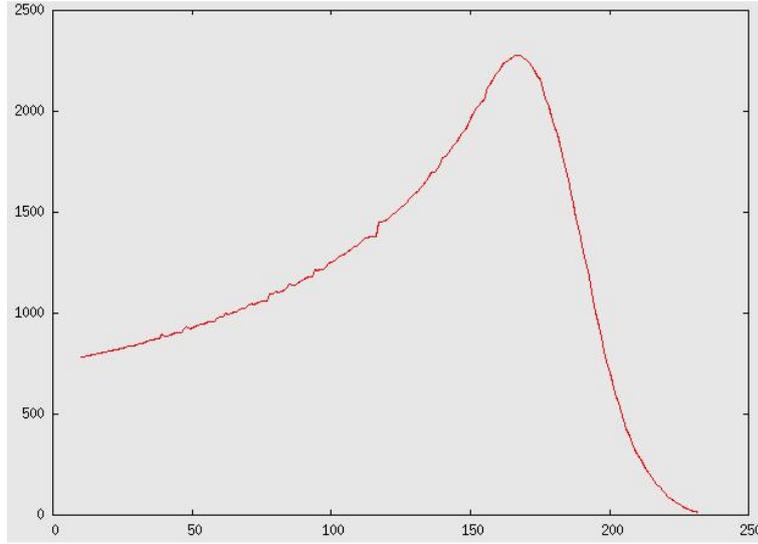


FIGURE 2.33 – Distribution de $C_{edges}(\frac{B\Lambda}{e^{-\beta d}})$ en fonction de F

En supposant le voile atmosphérique F lisse sur toute l'image, nous pouvons le modéliser par des champs de Markov aléatoires (MRFs). Dans ce cas la fonction potentiel s'écrit :

$$E(F | p_x) = \Sigma \phi(p_x | F_x) + \eta \sum_{x,y \in N_x} \psi(F_x, F_y) \quad (2.84)$$

avec p_x une fenêtre de petite taille centrée sur x et supposée avoir une valeur constante de F_x , η représente la force du lissage, N_x le voisinage du pixel x .

Le terme d'attache aux données est :

$$\phi(p_x | F_x) = \frac{C_{edges}(\frac{B\Lambda}{e^{-\beta d}})}{m} \quad (2.85)$$

m étant un coefficient de normalisation.

Le terme du lissage est :

$$\psi(F_x, F_y) = 1 - \frac{|F_x - F_y|}{\sum_c L_\infty^c} \quad (2.86)$$

Afin de retrouver toutes la valeurs de F_x , nous devons maximiser la probabilité $p(F_x)$, décrite par la probabilité de Gibbs en utilisant les Graph-cuts par exemples.

La valeur initiale de F est choisie comme étant la luminance dans l'espace couleur YIQ. La figure 2.34 montre les résultats de restauration obtenue par cette méthode :



FIGURE 2.34 – Image originale et image restaurée

La limite de cette méthode est le temps de traitement qui est de 5 à 7 minutes pour traiter une image de taille 600×400 sur un PC P4 de 1Go de mémoire.

Nous présentons dans ce qui suit une méthode plus rapide de restauration d'images de brouillard.

2.2.3.3 Dark Channel

La méthode [He 2009] est fondée sur le constat que dans la plupart des images en extérieur non affectées par le brouillard, les régions n'appartenant pas au ciel possèdent au moins un canal ayant une intensité très faible. Cette faible intensité est due principalement à 3 facteurs : les ombres, les objets colorés et les objets sombres. L'auteur a vérifié cette propriété du canal sombre en parcourant 5000 images de l'extérieur avec une fenêtre Ω de taille 15×15 et en calculant sur chaque fenêtre ω le canal sombre du pixel x défini par l'équation 2.87.

$$I^{dark}(x) = \min_{c \in \{r,g,b\}} (\min_{y \in \Omega(x)} (I^c(y))) \quad (2.87)$$

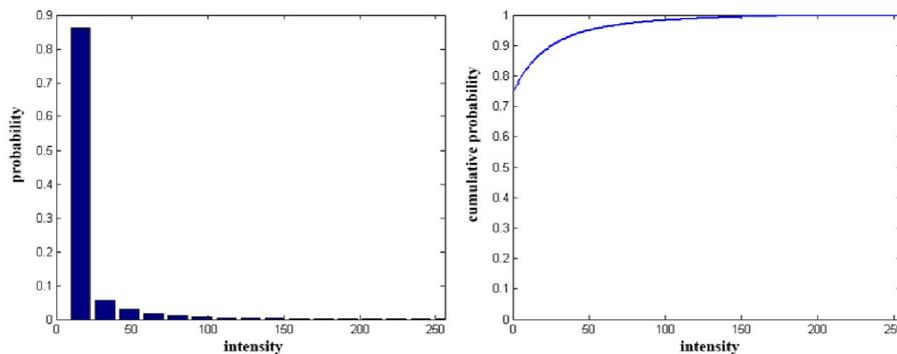


FIGURE 2.35 – Statistiques du Dark Channel calculées sur 5000 images naturelles variées : histogramme de 5000 images sans brouillard (à gauche) et histogramme cumulatif (à droite). 75 % des pixels ont une intensité nulle (hypothèse Dark Channel).

La figure 2.35 montre l'histogramme et l'histogramme cumulé du canal sombre des 5000 images. Nous remarquons que 75 % des pixels ont un canal d'intensité nulle et 90 % ont une intensité inférieure à 25. Ce qui nous donne l'hypothèse "Dark channel" :

$$I^{dark}(x) = 0 \quad (2.88)$$

Le "Dark Channel" nous permet d'obtenir une approximation grossière de la densité du brouillard. En introduisant l'équation 2.89 dans Koschmieder nous obtenons :

$$t(x) = 1 - \min_{c \in \{r, g, b\}} (\min_{y \in \Omega(x)} (I^c(y)/A^c)) \quad (2.89)$$

avec $t = e^{-\beta d}$.

Afin de garder une petite quantité de brouillard (effet de profondeur plus réel) nous introduisons dans 2.89 un paramètre ω :

$$t(x) = 1 - \omega \min_{c \in \{r, g, b\}} (\min_{y \in \Omega(x)} (I^c(y)/A^c)) \quad (2.90)$$

tel que $\omega = 0, 95$

Nous estimons la luminance atmosphérique L_∞ comme étant la plus grande intensité dans I des 0.1 % des pixels les plus lumineux de J^{dark} .

L'auteur propose d'utiliser un "Soft Mapping" afin de lisser t . La figure 2.36 montre les résultats obtenus par cette méthode.



FIGURE 2.36 – Résultat du "Dark Channel"

Notre explication du "Dark Channel" Nous allons expliquer pourquoi cette méthode permet d'améliorer la qualité d'images couleurs de brouillard et quelles sont ses limites.

À partir du modèle de Koschmieder nous avons :

$$t = \frac{I - I_\infty}{I_0 - I_\infty} \quad (2.91)$$

La figure 2.37 montre le tracé de t en fonction de I_0 pour différentes valeurs de I . Chaque courbe correspond à une valeur de I qui correspond au maximum de I_0 . Nous constatons que l'écart horizontal entre deux courbes différentes (deux canaux) se resserre (désaturation de la couleur) lorsque t augmente (brouillard dense ou objets loin).

Nous considérons que le plus petit canal I^c doit avoir une intensité nulle $I_0^c = 0$ (hypothèse "Dark Channel") : les trois points en haut de la figure correspondent

aux valeurs R,G et B d'un pixel de l'image I . Nous prenons la plus petite valeur de ces trois canaux et considérons qu'elle a une intensité sans brouillard $I_0 = 0$ (nous faisons glisser ce point sur sa courbe jusqu'à l'origine). Nous estimons ensuite la transmittance t à partir de ce canal et effectuons la restauration sur les trois canaux en utilisant le t estimé et le modèle de Koschmieder inverse. Nous obtenons donc, comme le montre la figure 2.37 un plus grand écart entre les différents canaux et donc récupérons la saturation de la couleur.

Notons que cette méthode ne fonctionne que pour des pixels ayant des grandes valeurs d'intensité puisque pour des petites valeurs de I la courbe de $t(I_0)$ est presque linéaire et la différence entre les canaux reste la même après l'application du "Dark channel".

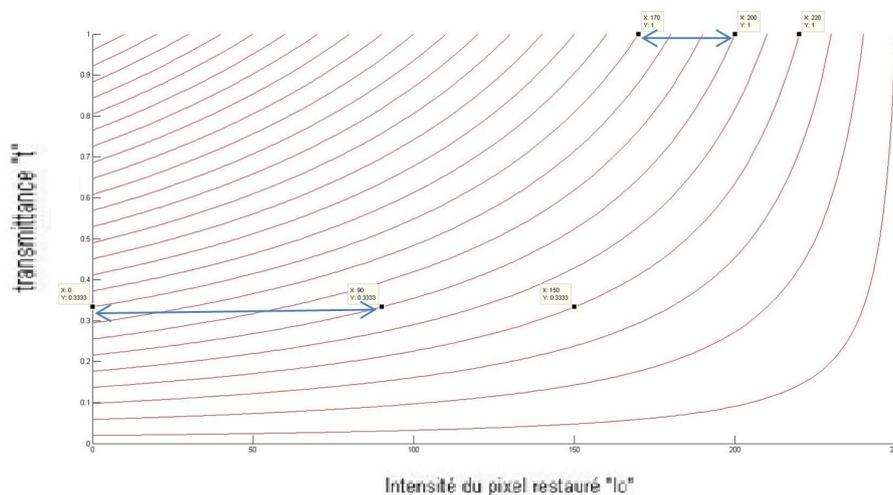


FIGURE 2.37 – Transmittance t en fonction de I_0 pour différentes valeurs de I .

Le problème est que cette méthode ne fonctionne pas dans le cas où le pixel ne contient aucune couleur (3 canaux égaux). C'est pour cette raison que l'auteur propose de chercher à sélectionner le plus petit dark channel sur une fenêtre afin de transférer l'information couleur aux zones sans couleurs.

Nous proposons ici une implémentation plus simple permettant de transférer l'information couleur aux pixels voisins. Pour cela nous appliquons une simple fermeture morphologique à la carte de transmittance obtenue.

La figure 2.38 montre le résultat sans application de la fermeture (restauration des couleurs) et avec fermeture (transfert des couleurs aux zones grises). Nous remarquons que dans les endroits ne contenant pas de couleurs nous avons une intensité nulle qui correspond au "Dark Channel". Notons qu'aucun lissage n'est effectué, d'où les artefacts générés sur les contours.

Le défaut de cette méthode est qu'elle ne s'applique qu'à des images contenant partout des couleurs.



FIGURE 2.38 – Image originale, amélioration de la saturation des couleur, transfert de l'information couleur aux régions voisines

2.2.3.4 Utilisation de contraintes sur le voile atmosphérique

L'approche proposée par l'auteur [Tarel 2009] utilise le modèle optique du brouillard et des contraintes sur le voile atmosphérique afin de restaurer l'image sans brouillard I_0 . L'équation de Koschmieder s'écrit en fonction du voile atmosphérique $V(x, y) = I_\infty(1 - \exp(-\beta d(x, y)))$ sous la forme suivante :

$$I(x, y) = R(x, y) \left(1 - \frac{V(x, y)}{I_\infty} \right) + V(x, y) \quad (2.92)$$

Afin d'estimer l'intensité du ciel I_∞ , une balance des blancs est effectuée de manière globale ou locale de façon à obtenir un ciel uniforme correspondant à la plus grande intensité dans l'image ; $I_\infty = 1$ sur les 3 canaux couleurs et $I \in [0, 1]$.

Ensuite, Deux contraintes sont appliquées sur le voile atmosphérique :

$$(1) V(x, y) \geq 0$$

(2) $V(x, y) \leq W(x, y)$ tel que $W(x, y) = \min(I(x, y))$: le minimum est calculé sur chaque canal couleur (dans le cas d'images en niveaux de gris $W(x, y) = I(x, y)$).

Comme dans [Tan 2008a], la restauration est effectuée de façon à maximiser le contraste en assumant que la profondeur varie d'une manière lisse dans toute l'image à l'exception des bords des objets de la scène. Le problème peut être reformuler en cherchant à maximiser V en supposant que V est lisse :

$$\operatorname{argmax}_V \int_{(x,y)} V(x, y) - \lambda \phi(\|\nabla V(x, y)\|^2) \quad (2.93)$$

sous les contraintes $0 \leq V(x, y) \leq W(x, y)$. λ est un paramètre qui contrôle la force du lissage et ϕ une fonction concave autorisant des grands sauts. Sauf que cette solution est très couteuse en temps de calcul. Une façon de surmonter le problème est d'utiliser, comme dans [He 2009] (contrainte Dark Channel $W(x, y) = 0$ appliquée localement ensuite lissage), un filtrage adapté pour effectuer le lissage.

V est donc calculé comme la différence entre la moyenne locale de $W(x, y)$ et l'écart type local de $W(x, y)$.

La moyenne locale de $W(x, y)$ doit être calculée avec un algorithme qui autorise les grands sauts le long des contours tel qu'un filtre bilatéral ou un filtre médian (plus rapide). La moyenne locale de W est donc calculée de la façon suivante $M = \text{median}_{s_v}(W)$, avec s_v la taille de la fenêtre utilisée. L'écart type est estimé de la même façon avec un filtre médian $\text{median}_{s_v}(|W - M|)$.

Nous obtenons donc une première approximation B de V : $B = M - \text{median}_{s_v}(|W - M|)$. Enfin, en appliquant les 2 contraintes sur V nous obtenons :

$$V(x, y) = \max(\min(pB, W), 0) \quad (2.94)$$

avec p un paramètre choisi égal à 0,95 afin de garder une petite quantité de brouillard et obtenir une image plus réaliste. La taille s_v est choisie égale à 41.

Enfin pour effectuer la restauration, l'équation 2.92 est inversée pour obtenir la formule de R :

$$R(x, y) = \frac{I(x, y) - V(x, y)}{1 - \frac{V(x, y)}{I_\infty}} \quad (2.95)$$

Le problème avec le filtre médian est qu'il ne préserve pas les coins. L'auteur [Tarel 2009] propose d'utiliser un filtre *médian des médians le long des lignes* afin de surmonter ce problème. Afin de réduire la quantité de bruit amplifié par la restauration, l'auteur [Tarel 2009] propose une méthode permettant d'adapter la taille de la fenêtre s_v afin d'obtenir le même écart type de bruit dans les images avec brouillard et restaurée. Enfin, une correction gamma peut être utilisée pour obtenir une image restaurée qui ressemble à l'image originale dans la partie inférieure de l'image (la moins affectée par le brouillard). La figure 2.39 montre des exemples de résultats obtenus par cette méthode. Nous constatons que le contraste de l'image est meilleur que dans l'image originale.

Nous abordons maintenant le deuxième problème de cette thèse concernant les images dégradées par la pluie.

2.3 Images dégradées par les gouttes de pluie

Les altérations générées par la pluie sont de trois types : les gouttes dans l'air, les gouttes sur le pare-brise et la modification de la réflexion dans la scène. De jour, les gouttes dans l'air produisent sur les images des traînées qui ne sont généralement pas perçues par des caméras embarquées. Nous n'aborderons donc pas ce phénomène dans cette thèse. De la même façon, les modifications des propriétés de réflexion de la scène n'est pas le sujet de notre étude. Finalement, nous allons nous concentrer sur les gouttes sur le pare-brise. Pour ce faire, dans un premier temps nous nous intéresserons aux propriétés physiques des gouttes. Puis, nous décrirons des méthodes de détection de gouttes sur le pare-brise.



FIGURE 2.39 – De gauche à droite : image originale et image restaurée par [Tarel 2009]

2.3.1 Physique des gouttes de la pluie

Une goutte d'eau peut être vue comme une lentille grand angle réfléchissant et réfractant les lumières incidentes (à partir d'un grand champ de vue de la scène, de l'ordre de 165°) vers l'observateur [Garg 2007]. Ceci peut affecter la qualité des images acquises par temps de pluie en introduisant des fluctuations temporelles et spatiales dans l'image.

L'intensité d'un pixel dans le cas d'une image de pluie va dépendre des effets dus à la présence de plusieurs gouttes de pluie et de l'angle solide du pixel.

Géométrie des gouttes

Les petites gouttes de pluies possèdent généralement une forme sphérique. Plus leur taille est grande plus leur forme s'aplatit à cause de la pression atmosphérique. L'équation 2.96 [Beard 1987] décrit ces distorsions en approximant le rayon de sphère en fonction de l'angle par rapport à la verticale.

$$r(\theta) = a \left(1 + \sum_{n=1}^{10} c_n \cos(n\theta) \right) \quad (2.96)$$

avec a le rayon de la sphère non-distordue, $c_1 \dots c_{10}$ des coefficients qui dépendent du rayon de la goutte et θ l'angle par rapport à la verticale. La figure 2.40 montre la forme de gouttes de différentes tailles.

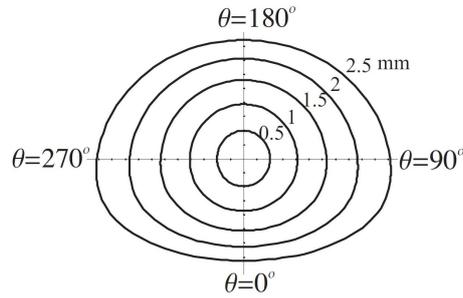


FIGURE 2.40 – Forme des gouttes en fonction de leur taille

La distribution de la taille des gouttes peut être décrite selon le modèle empirique de Marshall et Palmer :

$$N(a) = 8 \times 10^6 e^{-8200 * h^{-0.21} a} \quad (2.97)$$

avec h l'intensité des précipitations en mm/h et $N(a)$ le nombre de gouttes par unité de volume.

Modèle optique de la pluie

Lorsqu'un rayon lumineux rencontre une goutte d'eau il subit une réfraction, une réflexion et une seconde réfraction en quittant la goutte.

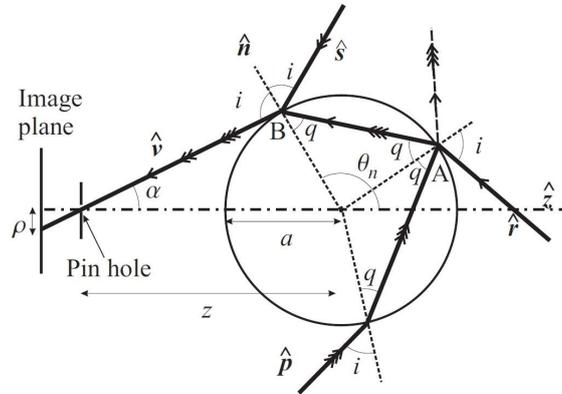


FIGURE 2.41 – Réflexion et réfractions des rayons lumineux par la goutte de pluie

Considérons la figure 2.41. En rencontrant la goutte, les trois rayons incidents r , s et p , vont donner naissance à un rayon émergent du point B en direction de la caméra. La radiance du point B est la somme des radiances dues à la réfraction, la réflexion spéculaire et la réflexion interne, respectivement, L_r , L_s et L_p :

$$L(\hat{n}, \hat{v}) = L_r(\hat{n}, \hat{v}) + L_s(\hat{n}, \hat{v}) + L_p(\hat{n}, \hat{v}) \quad (2.98)$$

avec \hat{n} la normale au point B et \hat{v} la direction d'observation.

L'équation 2.98 peut se réécrire :

$$L(\hat{n}) = RL_r(\hat{r}) + SL_s(\hat{s}) + PL_p(\hat{p}) \quad (2.99)$$

avec R , S et P correspondant aux fonctions de transfert des radiances de réfraction, de réflexion et de réflexion interne. Les formules décrivant ces fonctions de transferts sont les suivantes [Garg 2007] :

$$\begin{aligned} R &= (1 - k(i, \mu))^2 \\ S &= k(i, \mu) \\ P &= k(i, \mu)^N (1 - k(i, \mu))^2 \end{aligned} \quad (2.100)$$

avec k le coefficient de réflexion de Fresnel pour une lumière non-polarisée, i l'angle d'incidence, μ l'indice réfractif de l'eau et N le nombre de réflexions internes.

Ce modèle est valable pour des gouttes sphériques stationnaires.

Notons que la radiance de la goutte est due principalement à la réfraction, sauf sur la périphéries de la gouttes ou les réflexions sont plus importantes. Dans le cas, relativement rare, de présence de soleil et de pluie, la réflexion spéculaire contribue d'une manière significative à la luminance de la goutte. La réflexion interne peut aussi disperser la lumière et former un arc en ciel.

Nous allons maintenant montrer une méthode utilisant le modèle optique afin de détecter les gouttes de pluies et restaurer l'information masquée par les gouttes.

2.3.2 Détection des gouttes de pluie

Dans l'état de l'art, des auteurs ont traité le problème de détection des gouttes par caméra embarquée. [Kurihata 2005] propose une approche fondée sur un apprentissage statique, par analyse en composante principale, des propriétés des gouttes sur une base de données permettant la détection des gouttes. Cette méthode permet principalement la détection sur la zone du ciel. La méthode de [Roser 2009], que nous allons présenter dans la suite, utilise le modèle physique des gouttes afin d'effectuer la détection des gouttes focalisées. Cependant, l'utilisation du modèle physique est complexe à mettre en œuvre. L'auteur [Nashashibi 2010] propose une approche spatio-temporelle permettant la détection des gouttes défocalisées en temps réel. Nous nous sommes intéressé aux gouttes focalisées pour la restauration. La méthode [Cord 2011], que nous allons présenter dans la suite, est celle que nous avons trouvée la plus adaptée à notre problème de gouttes focalisées du fait qu'elle détecte bien les gouttes au niveau de la route et qu'elle est rapide.

2.3.2.1 Méthode s'appuyant sur le modèle physique des gouttes

L'auteur [Roser 2009] a proposé un algorithme de vision monoculaire permettant de détecter les gouttes de pluie. Il s'appuie sur une génération de gouttes virtuelles puis une comparaison entre celle-ci et l'image pour vérifier si une goutte est présente.

Utilisation du modèle optique pour générer des gouttes virtuelles

Considérons une goutte sphérique de rayon r se trouvant à la position $P = (x, y)$ sur le plan image. Nous allons déterminer la partie de la scène qui sera observée à travers cette goutte.

Comme le montre la figure 2.42 un rayon lumineux provenant d'un point de la scène est réfracté par la goutte de pluie avant d'atteindre la caméra en P . Le même point peut être observé directement (sans passer par la goutte) en \hat{P} .

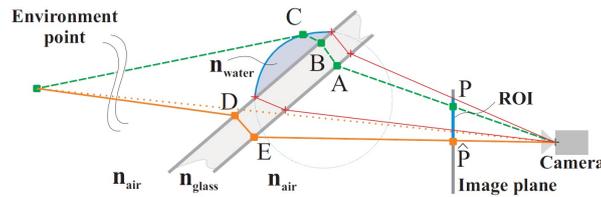


FIGURE 2.42 – Trajets des rayons lumineux en provenance d'un point de la scène. En vert, à travers la goutte et en jaune direct.

En utilisant la Loi de Snell Descartes, nous pouvons déterminer une relation géométrique précise [Roser 2009] (en fonction des angles de réfraction) entre P et \hat{P} . La figure 2.43 montre les échantillons (en vert) des pixels utilisés pour générer la goutte artificielle.

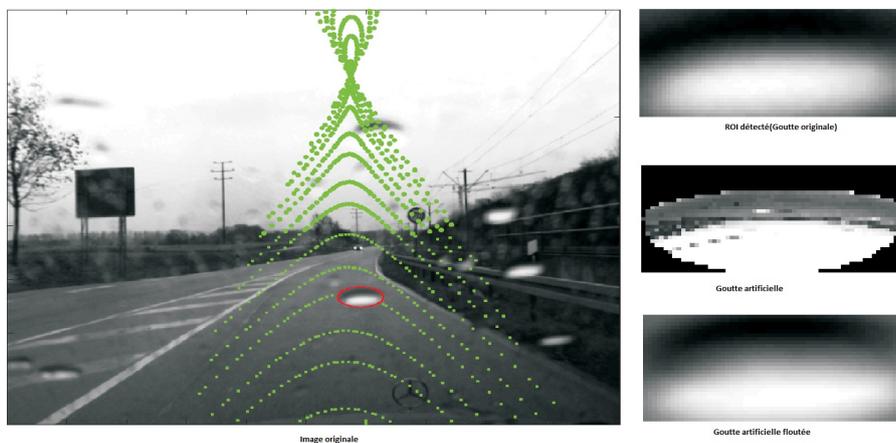


FIGURE 2.43 – À gauche, image originale ; à droite et de haut en bas, goutte originale, goutte artificielle correspondant à la position de la ROI et générée en utilisant le modèle optique à partir des points tracés en vert et goutte artificielle floutée

Comparaison d'une goutte virtuelle avec l'image

Des régions d'intérêts (ROI) de l'image sont extraites en utilisant une méthode de détection des points d'intérêts. Pour chaque ROI, une goutte artificielle de pluie

est générée à partir du reste de l'image. Cette goutte est ensuite comparée à la ROI pour vérifier si une goutte est présente en s'appuyant sur le modèle photométrique. L'intensité du pixel P est estimée comme suit :

$$I_P = \frac{I_{\hat{P}}}{\prod_i (1 - R_i)^j} \quad (2.101)$$

avec R_i le coefficient de réflectivité de Fresnel (voir modèle optique de la pluie) et $j = \pm 1$ correspond à la direction de l'intensité.

La distance des gouttes par rapport à la caméra dépend de leur position sur le pare-brise, ce qui se traduit par une défocalisation (flou) des gouttes. Un flou est donc ajouté à la goutte artificielle en utilisant les paramètres de la caméra et sa position par rapport au pare brise. La figure 2.42 montre le résultat d'ajout du flou.

La génération des gouttes de pluie dans les différentes positions de l'image est coûteuse en temps de calcul. L'auteur propose un algorithme rapide de détection des ROI, nommé RIGSEC [Roser 2009] permettant de limiter la recherche.

Ensuite, la vérification de la détection est effectuée en cherchant le maximum de corrélation des intensités et des gradient et en considérant la goutte artificielle à différentes échelles entre 1 – 1.5.

2.3.2.2 Détection des gouttes de pluie focalisées ayant un fort gradient

L'algorithme [Cord 2011] que nous allons présenter dans cette partie utilise les propriétés photométrique de la pluie et des techniques de traitement d'images afin de détecter et quantifier les gouttes focalisées sur le pare-brise. Pour cela les hypothèses suivantes sont considérées :

- Seules les gouttes présentes sur le pare-brise sont visibles dans l'image ;
- Les gouttes sont similaires à des lentilles déformées ;
- Les gouttes sont quasi-immobiles durant une courte période alors que le reste de la scène est en mouvement.

Notons que les gouttes présentes dans la moitié supérieure de l'image (contenant le ciel) reflète une partie de la route qui est plus sombre que le ciel. Par conséquent, ces gouttes affichent un contraste élevé avec le ciel. De la même façon les gouttes présentes dans la moitié inférieure (contenant la route) affichent un contraste élevé avec la route.

Les étapes de l'algorithme sont les suivantes :

1. Moyenne sur 6 images successives : à cause du caractère stationnaire de la goutte, nous obtenons un meilleur contraste des gouttes par rapport au fond et réduisons en même temps le bruit ;
2. Calcul de la norme du gradient de l'image ;
3. Seuillage à l'aide d'un percentile à 95 % calculé sur une fenêtre de taille 21×21 autour de chaque pixel ;

4. Le gradient de l'image originale est comparé au résultat de seuillage afin de sélectionner les pixels avec un fort gradient (supérieur à 5 %). Une fermeture est appliquée pour regrouper les pixels appartenant à la même goutte ;
5. Reconnaissance de forme : les fausses détections sont éliminées en sélectionnant les régions connexes contenant entre 6 et 160 pixels, le rapport hauteur/largeur doit être compris entre 1/3 et 3 et l'excentricité doit satisfaire $\frac{Perimetre^2}{4\pi \times Air} \leq 1,2$;
6. Une détection est confirmée comme goutte si elle est détectée 6 fois sur 8 images successives (stationnarité de la goutte). Cette étape élimine plusieurs fausses détections ainsi que des vrais détections et donc un compromis doit être fait suivant l'application visée : détection, quantification ou restauration.

La figure 2.44 montre le résultat obtenu par cette algorithme. Nous constatons que la méthode donne un bon résultat pour différentes quantités de pluie et différentes conditions d'illumination de la scène.



FIGURE 2.44 – De gauche à droite : image originale et gouttes détectées.

Comme nous allons le voir dans le chapitre 5, la détection des gouttes va constituer une première étape de l'algorithme de restauration d'images de pluie.

2.4 Conclusion

Dans ce chapitre, après avoir expliqué les phénomènes du bruit et de formation d'images dans le brouillard et la pluie, nous avons présenté un état de l'art de méthodes de débruitage, d'amélioration d'images, de restauration d'images de brouillard et de détection des gouttes de pluie sur le pare-brise par caméra embarquée. Ces méthodes ont servi comme base pour nos travaux de thèse.

Les méthodes de débruitage sélectionnées ont été adaptées à nos problématiques de brouillard et de pluie. Seules les méthodes de restauration d'images de brouillard adaptées aux ADAS ont été sélectionnées. D'autres méthodes de restauration plus générales (n'utilisant pas de modèle physique) ont aussi été testées. Les méthodes sélectionnées ont servi pour la comparaison avec les méthodes que nous avons proposées.

La méthode de détection de pluie est utilisée comme première étape de restauration : reconstitution des parties masquées par les gouttes détectées.

Pour effectuer ce travail, en plus des différents outils de développement (Matlab, C++, OpenCV, RTMaps), comme nous allons le voir dans le chapitre suivant, différentes bases de données ont été créées et différentes méthodes permettant l'évaluation ont été utilisées.

Outils et méthodes d'évaluation des restaurations

Sommaire

3.1 Base de données d'images	61
3.1.1 Images de synthèse	61
3.1.2 Ajout du brouillard sur des images réelles	65
3.2 Méthodes d'évaluation quantitative	66
3.2.1 Utilisation de la vérité terrain	66
3.2.2 Problème de pertes d'informations dans le brouillard	67
3.2.3 Indicateurs fondés sur les contrastes visibles à 5 %	69
3.2.4 Évaluation sur des ADAS	72
3.3 Vidéos ICADAC	72
3.4 Conclusion	72

Dans cette partie, nous allons présenter les outils (logiciels, matériel et méthodes) qui ont servi pour évaluer les différentes méthodes de restauration. Dans un premier temps, nous avons créé une base de données d'images qui a permis d'évaluer et de comparer les méthodes de restauration de brouillard proposées à celles de l'état de l'art. Pour cela, nous avons utilisé un logiciel de simulation de capteurs (proprioceptif, extéroceptif et communication) [Gruyer 2009] nommé SiVIC que nous allons présenter dans ce chapitre.

D'autres bases d'images ont été créées par l'ajout de brouillard sur des images réelles, dont la carte de profondeur était connue par stéréovision.

En plus des indicateurs s'appuyant sur ces bases de données, nous allons présenter d'autres indicateurs de la qualité de la restauration sans utiliser de vérité terrain.

3.1 Base de données d'images

3.1.1 Images de synthèse

Il est difficile, voire impossible, d'obtenir deux images de la même scène routière dans des conditions météorologiques différentes puisque ce type de scènes sont constituées essentiellement d'éléments en mouvement et les conditions d'éclairages varient beaucoup entre deux conditions météorologiques. Nous avons donc opté

pour des images de synthèses (brouillard et pluie), constituant la vérité terrain dédiée à l'évaluation des méthodes proposées, à l'aide du logiciel SiVIC (simulateur véhicule-infrastructure-capteur).

3.1.1.1 Logiciel SiVIC

Il s'agit d'une plate-forme utilisée pour le prototypage de capteurs virtuels. Son objectif est de reproduire de la meilleure façon l'aspect réaliste d'une situation, le comportement d'un véhicule et le fonctionnement des capteurs qui pourraient être embarqués sur ce véhicule.

Pour construire la scène en 3D nous disposons de plusieurs véhicules, différents types de chaussées, bâtiments, panneaux, arbres, ciel... Ensuite un éclairage de la scène est généré (homogène, localisé, ombres) et enfin nous pouvons utiliser plusieurs capteurs (caméras, radar, laser, GPS) dans la même scène et récupérer les informations capturées (images et cartes de profondeur dans notre cas). Enfin les différents paramètres du capteur caméra peuvent être configurés (focale, taille du capteur, taille des pixels, repère). SiVIC fonctionne soit en mode glisser-déposer soit à l'aide de langage script. Dans notre cas nous avons utilisé le deuxième choix pour plus de précision et afin d'utiliser les événements qui offrent la possibilité de générer plusieurs configurations automatiquement et donc de manière rapide. Le grand intérêt d'utiliser SiVIC est de permettre de simuler des situations difficiles à reproduire en réel. Par exemple, nous citerons le parcours de plusieurs kilomètres en situations dangereuses ou la récupération d'une carte de profondeur en plus des images enregistrées.



FIGURE 3.1 – Exemples d'images SiVIC

L'image 3.1 montre des exemples d'images générées par SiVIC : nous avons différents types d'horizons, de chaussées, d'emplacements caméra. SiVIC utilise aussi plusieurs filtres pour simuler différents phénomènes (brouillard, pluie, floue optique...).

Dans le cas du brouillard, SiVIC ne permet de créer qu'un brouillard de type uniforme (coefficient d'extinction atmosphérique et luminance du ciel constants). Nous avons donc créé des images sans brouillard avec SiVIC et les différents types de brouillard à l'aide de MATLAB en utilisant le modèle de Koschmieder et les cartes de profondeurs denses correspondantes aux images et fournies par SiVIC. La figure 3.2 montre des images avec leur carte de profondeur.

Le filtre pluie de SiVIC nous a permis de simuler la présence des gouttes (de différentes tailles et formes) sur le pare-brise. Nous avons ensuite créé un masque des gouttes pour permettre l'évaluation de méthodes de restauration.

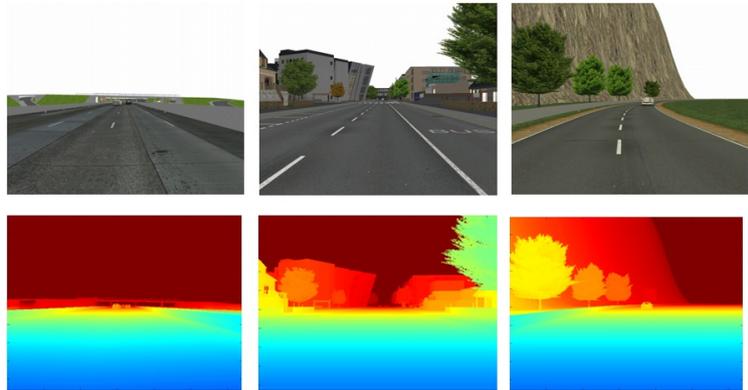


FIGURE 3.2 – Images et cartes de profondeurs générées par SiVIC

3.1.1.2 Bases de données d'images routières de brouillard

Dans le cadre d'un article [Tarel 2010] destiné à évaluer et comparer plusieurs méthodes de restauration d'images de brouillard, une base de données d'images de synthèses contenant 18 images de différents types de scènes : urbaine, autoroute et rurale a été créée. Cette base a été complétée pour contenir 66 images différentes [Halmaoui 2011, Tarel 2012] et porte le nom de FRIDA (Foggy Road Images DAtabase).

Rappelons que, dans le cadre de cette thèse, nous sommes limités à l'utilisation d'une seule caméra, mais comme la base FRIDA a été mise en ligne pour servir la communauté scientifique, les images ont été créées en utilisant deux caméras afin de pouvoir évaluer d'autres méthodes utilisant la stéréovision.

Pour faciliter l'ajout du brouillard dans les images de synthèse, le ciel a été créé en blanc (intensité de 255 sur les trois canaux). À l'aide des cartes de profondeur fournies par SiVIC, nous avons créé 4 types de brouillard en faisant varier la densité du brouillard β et/ou la luminance du ciel I_∞ .

- *Brouillard uniforme* : l'équation de Koschmieder est appliquée sur les 66 images en choisissant un $\beta \simeq 0,035$ correspondant à une distance de visibilité de 85 m d'après la formule $Vis = -\frac{\ln(0,05)}{\beta}$ et une intensité du ciel fixée à 255.
- *Brouillard de densité β hétérogène* : dans la réalité le brouillard n'est pas tout le temps uniforme. Nous avons donc introduit une variabilité dans l'équation de Koschmieder en fixant I_∞ et en pondérant β en fonction de la position du pixel. Les poids spatiaux sont obtenus à partir d'un bruit de Perlin qui est créée à partir d'une somme de fonctions de bruit générées à différentes échelles avec un poids $\log_2(s)$ à l'échelle s . La distance de visibilité moyenne correspond à 80 m.
- *Brouillard avec une luminance du ciel I_∞ hétérogène* : la variable β est fixée telle que la distance de visibilité moyenne est de 80 m. un bruit de Perlin nous sert pour créer un brouillard tel que I_∞ est variable.

- *Brouillard mixte* : grâce à l'utilisation de deux bruits de Perlin indépendants nous avons créé un brouillard hétérogène où nous faisons varier à la fois β et I_∞ . La moyenne de β est choisie telle que la distance de visibilité est moyenne soit de 80 m.

La base de données contient donc 66×4 images en plus des images originales (330 images au total). La figure 3.3 montre une même scène avec différents brouillards.

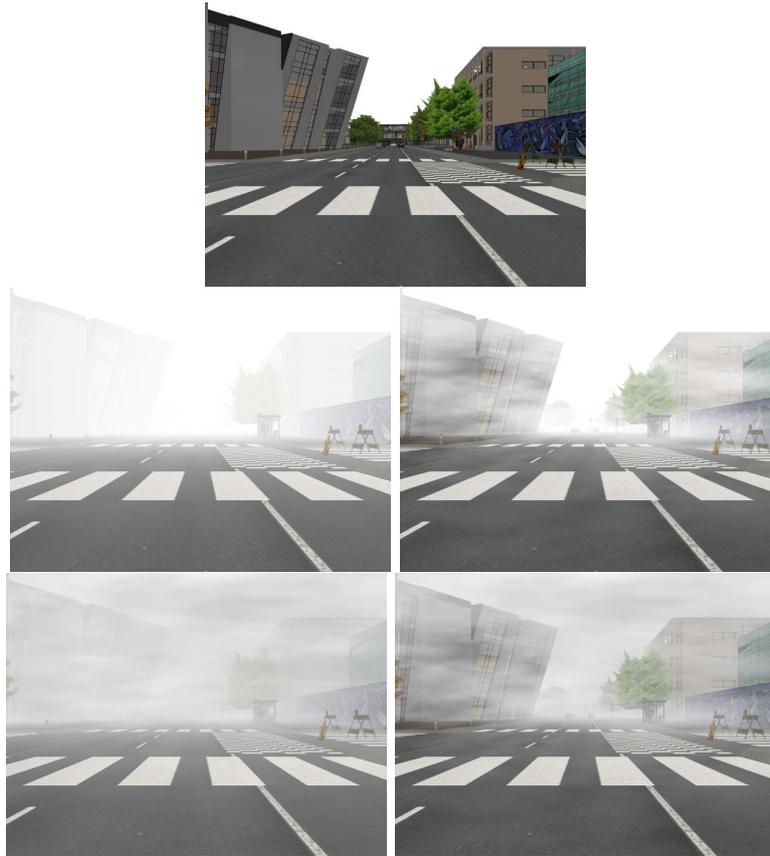


FIGURE 3.3 – Image de synthèse originale et les différents types de brouillards : uniforme, β variable, I_∞ variable et β et I_∞ variables

Cette base de données a été utilisée dans le cadre d'un autre article [Halmaoui 2011] pour évaluer une nouvelle méthode de restauration et comparer les résultats avec d'autres méthodes.

3.1.1.3 Images de pluie

Comme nous l'avons déjà précisé, SiVIC possède un filtre pluie qui permet de générer des gouttes de pluie de différentes tailles et formes et en différentes quantités. L'image 3.4 montre des images créées pour simuler différentes situations de pluies.

Nous avons utilisé ce filtre pour créer des images avec et sans pluie afin d'évaluer les méthodes de restauration d'images de pluie. Pour cela nous avons besoin de



FIGURE 3.4 – Différentes gouttes de pluie

connaître la position des gouttes.

Dans SiVIC nous n'avons pas la possibilité de récupérer la position exacte des gouttes. Comme les gouttes sont simplement un filtre appliqué à l'image, pour chaque scène capturée nous avons créé une autre image sans le filtre et en faisant la différence entre les deux nous obtenons un masque sur les gouttes. La figure 3.5 montre l'image originale, l'image avec pluie et le masque des gouttes.



FIGURE 3.5 – Images originale, avec des gouttes de pluie simulée et masque des gouttes

3.1.2 Ajout du brouillard sur des images réelles

Une autre base de données d'images réelles a été utilisée pour permettre l'évaluation et afin de comparer les résultats avec ceux obtenus sur les images de synthèses [ITS].

Pour cela nous avons utilisé des images stéréo d'une base de donnée "Karlsruhe" [Geiger 2010, Geiger 2011b, Tarel 2012] afin de récupérer une carte de disparité à l'aide de l'algorithme de reconstruction stéréo Libelas [Geiger 2011a]. Ensuite un filtre bilatéral guidé par l'image originale a permis de combler les trous dans cette carte. Une carte de profondeur est récupérée à l'aide d'un calibrage de la caméra.

Cette base de données contient 10 images originales (en stéréo). En appliquant la même procédure que pour les images de synthèses nous avons ajouté 4 types de brouillards. La figure 3.6 montre une image de la base "Karlsruhe" et les 4 types de brouillards correspondants.

Dans le cadre d'une autre étude portant sur la détection de l'espace navigable [Hautière 2011] nous avons utilisé une autre base d'images monoculaires contenant 15 images sur lesquelles nous avons détecté manuellement (à l'aide de Matlab) les



FIGURE 3.6 – Image originale en haut. De gauche à droite et de haut en bas, les différents types de brouillards ajoutés : brouillard uniforme, β variable, I_∞ variable, β et I_∞ variables

régions correspondantes à la route. La figure 3.7 montre une image de cette base de données et le masque de la route.



FIGURE 3.7 – Image avec brouillard et masque de la route

3.2 Méthodes d'évaluation quantitative

3.2.1 Utilisation de la vérité terrain

Afin d'évaluer les performances d'une méthode de restauration, une façon simple est de comparer l'image restaurée à la vérité terrain (image originale sans brouillard) en calculant la moyenne des différences absolues des intensités de pixels des deux images. Nous ne prenons pas en compte le ciel afin d'éviter de biaiser le résultat. En effet, les pixels du ciel restent inchangés lors de la restauration et ont un score

nul. Toutefois, le fait de se comparer avec l'image originale pose un problème que nous allons expliquer dans ce qui suit.

3.2.2 Problème de pertes d'informations dans le brouillard

La qualité de l'image restaurée va dépendre du codage des pixels de l'image avec brouillard. Dans notre cas nous avons utilisé des images codées sur 8 bits par pixel et par canal couleur.

Pour un objet ayant, en absence de brouillard, une intensité I_0 , pour un certain β et à partir d'une certaine distance d , le modèle de Koschmieder nous montre que l'intensité apparente I de l'objet commence à être saturée (tend vers l'intensité du ciel I_∞). La figure 3.8 montre la courbe de I en fonction de d pour $I_0 = 0$ et $\beta = 0,0375$. Notons que plus l'intensité I_0 est grande plus la saturation est atteinte rapidement.

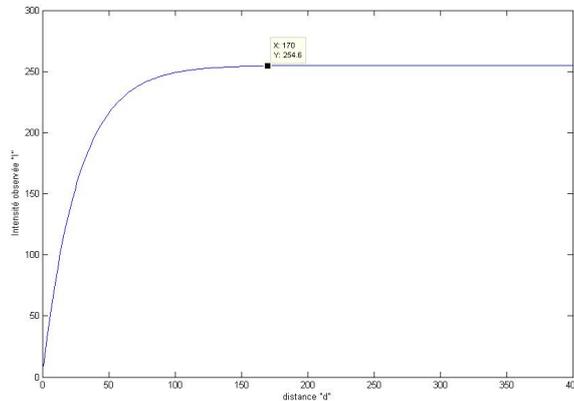


FIGURE 3.8 – Saturation de l'intensité d'un objet noir en fonction de la distance dans le cas d'un brouillard de densité $\beta = 0.0375$

Comme les pixels de nos images de brouillard I sont codés en entiers naturels sur 8 bits nous allons donc avoir une perte d'information (due à l'échantillonnage) par rapport à l'image originale I_0 initialement créée par SiVIC. Cette perte sera de plus en plus importante lorsque I commence à atteindre la saturation. Par exemple nous remarquons sur la figure 3.8 que l'intensité I est de 254,6 lorsque l'objet se trouve à une distance de 170m, cette valeur de I correspondra à 255 dans le cas d'images 8 bits et il y a donc une absence totale d'information dans I à partir de cette distance (l'objet se confond avec le ciel $I_\infty = 255$). En utilisant donc un codage avec plus de poids (16 bits par exemple) nous allons pouvoir garder plus d'informations sur les objets les plus distants.

La figure 3.9 montre l'image originale, l'image après l'ajout de brouillard et les images restaurées idéales obtenues en inversant Koschmieder à l'aide des mêmes paramètres utilisés pour l'ajout du brouillard (connaissance parfaite du modèle), pour les cas d'images I codées sur 8 bits et 16 bits. Nous remarquons que les deux

images restaurées contiennent moins d'informations (sur les objets lointains) que l'image originale et que l'image 16 bits contient plus d'informations que l'image 8 bits.



FIGURE 3.9 – Image originale, image avec brouillard et restauration idéale dans le cas d'images 8 bits et 16 bits

De même, les méthodes de restauration ne pourront pas restaurer cette information puisqu'elle est inexistante. Donc, au lieu d'évaluer les méthodes en comparant le résultat à l'image originale, nous avons choisi d'effectuer la comparaison du résultat avec l'image restaurée idéale. La figure 3.10 montre un schéma illustrant la procédure d'évaluation.

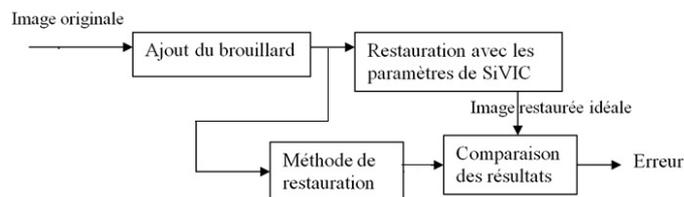


FIGURE 3.10 – Schéma d'évaluation d'une méthode de restauration.

Pour résumer, en plus d'avoir une perte de visibilité dans le brouillard, nous avons une absence totale d'information à partir d'une certaine distance qui dépend de la profondeur des pixels (en bits) de l'image capturée.

Nous tenons à préciser que ce problème existe aussi dans le cas d'images réelles

de brouillard et la quantité d'information capturée va dépendre de l'optique et de l'électronique de la caméra.

3.2.3 Indicateurs fondés sur les contrastes visibles à 5 %

Une autre façon pour évaluer quantitativement et permettre la comparaison de méthodes de restauration est présentée dans cette partie. Elle consiste à calculer des indicateurs s'appuyant uniquement sur l'image originale et l'image restaurée. Elle se fonde sur une extraction des contours visibles à 5 %.

Extraction des contours visibles

La méthode proposée par [Hautière 2008] permet de détecter les contours visibles dans une image (un minimum de 5 % de contraste).

Deux pixels $I(x)$ et $I(y)$ sont dits séparés par un seuil s si :

- $y \in V_4(x)$ avec $V_4(x)$ les 4-voisins de x ;
- $\min(I(x), I(y)) < s < \max(I(x), I(y))$

Une façon rapide de calculer le minimum et le maximum entre deux pixels appartenant à un voisinage V_4 est d'appliquer des dilatations et des érosions en utilisant les éléments structurants correspondant à un voisinage V_4 :

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Nous commençons donc par parcourir notre image à l'aide d'une fenêtre de taille 7×7 ou 9×9 et on teste si deux pixels sont séparés par un seuil s en faisant varier s du $\min(I(x), I(y))$ au $\max(I(x), I(y))$. Lorsque c'est le cas, nous calculons le contraste de Weber correspondant, suivant l'équation 3.1 et nous l'étiquetons par son seuil s . Ensuite nous gardons le seuil s_{max} qui nous donne le maximum de contraste et attribuons ce contraste aux pixels séparés par s .

$$C(x, y, s) = \min \left(\frac{|s - I(x)|}{\max(s, I(x))}, \frac{|s - I(y)|}{\max(s, I(y))} \right) \quad (3.1)$$

Gain en niveau de visibilité

Nous calculons les gradients de l'image restaurée et ceux de l'image originale (gradient de Sobel par exemple) et calculons le rapport r_i entre ces deux gradients mais seulement pour les pixels ayant un contraste supérieur à 5 % (dans chaque image). Nous obtenons donc un indicateur sur le niveau d'amélioration de la visibilité qui pourra être utilisé pour comparer deux méthodes de restauration.

Indicateur e : Taux des nouveaux contours visibles

Soit n_r et n_o les nombres de pixels ayant un contraste supérieur à 5 %, respectivement dans l'image restaurée et dans l'image originale. La mesure de taux des

nouveaux contours visibles e (équation 3.2) nous indique la capacité de la méthode à restaurer des nouveaux contours.

$$e = \frac{n_r - n_o}{n_o} \quad (3.2)$$

Indicateur \bar{r} : Qualité de la restauration

Nous calculons la moyenne géométrique des ratios des gains de visibilité suivant l'équation 3.3.

$$\bar{r} = \exp\left(\frac{1}{n_r} \sum_{p \in P_r} \log(r_i)\right) \quad (3.3)$$

avec P_r l'ensemble des contours visibles et r_i le rapport des gradients entre l'image restaurée et l'image originale.

Indicateur σ : Taux de pixels saturés

Nous pouvons aussi évaluer la qualité de la restauration en détectant les pixels qui deviennent saturés après la restauration en calculant un indicateur σ selon l'équation 3.4.

$$\sigma = \frac{n_s}{n_l n_c} \quad (3.4)$$

avec n_s le nombre de pixels saturées et n_l et n_c le nombre des lignes et des colonnes de l'image.

La figure 3.11 montre le résultat obtenue par cette méthode sur une image de brouillard et la restaurée correspondante en utilisant la méthode du monde plan. Les indicateurs déduit montre que nous obtenons, après restauration, plus de contours visible à 5 %, que la qualité de la restauration est améliorée et que aucun pixel n'est saturé après restauration.

Conclusion

Le défaut de ces indicateurs est qu'ils supposent que les contours ne correspondent pas à du bruit. Certaines méthode de restauration améliorent la visibilité mais amplifient le bruit présent dans l'image et il sera considéré comme des contours. C'est pour cette raison que nous avons choisi d'utiliser principalement d'autres méthodes d'évaluation : erreur absolue moyenne entre l'image restaurée et l'image restaurée idéale. Pour cela, nous sommes obligé d'utiliser des images de synthèse ou des images réelles avec du brouillard synthétique. Donc, c'est pour cette raison que les indicateurs présentés dans cette section ont été utilisés : évaluation quantitative de la restauration d'images réelles. Une autre piste permettant une évaluation quantitative est la combinaison de la restauration avec des ADAS.

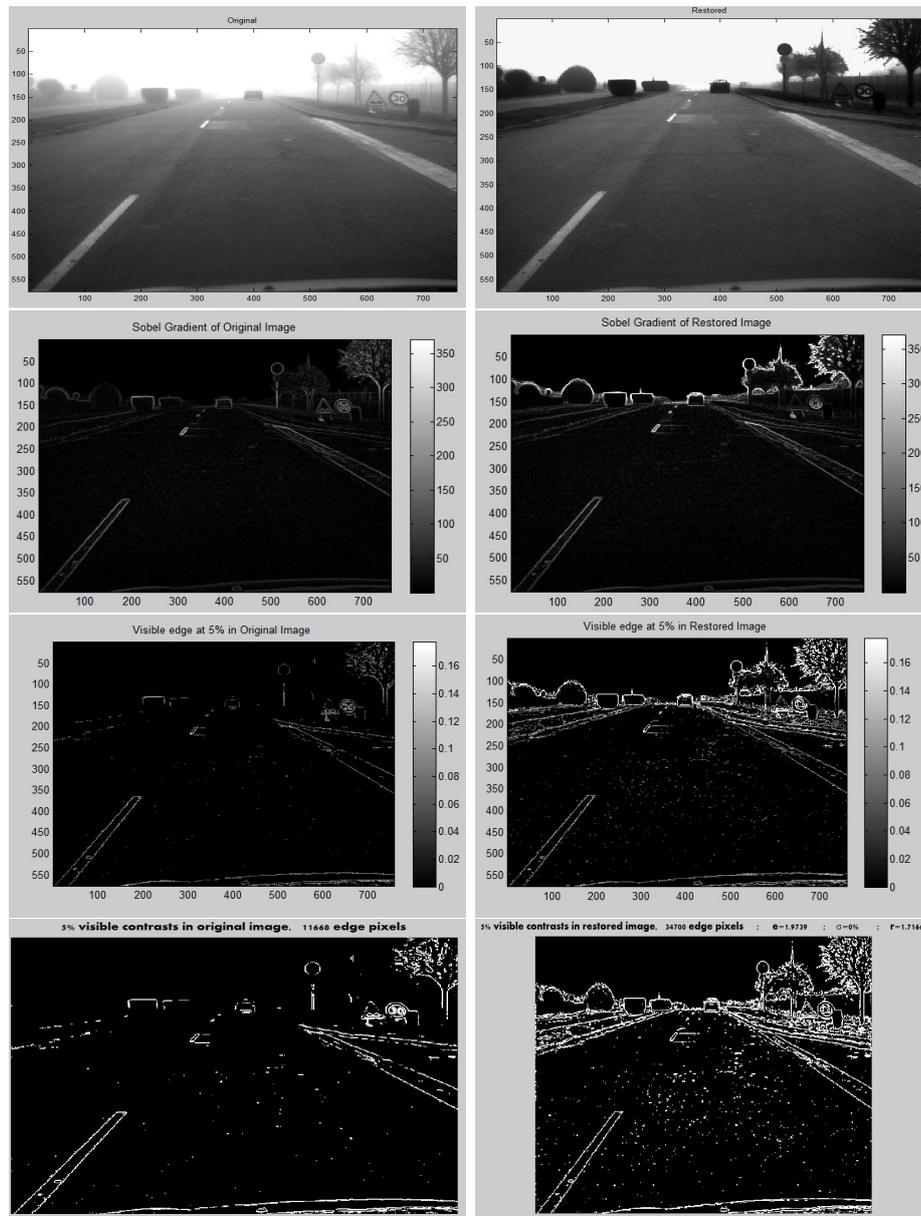


FIGURE 3.11 – Exemple d'évaluation de la restauration par la méthode [Hautière 2007] : utilisation de la méthode d'extraction des contours visibles à 5 %. Image originale et image restaurée (première ligne), gradients de Sobel correspondants (deuxième ligne), contours visibles avec un contraste de 5 % (troisième ligne) et indicateur sur la qualité de restauration (quatrième ligne). Les indicateurs montrent que nous obtenons plus de contrastes visible à 5 % après restauration $e = 1,97$, que nous avons un taux de pixels saturés nul $\sigma = 0\%$ et que la qualité de la restauration est améliorée ($r = 1,7$).

3.2.4 Évaluation sur des ADAS

Le but de notre étude étant l'amélioration des systèmes de vision d'aides à la conduite par temps de brouillard et de pluie, la meilleure façon d'évaluer les méthodes de restauration est de les utiliser comme pré-traitement ou traitement ADAS. Nous allons voir plus en détails dans le chapitre 6, l'évaluation d'un algorithme de détection de panneaux par temps clair, par temps de brouillard et en utilisant la restauration. Dans ce chapitre, nous avons aussi évalué la détection de l'espace navigable en utilisant une version améliorée de la méthode [Hautière 2007] que nous avons présentée précédemment dans le chapitre 2.

3.3 Vidéos ICADAC

Afin d'évaluer de manière qualitative les méthodes de restauration dans des situations de conduite réelles, plusieurs vidéos ont été enregistrées en présence de brouillard à l'aide d'une caméra embarquée dans un véhicule. Des caméras couleurs et en niveaux de gris ont été utilisées. Les scènes filmées sont très variées : milieu urbain, autoroute, crépuscule, milieu de la journée, brouillard dense ou moyen. . .

La voiture utilisée est équipée d'autres capteurs. Pour notre étude nous avons récupéré l'information capturée par un lidar spécifique qui nous fournit la distance de visibilité.

La figure 3.12 montre 3 images extraites d'une vidéo de brouillard diurne, par caméra embarquée ICADAC.



FIGURE 3.12 – Images de brouillard extraites d'une vidéo ICADAC.

3.4 Conclusion

Pour l'évaluation des restaurations d'images dégradées par le brouillard, nous pouvons utiliser, du qualitatif au quantitatif :

- des images de brouillard extraites de vidéos de différentes scènes, acquises par caméra embarquée ICADAC ;
- la méthode des contrastes à 5 % qui nous donne des indicateurs sur la qualité de la restauration ;
- les images de synthèse FRIDA [Tarel 2010, Halmaoui 2011] et les images réelles "Karlsruhe" avec brouillard synthétique [Tarel 2012].

Notons que la base d'images FRIDA est mise à la disposition de la communauté scientifique.

Pour évaluer d'une manière qualitative la restauration d'images de pluie, quelques images de synthèse ont été créées. Pour l'évaluation quantitative, nous utiliserons des images réelles extraites d'une vidéo de pluie acquise par caméra embarquée.

L'utilisation de ces différentes bases de données nous a permis de concevoir et de valider des algorithmes de restauration d'images de brouillard et de pluie, que nous allons présenter dans les deux chapitres suivants.

Restauration d'images de brouillard

Sommaire

4.1 Approche proposée	75
4.1.1 Algorithme	76
4.1.2 Optimisation des paramètres : Recuit simulé	85
4.1.3 Évaluation	89
4.2 Amélioration du rendu visuel	94
4.2.1 Accroissement de la dynamique	94
4.2.2 Lissage de la carte de transmittance par un filtre bilatéral	97
4.2.3 Couleur	100
4.3 Conclusion	100

Dans cette partie, nous allons présenter, en détail, la méthode proposée permettant d'effectuer la restauration d'images acquises par temps de brouillard. Elle sera ensuite évaluée qualitativement et quantitativement. Puis, quelques améliorations du rendu visuel seront proposées.

4.1 Approche proposée

Le problème de restauration peut se résumer à trouver l'image R en inversant le modèle de Koschmieder :

$$R = Ie^{\beta d} + I_{\infty}(1 - e^{\beta d}) \quad (4.1)$$

Le problème est que la profondeur des objets, la densité du brouillard et l'intensité du ciel sont inconnues. Dans l'équation 4.1 nous n'avons accès qu'à l'image I observée.

Comme nous l'avons vu dans le chapitre 2, suivant le coefficient d'extinction β et la distance des objets de la scène, les effets du voile atmosphérique et de l'atténuation vont dégrader d'une manière plus ou moins importante les contrastes et les couleurs des objets observés. Nous avons proposé dans cette thèse une nouvelle méthode permettant d'améliorer la qualité des images de brouillard en récupérant les contrastes et les couleurs des images tant que ces informations sont toujours disponibles dans l'image (voir section perte de l'information 3.2.2). La méthode

proposée combine une approche physique fondée sur le modèle optique du brouillard et une approche signal fondée sur l'égalisation locale de l'histogramme des images.

Nous avons ensuite proposé des améliorations de la méthode afin d'obtenir des meilleurs résultats visuels en utilisant toute la dynamique de l'image, un lissage robuste au bruit ou les propriétés des espaces couleurs.

L'application d'un algorithme d'optimisation, le recuit simulé, nous a permis d'optimiser les différents paramètres de la méthode en utilisant notre base d'images FRIDA (voir 3).

Nous avons ensuite évalué cette méthode, d'abord de manière subjective (visuellement), ensuite d'une manière objective avec notre base d'images FRIDA. Une autre façon d'effectuer l'évaluation est de comparer les performances d'un ADAS par vision avec et sans restauration dans le brouillard ou de comparer les performances de perception du conducteur dans le cas d'utilisation d'un "afficheur tête haute" affichant l'image restaurée au conducteur comme nous allons le voir dans le chapitre 6.

Enfin nous proposons un schéma d'implantation modulaire constitué de plusieurs étapes qui sont indépendantes les unes des autres et qui peuvent être effectuées en utilisant d'autres méthodes accomplissant la même tâche.

Dans la suite nous allons présenter notre méthode de restauration des images de brouillard.

4.1.1 Algorithme

Le schéma sur la figure 4.1 montre les différentes étapes de la méthode. Après avoir caractérisé le brouillard, nous commençons par estimer une approximation grossière de l'image sans brouillard en utilisant une approche signal dans laquelle nous avons introduit une contrainte déduite du modèle optique du brouillard. L'image sera ensuite améliorée à l'aide d'un lissage adapté en utilisant des contraintes sur la scène de brouillard.

4.1.1.1 Densité du brouillard

Pour caractériser, le brouillard nous avons besoin d'estimer la densité du brouillard β . Pour cela, nous avons utilisé deux procédés. D'une part un Lidar dédié, d'autre part l'approche utilisée par [Hautière 2007] et que nous avons présentée dans le chapitre 2.

4.1.1.2 Intensité du ciel

Afin d'estimer l'intensité du ciel, la façon la plus simple de procéder est d'utiliser l'hypothèse que l'intensité du ciel correspond à l'intensité maximale dans l'image. Cependant, dans certains cas, la scène contient des objets réfléchissants et d'autres sources de lumières (feux des voitures ou de l'infrastructure) et qui peuvent avoir des intensités plus grandes que celle du ciel. Pour surmonter ce problème, nous pouvons considérer que les objets lumineux ou brillants ayant une intensité supérieure à celle

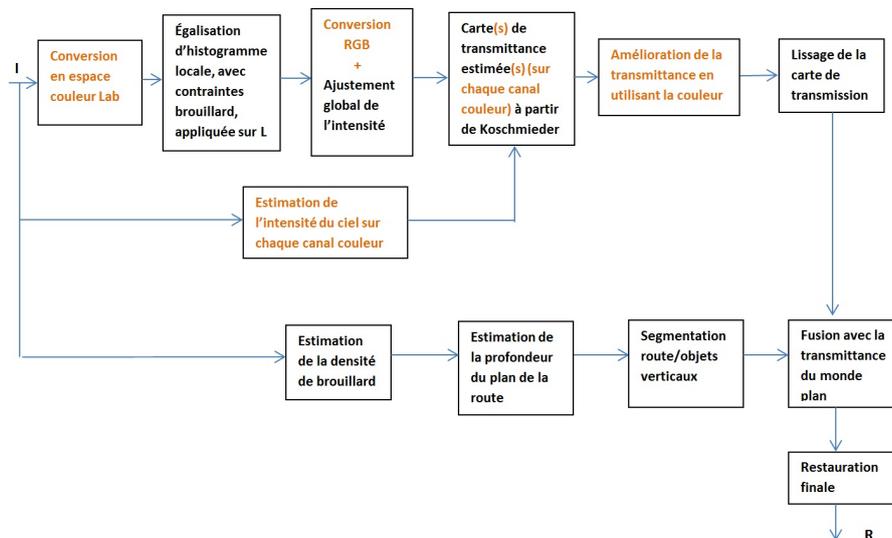


FIGURE 4.1 – Différentes étapes de la restauration. Les étapes en couleur orange ne s’appliquent que dans le cas d’images couleur. Les deux étages du haut correspondent à la méthode de restauration proposée : restauration du contraste grâce à l’égalisation d’histogramme ensuite amélioration de la couleur et lissage. Les deux étages du bas correspondent à la caractérisation du brouillard en estimant sa densité, ensuite, segmentation de la route et enfin, fusion avec notre méthode pour la restauration des objets verticaux.

du ciel ne couvrent pas une grande zone de l’image. Nous pouvons estimer l’intensité I_∞ à l’aide d’un percentile à 95 % par exemple, sauf que dans les 5 % restants le modèle de Koschmieder n’est plus valable, et leur restauration va introduire des artefacts. La figure 4.2 montre le résultat obtenu avec cette méthode. Dans



FIGURE 4.2 – Image originale, restauration en estimant l’intensité du ciel comme le maximum sur toute l’image et en utilisant le percentile à 95 %

certain cas, l’intensité du ciel n’est pas parfaitement blanche. Après restauration des effets indésirable peuvent apparaître dans l’image restaurée : saturation, couleur irréalistes. Pour surmonter ce problème, une balance des blancs est classiquement appliquée, reposant sur une méthode telle que celles présentées dans le chapitre 2.

Cependant, une telle méthode ne se combine pas toujours bien avec les algorithmes de restauration. La figure 4.3 montre le résultat obtenu en utilisant la méthode “Monde Gris” présentée dans le chapitre 2, nous remarquons que la balance

des blancs améliore l'image avec brouillard, en revanche dans l'image restaurée le ciel garde un aspect bleuâtre. Une autre piste explorée consiste à modifier les his-



FIGURE 4.3 – Image originale, à gauche ; Image après balance des blancs, au centre ; Image restaurée en estimant l'intensité du ciel comme le maximum sur toute l'image, à droite

togrammes de chaque canal couleur pour obtenir un ciel de couleur blanche. Cette approche consiste à détecter les pixels du ciel et à ensuite modifier les histogrammes. Elle permet d'obtenir directement l'intensité du ciel. Le résultat d'un tel ajustement est présenté sur la figure 4.4. Nous constatons que nous avons toujours le problème du ciel qui devient bleu après restauration car la balance n'est pas parfaite (le ciel n'est pas parfaitement blanc). En revanche, nous avons une amélioration des contrastes par rapport aux résultats précédents. Le problème avec cette méthode est que nous dégradons certaines couleurs de l'image originale en décalant les histogrammes des 3 canaux. L'utilisation d'une balance des blancs avec notre méthode de



FIGURE 4.4 – Image originale, à gauche ; Image après balance des blancs par modification d'histogramme, au centre ; Image restaurée en estimant l'intensité du ciel comme le maximum sur toute l'image, à droite

restauration n'est pas très efficace et nous devons donc avoir une bonne estimation de l'intensité du ciel afin d'éviter les problèmes de couleurs irréalistes. Aussi dans la suite, nous choisirons l'intensité du ciel en utilisant le maximum dans l'image. Il est tout à fait envisageable, compte tenu du caractère modulaire de l'algorithme de remplacer cette estimation par une méthode plus fine.

4.1.1.3 Restauration locale du contraste

Une égalisation d'histogramme effectuée de manière locale est utilisée pour obtenir une première approximation de l'image sans brouillard. Cette étape permet d'améliorer le contraste sur toute l'image et donc de réduire l'effet d'atténuation par le brouillard de l'intensité et des couleurs des objets de la scène.

En effet, comme le brouillard a pour effet de décaler l'histogramme des pixels vers le maximum d'intensité (le ciel) et donc de réduire le contraste entre ces pixels, une égalisation d'histogramme, comme nous l'avons vu dans le chapitre 2, distribue les intensités de l'image sur toute la dynamique de l'histogramme. Elle va ainsi permettre d'augmenter le contraste de l'image. Comme dans le brouillard le niveau de dégradation du contraste est fonction de la distance, une égalisation d'histogramme globale ne donne pas de bons résultats. Nous avons donc choisi d'effectuer la restauration localement.

L'image est partitionnée en plusieurs blocs de petites tailles W_s (par exemple $W_s = 3$ pour récupérer les détails les plus fins) et le principe d'égalisation est appliqué sur chaque bloc séparément.

Pour chaque bloc, nous prenons en compte les 8 blocs voisins pour le calcul de la fonction de distribution d'intensités normalisée et seul le bloc central sera modifié. Ceci a pour effet de réduire la différence d'intensité entre blocs voisins dans l'image restaurée. Cette opération est donc effectuée sur 81 pixels dans le cas où $W_s = 3$ et seuls les 9 pixels centraux seront modifiés (voir figure 4.5). Nous appliquons la

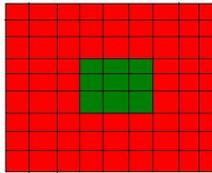


FIGURE 4.5 – Bloc central à modifier en vert et les blocs voisins en rouge

contrainte additionnelle $I_0 < I$ qui résulte du modèle de Koschmieder. En effet, en considérant que l'intensité du ciel est la plus grande intensité de la scène, à partir du modèle de Koschmieder nous déduisons que l'intensité d'un pixel dans le brouillard sera toujours plus grande que son intensité sans brouillard. Nous appliquons donc l'égalisation entre 0 et le maximum d'intensité du bloc central que nous voulons modifier.

Lorsque la région à traiter est uniforme, l'égalisation ne modifie pas les intensités des pixels du bloc. Notons que de telles régions ne nous intéressent pas puisqu'elles ne contiennent pas d'information. Dans le cas contraire, lorsque la région contient des contours, l'égalisation dépend des blocs voisins : deux zones identiques ayant des blocs voisins différents ne seront pas restaurés de la même façon à cause de la forme de leurs fonctions de distribution d'intensité qui vont être différentes, ce qui génère des effets blocs dans l'image résultante.

Dans le cas d'images couleurs, l'image est convertie dans l'espace *Lab* et l'égalisation est effectuée sur la composante luminance.

La figure 4.6 montre l'image avant et après égalisation. Le résultat montre que nous obtenons une meilleure visibilité pour les différents objets de la scène et l'effet d'atténuation dû au brouillard a bien été réduit.

La figure 4.7 montre le résultat d'égalisation avec la méthode CLAHE classique

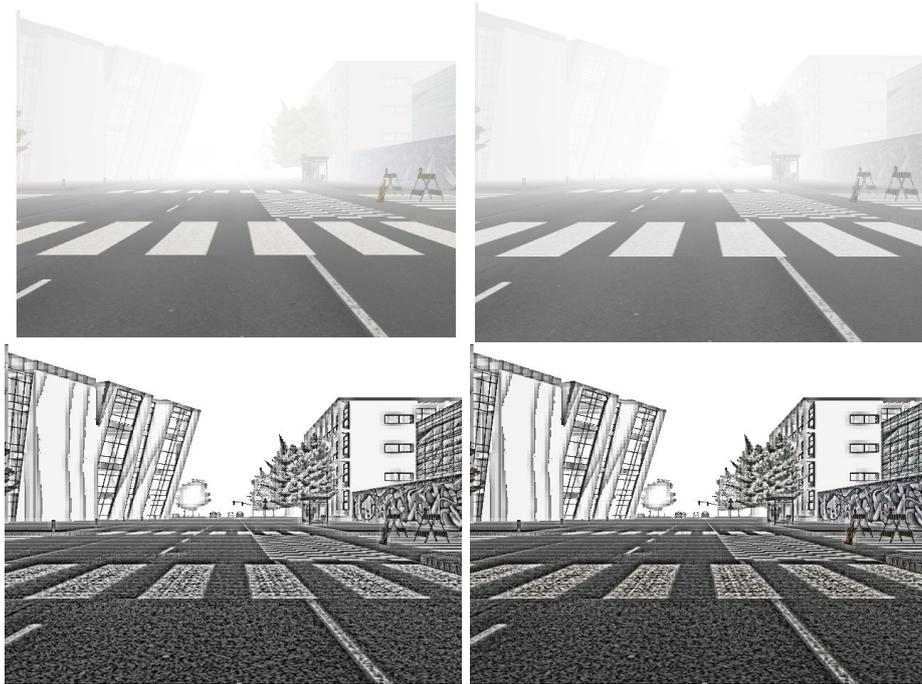


FIGURE 4.6 – Rehaussement du contraste par notre méthode “CLAHE avec contraintes brouillard” : En haut à gauche, l’image originale ; en haut à droite, la composante de luminance ; en bas à gauche, la restauration du contraste de la composante L ; en bas à droite, la conversion de l’image restaurée en couleur

(sans utiliser les contraintes brouillard) en utilisant une fenêtre de taille 3. Nous constatons qu’avec notre méthode, en introduisant des contraintes physiques sur la scène de brouillard, nous améliorons tout aussi bien le contraste des différents objets de la scène mais en réduisant le bruit généré sur la route, par rapport à la méthode classique. Dans la section suivante, nous allons montrer comment les

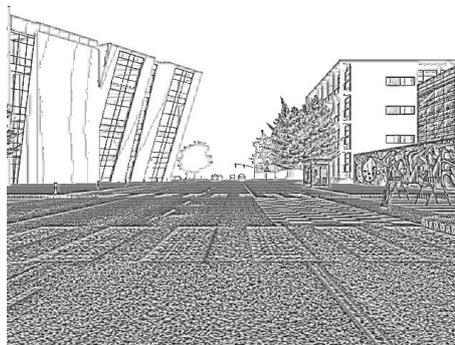


FIGURE 4.7 – Rehaussement du contraste avec “CLAHE classique” sur l’image de la figure 4.6

effets indésirables de l’égalisation seront réduits afin d’obtenir une image plus lisse

et comment les couleurs seront améliorées.

4.1.1.4 Ajustement global de l'intensité

Comme chaque bloc égalisé garde son intensité maximale inchangée, certains pixels vont toujours contenir une part de brouillard. Nous avons donc choisi d'effectuer un changement global de l'intensité de l'image afin de se rapprocher de l'image sans brouillard. Nous appliquons donc une transformation affine $aI_0 + b$ à l'image après égalisation telle que $a < 1$ et $b > 0$. Les valeurs optimales de a et b seront choisies à l'aide d'un algorithme de recuit simulé en utilisant notre base de données FRIDA comme nous allons le voir dans la suite. La figure 4.8 montre le résultat obtenu après ajustement d'intensité avec $a = 0,3$ et $b = 50$.



FIGURE 4.8 – Ajustement affine d'intensité

4.1.1.5 Carte de transmittance

Comme nous l'avons expliqué dans le chapitre 2, la carte de profondeur est difficile à estimer de façon précise. Nous allons donc regrouper les deux variables β et d dans une seule variable : la transmittance $t = e^{-\beta d}$.

Nous déduisons t en faisant l'hypothèse que la scène sans brouillard possède un meilleur contraste que celle avec brouillard. Nous supposons donc que l'image obtenue après égalisation d'histogramme locale et ajustement global d'intensité est une approximation de l'image I_0 .

À partir du modèle de Koschmieder, nous déduisons l'expression de la transmittance t en fonction de I et des inconnues I_0 et I_∞ :

$$t = \frac{I - I_\infty}{I_0 - I_\infty} \quad (4.2)$$

4.1.1.6 Exploitation de l'information couleur pour améliorer la carte de transmittance

Dans le cas d'images couleurs, nous convertissons l'image d'entrée dans un espace couleur tel que Lab . Nous appliquons l'égalisation d'histogramme locale sur la

composante luminance et récupérons les couleurs d'origine à partir des composantes chromatiques de l'espace *Lab*. Ensuite nous appliquons l'équation 4.2 sur chaque canal séparément et nous choisissons la transmittance finale comme le minimum des trois transmittances de chaque canal : $t = \min(t_R, t_G, t_B)$.

La figure 4.9 montre la carte de transmittance déduite et l'image restaurée avec cette carte.

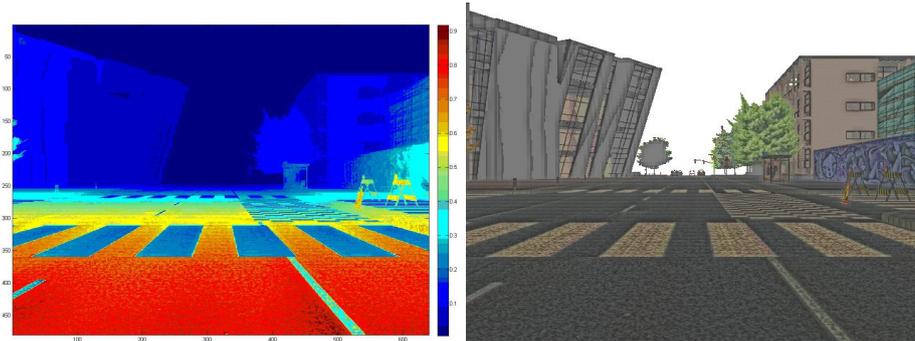


FIGURE 4.9 – Carte de transmittance estimée par le minimum des cartes de transmittance de chaque canal, à gauche. Résultat de la restauration en utilisant cette carte, à gauche.

Comme nous l'avons vu dans le chapitre 2, le fait de prendre le minimum des cartes de transmittance des 3 canaux nous permet d'améliorer la couleur des objets les plus atténués par le brouillard. En revanche, un défaut de cette méthode est que certaines régions de l'image telles que les marquages qui apparaissent blancs dans l'image avec brouillard n'ont pas, en réalité, les mêmes valeurs d'intensités sur les trois canaux (une faible différence). L'opération $\min(t_R, t_G, t_B)$ va faire ressortir une couleur. C'est pourquoi ils apparaissent avec une couleur jaune dans l'image restaurée. Ce problème peut aussi être rencontré dans les zones du ciel qui parfois ne sont pas blanches (voir figure 4.2) et peuvent avoir différentes couleurs (bleu, orange...) en fonction des conditions atmosphériques.

4.1.1.7 Lissage de la carte de transmittance

En appliquant à la carte t un lissage avec une gaussienne bidimensionnelle, nous allons homogénéiser les différentes régions et réduire les effets blocs générés par l'égalisation. La figure 4.10 montre la carte de transmittance lissée et l'image restaurée avec cette carte.

Pour éliminer les effets blocs, la taille de la gaussienne G_{s1} doit être supérieure de 3 fois minimum (vérifié expérimentalement) à la taille de la fenêtre d'égalisation ce qui permet de bien lisser les zones uniformes. En revanche, ce processus dégrade l'image au niveau des contours.

Nous constatons que dans le voisinage des différents contours de l'image, les pixels deviennent plus sombres à cause d'une surestimation de la profondeur due à la diffusion par la gaussienne de l'information des objets les plus distants vers les

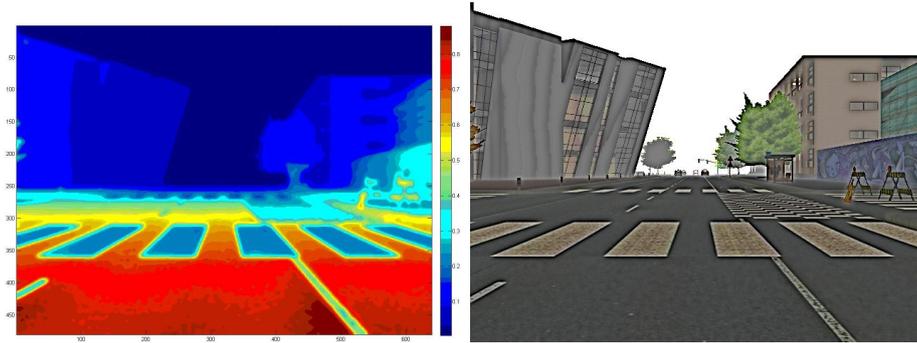


FIGURE 4.10 – Application d’une gaussienne de taille $G_{s1} = 15$ et d’écart type $\sigma = 3$ à la carte de transmittance, à gauche ; Restauration avec cette nouvelle carte de transmittance, à droite

objets les plus proches. De la même façon, certains pixels vont devenir plus clairs (on rajoute du brouillard) à cause d’une sous-estimation de la profondeur due à la diffusion par la gaussienne de l’information des objets les plus proches vers les objets les plus distants. Pour résoudre ce problème nous allons d’abord détecter ces pixels et appliquer un lissage pondéré en fonction de la distance spatiale des pixels aux contours.

Afin de détecter les pixels dégradés par le lissage, nous commençons par détecter les contours à fort gradient qui correspondent aux régions où nous avons surestimé ou sous-estimé la profondeur. Pour cela, nous utilisons un détecteur de Sobel en fixant un seuil assez haut pour détecter les fortes transitions. Ensuite, comme nous savons que la dégradation se trouve autour de ses contours et s’étale sur une distance de la même taille que la gaussienne, nous appliquons une simple dilatation de taille G_{s1} aux contours détectés. Sur ces régions, nous calculons la distance de chaque pixel aux bords comme présenté sur la figure 4.11.

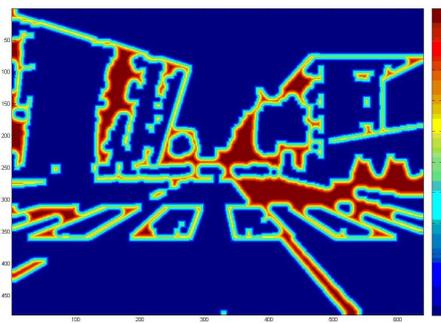


FIGURE 4.11 – Distance aux bords de pixels proches des contours.

Nous appliquons finalement aux pixels de ces régions un lissage pondéré par les distances aux bords à l’aide de l’équation 4.3.

$$t = \Delta t_2 + (1 - \Delta)t_1 \quad (4.3)$$

avec Δ la distance aux bords, t_1 et t_2 les cartes de transmittance lissées avec, respectivement, une gaussienne de grande taille G_{s1} et une gaussienne de petite taille G_{s2} .

La figure 4.12 montre la carte de transmittance et la restauration.

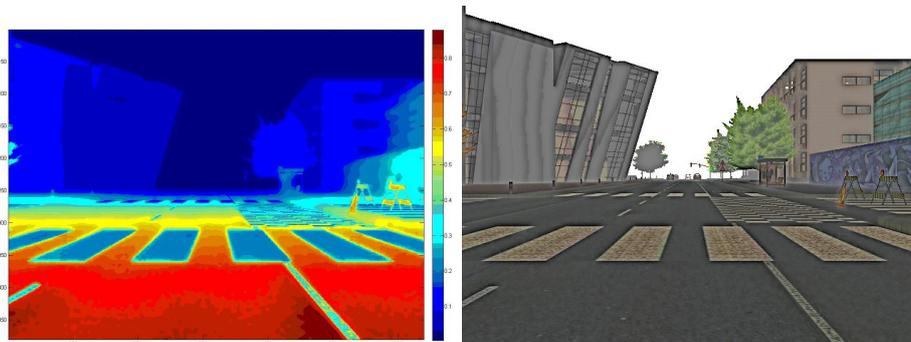


FIGURE 4.12 – Carte de transmittance estimée et restauration

Nous remarquons sur cette figure que les bords ont été améliorés et contiennent moins de pixels saturés (transmittance surestimée ou sous-estimée) par rapport au premier lissage avec G_{s1} (figure 4.10). Toutefois, il reste quelques pixels saturés. Afin d'améliorer le lissage, nous pouvons itérer le processus en appliquant l'équation 4.3 plusieurs fois. La figure 4.13 montre le résultat après 1, 2 et 4 itérations. Nous remarquons que les bords contiennent moins de pixels dégradés (voir les différents bords des objets de la scène : marquages, arbre, bâtiments).



FIGURE 4.13 – Lissage itératif : résultats après 1, 2 et 4 itérations

4.1.1.8 Fusion avec approche monde plan

Une méthode que nous avons trouvée intéressante est celle que nous avons présentée dans le chapitre 2 et qui permet d'estimer la profondeur de la route en supposant celle-ci plane et en utilisant un calibrage géométrique en connaissant les paramètres et la position de la caméra.

La restauration du plan de la route avec cette méthode (en supposant β et I_∞ connues) donne de très bons résultats quasi comparables à une restauration idéale dans le cas où la route est plane. La figure 4.14 montre la carte de profondeur et la restauration de la route obtenue avec cette méthode.

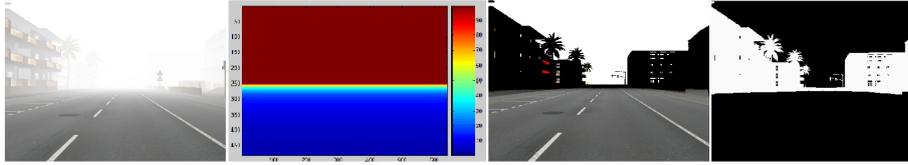


FIGURE 4.14 – De gauche à droite, image originale, profondeur du plan de la route, restauration et segmentation route/objets verticaux

Le problème est que le reste de la scène (objets verticaux) est mal restaurée puisque la profondeur est surestimée. L'équation de Koschmieder montre que cette surestimation de la profondeur fait tendre l'intensité de l'image après restauration vers zéro. Sur la figure 4.15, nous avons tracé la courbe de l'image restaurée R en fonction de I et de la distance d .

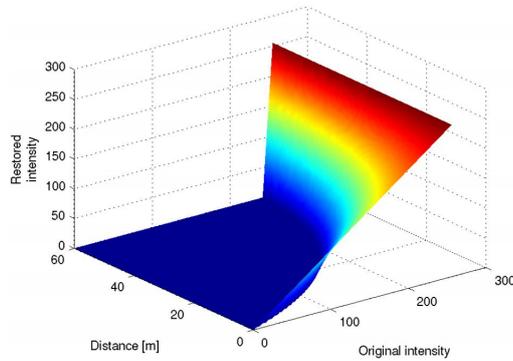


FIGURE 4.15 – Tracé 3D de l'intensité de l'image restaurée en fonction de I et de d . $I_\infty = 255$ et $\beta = 0,05$.

Cette surestimation peut être utilisée en notre faveur afin de détecter les objets verticaux : en segmentant l'image à l'aide d'un seuil très bas ou nul nous récupérons une image binaire correspondant à la séparation entre les objets verticaux et le plan de la route. La figure 4.14 montre les résultats de la segmentation obtenus par cette méthode. Nous constatons que le monde vertical a été correctement segmenté.

Nous pouvons, s'appuyant sur cette segmentation, combiner notre méthode de restauration avec cette approche. Il s'agira d'appliquer la restauration du monde plan sur les objets horizontaux et notre algorithme sur les objets verticaux. Un exemple de cette combinaison est proposé sur la figure 4.16.

4.1.2 Optimisation des paramètres : Recuit simulé

Rappelons d'abord les différents paramètres de la méthode :

- W_s : la taille de la fenêtre d'égalisation ;
- G_{s1} : la taille de la gaussienne (grande taille) ;
- G_{s2} : la taille de la gaussienne (petite taille) ;

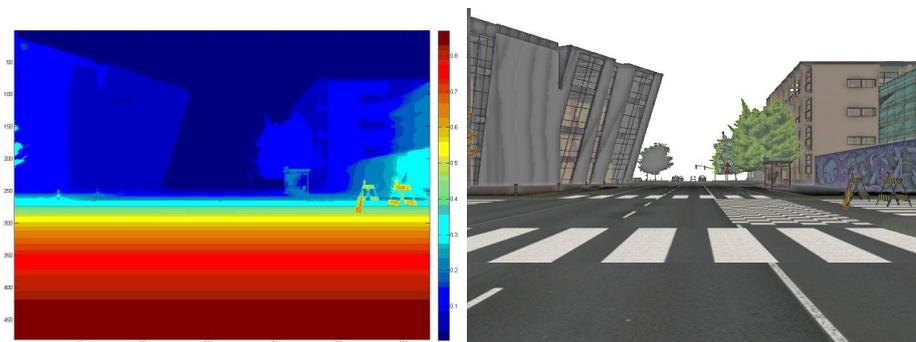


FIGURE 4.16 – Combinaison de notre approche avec l'approche monde plan : carte de transmittance et restauration

– a et b : les coefficients permettant l'ajustement d'intensité.

Afin de bien restaurer les détails nous avons choisi de fixer le paramètre G_{s2} à la valeur 3.

Recuit simulé

Afin de trouver la meilleure combinaison des paramètres, nous avons commencé par choisir les paramètres qui donnent les meilleurs résultats en testant différentes combinaisons. Ensuite, nous avons appliqué notre restauration sur toutes les images FRIDA dans le cas d'un brouillard uniforme. Après, nous avons calculé l'erreur absolue moyenne de chaque image par rapport à l'image restaurée idéale et nous avons déduit l'erreur moyenne sur toutes les images. Parmi les images FRIDA nous avons sélectionné 5 images les plus représentatives (ayant une erreur absolue moyenne proche de la moyenne des erreurs de toutes les images). Sur ces 5 images nous avons appliqué un algorithme de recuit simulé [Hwang 1988] afin d'optimiser nos paramètres.

Le recuit simulé est un algorithme d'optimisation itératif permettant de déduire un minimum global d'une fonction. Dans notre cas, nous cherchons à minimiser l'erreur moyenne entre les 5 images restaurées et leur correspondant restaurée idéale.

Le principe est le suivant : à chaque itération une nouvelle configuration de paramètres est générée aléatoirement dans le voisinage de la précédente configuration. Cette nouvelle configuration est utilisée pour effectuer la restauration et l'erreur trouvée est comparée à la meilleure erreur (la plus petite) précédemment trouvée. Une nouvelle configuration est acceptée systématiquement si l'erreur diminue et de façon aléatoire, si elle augmente, en fonction d'une variable appelée température, qui diminue à chaque itération (cycles de refroidissement). Nous comparons donc un nombre aléatoire entre 0 et 1 à la fonction $e^{\frac{-\Delta E}{k_b T}}$, avec $-\Delta E$ la variation de la nouvelle erreur par rapport à la meilleure erreur trouvée, k_b une constante choisie égale à 1. Si cette probabilité est plus grande que le nombre aléatoire nous acceptons la configuration : plus l'erreur est grande et plus la température diminue, moins la configuration a de chance d'être acceptée.

L'algorithme est répété jusqu'à ce qu'il conserve la même configuration après plusieurs itérations consécutives.

Afin de garantir la convergence vers un minimum global ou vers un minimum local très proche du minimum global, nous devons choisir les meilleurs paramètres (température, loi de décroissance de la température et nombres d'itérations successive et totale) permettant d'assurer la convergence. Pour cela, nous avons essayé plusieurs paramètres et étudié, à chaque fois, l'allure de la courbe d'erreur. Les paramètres sélectionnés, dans notre cas, sont : la température $T = 1$, le nombre d'itérations maximum de 10000, le nombre d'itérations successives de 100 et un coefficient de baisse de température tel que $T_{i+1} = 0,995T_i$ avec i le temps (itérations).

La figure 4.17 montre l'évolution de l'erreur en fonction des nombres d'itérations. Nous remarquons que l'erreur augmente parfois sans trop s'éloigner de la meilleure erreur précédemment trouvée et que vers la fin l'algorithme ne trouve pas de meilleure solution et arrête au bout de 100 itérations. Notons que l'algorithme du recuit simulé a été lancé plusieurs fois afin de s'assurer que nous convergions toujours vers le même résultat.

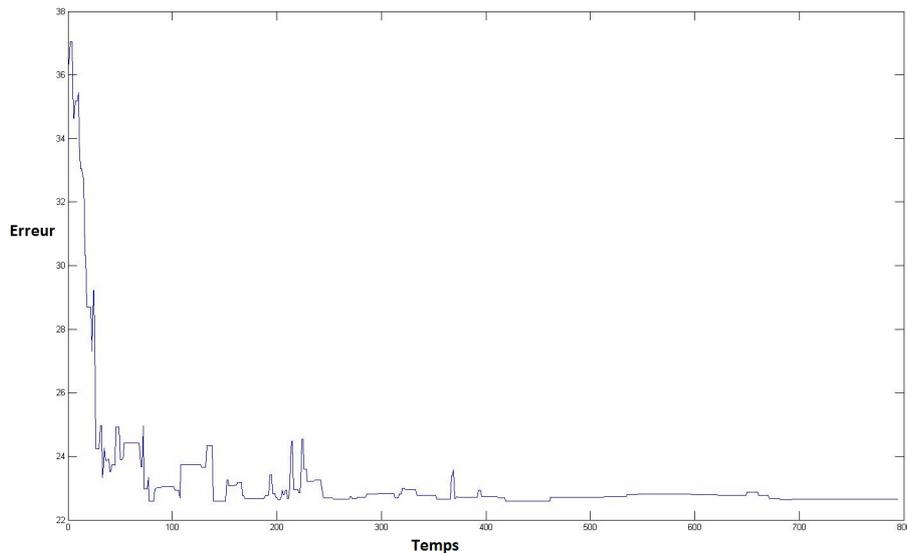


FIGURE 4.17 – Recuit simulé : évolution de l'erreur dans le temps. Seulement les configurations acceptées par le recuit simulé sont affichées sur cette figure.

Le tableau 4.1 montre les paramètres sélectionnés par le recuit simulé et qui donnent le meilleur score (plus petite erreur).

Les valeurs de W_s et G_{s1} auxquelles le recuit simulé converge peuvent être expliquées comme suit : les objets les plus distants sont les plus affectés par le brouillard et donc, à cause de leur petite taille dans l'image, une taille de W_s petite permet de restaurer leur contraste d'une meilleure façon. De même, les petits détails (signalisation par exemple) dans l'image seront mieux restaurés avec un petit W_s . Le paramètre G_{s1} doit être plus grand que W_s afin d'homogénéiser la

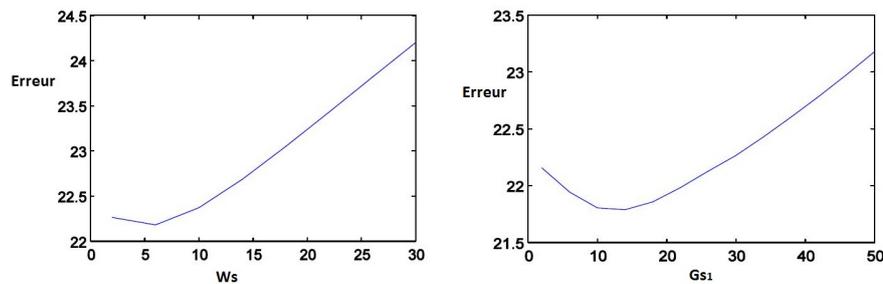
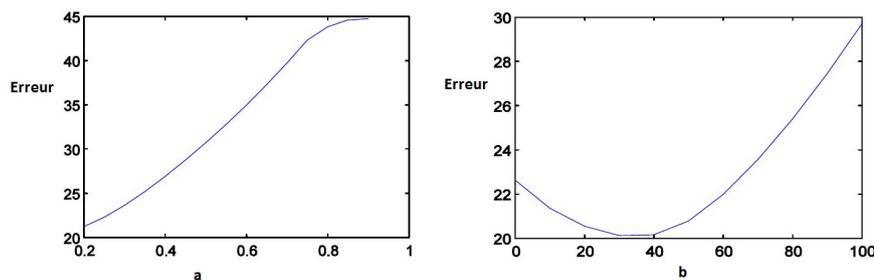
Paramètre	W_s	G_{s1}	a	b
Valeur	5	15	0,2	40
Écart type	0,55	0,51	0,06	2,32

TABLE 4.1 – Paramètres optimaux de l'algorithme déduits par le recuit simulé.

transmittance des objet les plus grands.

Étude de la sensibilité

Pour étudier la sensibilité de la méthode aux différents paramètres nous faisons varier l'un des paramètres en fixant les autres à la meilleure valeur trouvée par le recuit simulé. Ensuite, nous calculons l'erreur moyenne sur les 66 images de FRIDA dans le cas d'un brouillard uniforme. Les figure 4.18, 4.19 et 4.20 montrent les résultats obtenus.

FIGURE 4.18 – Erreur en fonction de W_s et G_{s1} FIGURE 4.19 – Erreur en fonction de a et b

Nous remarquons sur la figure 4.18 que la méthode n'est pas très sensible aux variations de W_s et G_{s1} (l'erreur reste comprise entre 22 et 24). Nous choisissons pour ces paramètres les valeurs suivantes $W_s = 5$ $G_{s1} = 11$. Au contraire, la figure 4.19 montre que la méthode est plus sensible aux variations de a et b . Nous sélectionnons $a = 0,2$ et $b = 40$.

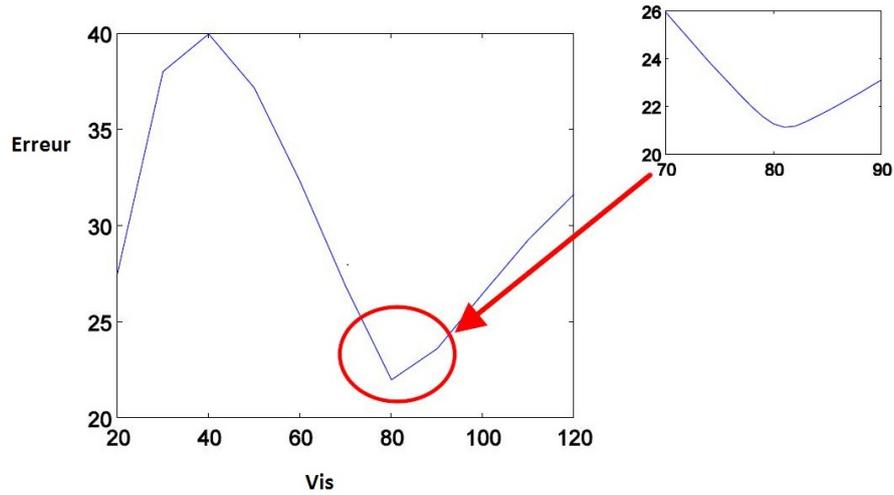


FIGURE 4.20 – Erreur en fonction de la distance de visibilité pour la restauration de la route

Comme nous l’avons expliqué précédemment, la méthode sans utilisation du monde plan est insensible aux variations de β puisqu’elle dépend seulement du produit βd . Nous étudions donc la sensibilité à β en utilisant la méthode du monde plan [Hautière 2007]. Comme nous l’avons montré, le paramètre β est inversement proportionnel à la distance de visibilité Vis . Nous avons choisi de travailler avec Vis au lieu de β . La figure 4.20 montre l’erreur en fonction de Vis . Cette courbe présente un minimum autour de 85m qui correspond à la distance de visibilité utilisée pour l’ajout du brouillard dans les images de synthèse. Nous constatons que la méthode avec l’hypothèse monde plan est très sensible aux variations de Vis .

Pour des petites valeurs de Vis (inférieures à 40m), l’erreur diminue. En effet, lorsque Vis diminue, une grande partie de l’image est classée comme étant des objets verticaux lors de l’étape de segmentation. Donc une petite partie de l’image est restaurée avec la mauvaise valeur de Vis en utilisant le monde plan, le reste de l’image est restauré avec notre méthode. Et il est moins pénalisant de restaurer la surface de la route avec notre méthode qu’avec une la méthode du monde plan et une Vis erronée.

4.1.3 Évaluation

4.1.3.1 Résultats qualitatifs

La figure 4.21 montre les résultats de restauration d’images de synthèse en utilisant la méthode présentée sans puis avec l’hypothèse “monde plan”.

Nous remarquons que la visibilité a été bien améliorée après la restauration. Nous pouvons dans cette figure comparer les résultats obtenus avec deux valeurs de W_s . Le fait d’utiliser une taille W_s petite permet de mieux restaurer les objets de

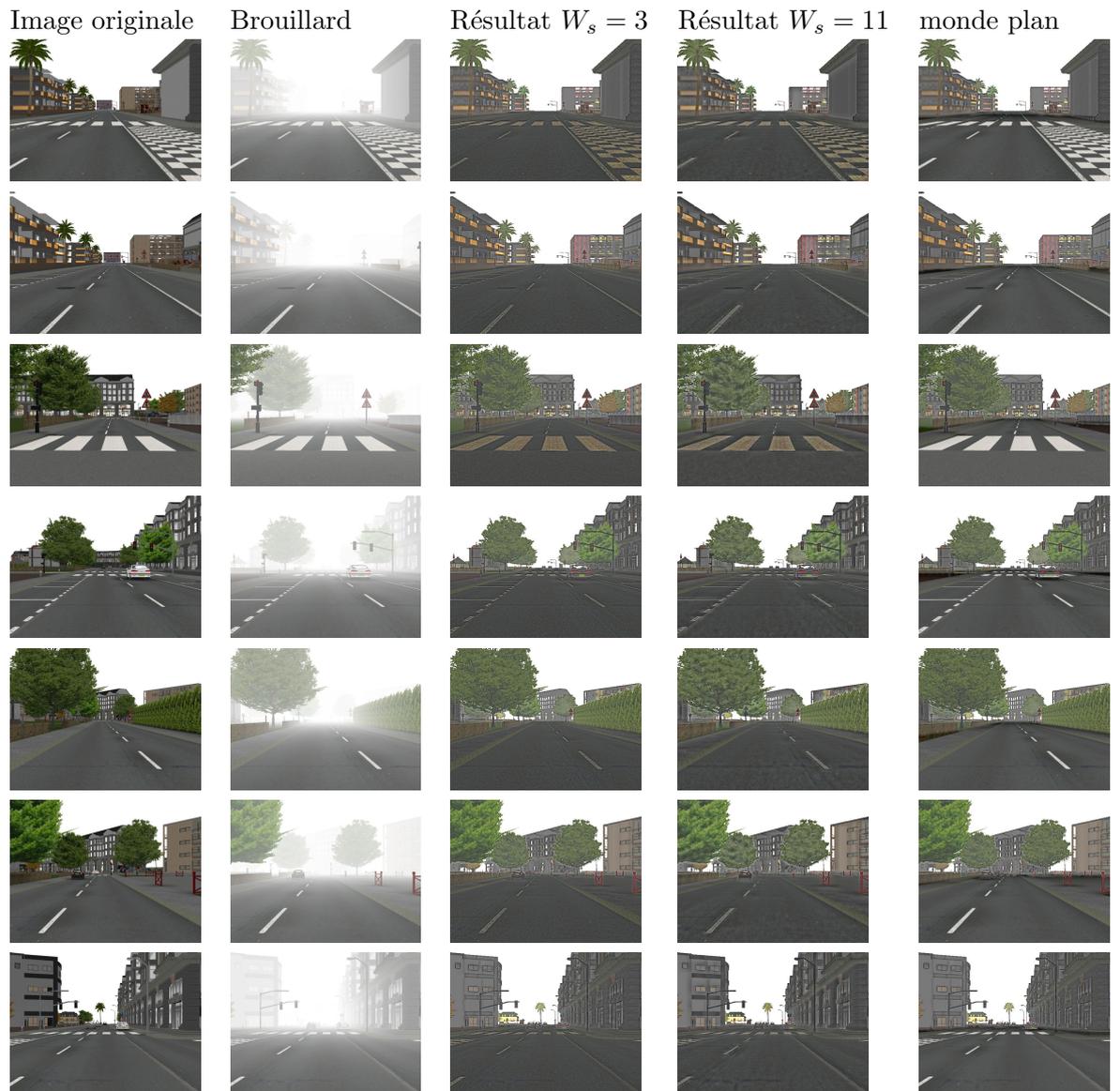


FIGURE 4.21 – De gauche à droite : images originales, images avec brouillard, résultats de la restauration avec $W_s = 3$, résultat avec $W_s = 11$, résultat avec le monde plan. Les autres paramètres utilisés pour effectuer la restauration sont $G_{s1} = 9$, $G_{s2} = 3$, $a = 0,3$, $b = 50$, $Vis = 85m$.

faible dimension et les détails. En revanche, pour les objets de grande dimension, les contrastes ne vont pas ressembler à l'image originale. D'un autre côté, une valeur de W_s plus importante (ici 11) permet de résoudre ce problème et obtenir un meilleur contraste au niveau des grands objets, comme le montre la quatrième colonne de la figure 4.21. Nous pourrions aussi utiliser différentes tailles pour effectuer l'égalisation et faire une moyenne des résultats obtenus afin d'avoir un contraste équitable entre les grand et les petits objets. Cependant, cela demanderait plus de temps de traitement. Notons aussi que la taille de la fenêtre peut aussi être adaptée en fonction de la résolution spatiale de l'image puisque la taille des objets en pixels va changer.

La figure 4.22 montre les résultats obtenus dans le cas d'images réelles. Ces images ayant été, pour la plupart, récupérées sur internet, nous n'avons donc pas accès aux paramètres caméra utilisés. Aussi, nous n'avons pas utilisé l'approche "monde plan".

Nous remarquons que nous améliorons la visibilité des détails qui sont difficilement visibles dans le brouillard. Nous avons privilégié une très forte restauration du contraste sur le rendu visuel. En effet, notre méthode est destinée à améliorer des systèmes ADAS. Aussi, nous avons opté pour l'estimation de l'intensité du ciel comme étant le maximum. Cela génère un ciel très coloré après la restauration. Notons enfin, comme le montre la dernière image, que notre méthode donne des résultats intéressants dans la cas d'un brouillard de nuit lorsque l'intensité de l'éclairage de la scène est assez élevée.

La figure 4.23 montre les résultats obtenus sur des images de la base de données Karlsruhe présentée dans le chapitre 3. Nous constatons que l'image devient très sombre après restauration. Une façon d'améliorer le résultat est d'appliquer une égalisation globale après restauration, comme présenté dans la dernière ligne de chaque colonne de la figure. Nous allons montrer dans la section 4.2.1 une autre façon pour surmonter ce problème.

Nous allons, dans la suite, présenter les résultats d'évaluation quantitatifs de la méthode sur les bases de données FRIDA et Karlsruhe.

4.1.3.2 Résultats quantitatifs

Afin d'évaluer les résultats de manière objective, nous comparons l'erreur de restauration obtenue avec notre méthode à différentes approches. La base d'images de synthèse FRIDA et la base d'images réelles Karlsruhe ont été utilisées pour effectuer l'évaluation. L'erreur d'évaluation est calculée comme étant l'erreur absolue moyenne entre l'image restaurée et l'image restaurée idéale (voir chapitre 3).

Les résultats obtenues sur ces deux bases de données sont présentés dans les tableaux 4.2 et 4.3. Les valeurs d'erreur des différentes méthodes ont été extraites de [Tarel 2010]. Nous avons évalué notre méthode avec et sans l'hypothèse "monde plan". Les paramètres utilisés correspondent à ceux sélectionnés par le recuit simulé.

Nous constatons que pour les différents types de brouillard, les résultats obtenues avec notre méthode, avec et sans monde plan, donnent un meilleur score (plus petite



FIGURE 4.22 – Images originales et restaurées avec $W_s = 11$, $G_{s1} = 21$, $G_{s2} = 11$, $a = 0,3$, $b = 50$. L'intensité du ciel est estimée comme étant le maximum sur toute l'image



FIGURE 4.23 – Base d’images de Karlsruhe. Dans chaque colonne (a),(b),(c),(d),(e) et (f) nous montrons l’image originale sans brouillard, ajout du brouillard, restauration avec $W_s = 11$, $G_{s1} = 21$, $G_{s2} = 11$, $a = 0, 3$, $b = 50$ et restauration globale du contraste pour une meilleur visibilité.

erreur) par rapport aux autres méthodes. Les résultats obtenues sur les deux bases de données sont similaires.

Algorithme	Uniforme	β variable	I_∞ variable	Mixte	Tous les types
Aucun	81,6±12,3	78,7±12,3	69,0±10,9	66,4±10,8	73,9±13,2
MSR	46,7±12,3	86,4±24,7	44,8±17,1	83,7±24,9	65,4±28,9
CLAHE	66,9±10,7	64,5±9,7	54,5±8,5	54,6±7,8	60,1±10,9
DCP	46,3±15,6	46,9±17,0	43,7±16,2	44,1±17,5	45,1±16,7
NBPC	50,8±11,5	50,5±11,5	38,5±9,0	38,0±8,7	44,5±12,1
NBPC+PA	31,1±10,2	36,0±10,3	26,7±5,1	28,4±5,9	30,6±8,9
CLAHE FOG	30,9±11,1	25,9±7,1	23,5±6,1	24,4±5,9	26,1±8,3
CLAHE FOG+PA	29,6±11,3	24,0±7,7	25,7±4,4	25,2±6,2	26,1±8,1

TABLE 4.2 – Évaluation quantitative et comparaison avec d'autres méthodes sur les images de synthèses : Retinex (MSR), égalisation d'histogramme locale classique (CLAHE), hypothèse Dark Channel (DCP), contraintes sur le voile (NBPC), NBPC + hypothèse monde plan (NBPC + PA) et notre approche (CLAHE FOG). Les valeurs sont l'erreur absolue moyenne entre l'image restaurée et la restaurée idéale et les écarts-type sur les 4 types de brouillard : uniforme, densité β variable, intensité du ciel I_∞ variable, β et I_∞ variables et la moyenne sur les 4 types.

4.2 Amélioration du rendu visuel

Dans cette section, nous allons présenter les améliorations effectuées sur notre méthode de façon à obtenir un meilleur rendu visuel.

4.2.1 Accroissement de la dynamique

La première amélioration consiste en une modification des étapes et des paramètres de la méthode précédente afin d'augmenter la dynamique de l'image restaurée tout en conservant les couleurs.

En effet, dans la méthode précédente, le paramètre b d'ajustement d'intensité permet d'obtenir une image globalement proche de l'image sans brouillard mais atténuée les contrastes (à la différence du paramètre a qui ne modifie pas les contrastes). Aussi la première modification est de choisir $b = 0$. La figure 4.24 montre le résultat obtenu, sur l'image présenté précédemment dans la figure 4.2, tel que $a = 0,5$ et $b = 0$. En comparant le résultat avec la méthode précédente, nous constatons une amélioration des contrastes.

Dans l'algorithme initial, les informations couleurs sont exploitées pour améliorer la carte de transmittance (section 4.1.1.6 avant l'étape de lissage de celle-ci (section 4.1.1.7). Cependant, cette dernière réduit et décale la dynamique de l'image. Le problème est qu'une fois que les informations couleur et contraste sont mélan-

Algorithme	Uniforme	β variable	I_∞ variable	Mixte	Tous les types
Aucun	73,1±8,9	71,4±10,1	61,8±8,0	60,4±8,5	66,6±10,5
MSR	47,5±8,8	74,5±21,7	47,6±14,0	72,2±20,4	60,5±10,5
CLAHE	53,4±8,8	55,8± 9,4	47,1±7,6	49,6±7,8	51,5±9,1
DCP	32,8±14,1	36,2±10,2	34,9±14,2	36,9±11,5	35,1±12,7
NBPC	41,8±6,7	43,0±6,4	35,8±5,3	36,5±4,8	39,3±6,7
NBPC+PA	29,8±5,9	31,5±6,8	27,3±5,7	29,6±6,7	28,8±6,6
CLAHE FOG	29,7±5,0	32,2±4,0	26,5±6,0	28,8±5,0	29,3±5,0
CLAHE FOG+PA	28,2±5,0	30,9±4,7	27,4±5,5	29,6±5,5	29,1±5,2

TABLE 4.3 – Évaluation quantitative et comparaison avec d’autres méthodes sur des images réelles avec brouillard synthétique : Retinex (MSR), égalisation d’histogramme locale classique (CLAHE), hypothèse Dark Channel (DCP), contraintes sur le voile (NBPC), NBPC + hypothèse monde plan et notre approche (CLAHE FOG). Les valeurs sont l’erreur absolue moyenne entre l’image restaurée et la restaurée idéale et les écarts-type sur les 4 types de brouillard : uniforme, densité β variable, intensité du ciel I_∞ variable, β et I_∞ variables et la moyenne sur les 4 types.



FIGURE 4.24 – À gauche, image restauré avec les paramètres optimisés ($a = 0,3$ et $b = 50$) ; à droite, image restauré avec $a = 0,5$ et $b = 0$

gées au travers du lissage, revenir à la dynamique initiale de l’image, par l’opération inverse de $aI + b$, introduit des couleurs très saturées.

La solution que nous proposons est de séparer les étapes amélioration du contraste et lissage de l’étape réhaussement des contrastes. Le nouvel algorithme est le suivant :

1. Amélioration des contrastes de la luminance L de l’image originale I dans l’espace couleur Lab ;
2. Ajustement d’intensité de la luminance de l’image après égalisation d’histogramme locale : $a.L_{heq}$ avec a choisi empiriquement égal à 0,6 ;
3. Estimation de la carte de transmittance de la luminance $t = \frac{L-L_\infty}{a.L_{heq}-L_\infty}$. L’intensité du ciel L_∞ est choisie comme le maximum de L ;

4. Lissage de la carte de transmittance ;
5. Restauration avec la transmittance lissée ;
6. Ajustement inverse d'intensité de l'image restaurée R/a afin de récupérer la dynamique initiale de l'image ;
7. Conversion de l'image restaurée en couleur (RGB) à l'aide des composantes chromatiques de l'image originale dans l'espace couleur Lab ;
8. Rehaussement des couleurs : estimation de la transmittance sur chaque canal R, G et B ensuite, estimation de la carte de transmittance finale comme le minimum des trois cartes. L'intensité du ciel de chaque canal est choisie comme le maximum du canal correspondant dans l'image originale : $I_\infty^c = \max(I^c)$ avec $c \in \{R, G, B\}$;
9. Restauration finale en utilisant la carte de transmittance déduite : applications du modèle de Koschmidier inverse avec $I_\infty^c = \max(I^c)$.

La figure 4.25 montre les résultats obtenus après les étapes 7 et 9. La première image possède un meilleur contraste que celle issue de l'algorithme initial (4.24) mais les couleurs sont toujours atténuées. Par l'utilisation des deux dernières étapes (8 et 9), la couleur bleuâtre du ciel ressort toujours mais dégrade moins l'aspect visuel de l'image restaurée.



FIGURE 4.25 – Amélioration du rang dynamique. À gauche, image restaurée obtenue à l'étape 7 de l'algorithme, juste avant le rehaussement des couleurs ; à droite, image restaurée finale.

La figure 4.26 présente l'image restaurée avec l'algorithme initial et l'image restaurée finale du nouvel algorithme sur l'image de synthèse précédemment utilisée. Le résultat visuel de la nouvelle approche apparaît moins bon, en particulier sur les bâtiments. Cela vient du fait que la nouvelle approche conserve les couleurs de l'image originale et qu'en présence d'un fort brouillard, les murs apparaissent blanc. Cependant, nous pouvons constater que les marquages sont quant à eux bien restaurés et ne présentent pas une couleur orangée comme dans la méthode initiale. Enfin, le contraste de l'image issue de la nouvelle méthode est bien plus important car toute la dynamique est utilisée.

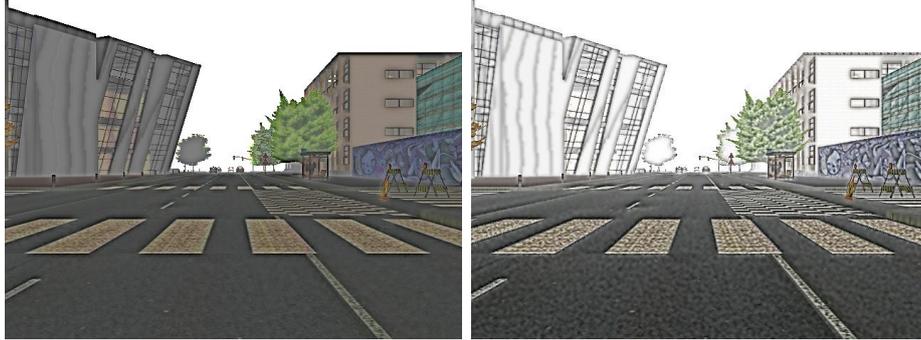


FIGURE 4.26 – De gauche à droite, image restaurée avec l’algorithme initial et image restaurée finale du nouvel algorithme.

4.2.2 Lissage de la carte de transmittance par un filtre bilatéral

Comme nous l’avons vu dans chapitre 2, il existe de nombreuses méthodes de lissage. Notre choix d’effectuer le lissage de la carte de transmittance, comme nous l’avons décrit dans la section 4.1.1.7, est justifié par le fait qu’il ne nécessite pas beaucoup de temps de traitement et qu’il est donc adapté aux systèmes ADAS. Cependant, afin de limiter les effets blocs induit par l’égalisation et pour des applications ne nécessitant pas le temps réel, nous avons étudié le filtre bilatéral.

Celui-ci permet de lisser les zones uniformes tout en conservant les contours d’une image guide. Afin d’obtenir une image ayant des contours proche de ceux de l’image originale, nous utilisons un filtre bilatéral guidé par l’image d’entrée. Le reste de l’algorithme, présenté dans la section 4.2.1, est identique sauf que les poids W_{ng} du filtre sont calculés à partir de l’image d’entrée.

La figure 4.27 montre les résultats obtenus avec ce filtre. Nous constatons que l’image est beaucoup plus lisse comparée à la méthode précédente, qu’elle ne contient pas d’artefacts et que les contours ne sont pas dégradés.



FIGURE 4.27 – De gauche à droite, image restaurée par l’algorithme de la section 4.2.1 et image restaurée en utilisant le lissage de t par un filtre bilatéral guidé sur l’image d’entrée.

Du point de vue qualité d’image, ce lissage bilatéral va induire une perte de

Image originale



Restauration



FIGURE 4.28 – Restauration avec le filtre bilatérale et l'accroissement de la dynamique

Image originale



Restauration



FIGURE 4.29 – Restauration avec le filtre bilatérale et l'accroissement de la dynamique

contraste par rapport à la méthode précédente. Il faudra adapter le choix du lissage à l'application souhaitée. Pour un observateur humain, le filtrage bilatéral sera à privilégier. En revanche, pour effectuer des traitement sur l'image restaurée, nous estimons que le lissage initial sera de meilleure qualité.

Les figures 4.28 et 4.29 montrent les résultats obtenus en utilisant un filtre bilatéral et l'accroissement de la dynamique de l'image.

Nous observons que les images noir et blanc obtenues affichent une meilleure visibilité que celles obtenues dans la section 4.1.3.1. Quant aux images couleurs, malgré les variations de la couleur du ciel d'une image à l'autre, nous obtenons des couleurs réalistes dans l'image restaurée.

4.2.3 Couleur

Afin d'améliorer le rendu colorimétrique de l'image restaurée, nous avons étudié les modifications induites par l'équation de Koschmieder sur chacun des canaux dans l'espace HSV . Dans l'équation 4.1, la teinte H ne change pas entre l'image originale et l'image restaurée. En revanche, la saturation S va être modifiée suivant l'équation $S_R = t.S_I$, avec S_R la saturation de l'image restaurée, S_I la saturation de l'image originale et t la transmittance. Ainsi, en estimant la transmittance de la scène sur la luminance V , nous pouvons déduire la saturation de l'image restaurée.



FIGURE 4.30 – Restauration (Image restaurée avec couleurs originales et image avec des couleurs rehaussées)

La figure 4.30 montre la comparaison entre la restauration en utilisant les couleurs originales et le résultat obtenu avec cette méthode. Nous constatons que les couleurs sont moins saturées au niveau des objets colorés. Le défaut de cette méthode est qu'au niveau du ciel la transmittance n'est pas très bien estimée ce qui introduit des couleurs saturées dans le cas d'un ciel coloré.

4.3 Conclusion

Nous avons proposé, dans ce chapitre, une méthode de restauration d'images de brouillard fondée sur le modèle optique du brouillard et sur une approche signal (égalisation d'histogramme). L'évaluation de la méthode sur des bases de données d'images de synthèse et réelles montre que quantitativement nous obtenons un gain

non négligeable par rapport à d'autres méthodes de l'état de l'art. Les résultats qualitatifs montrent aussi que nous améliorons la visibilité de la scène même pour les objets les plus éloignés et qui sont les plus dégradés par l'effet du voile atmosphérique. Le traitement des images se fait en temps réel et une telle méthode peut donc être embarquée dans des véhicules.

Nous avons proposé des modifications permettant d'améliorer le rendu visuel de la méthode. En particulier, l'accroissement de la dynamique permet à la méthode d'être plus robuste aux changements de couleur du ciel (l'image restaurée garde le même aspect couleur que l'image originale) et d'obtenir une image avec un meilleur contraste global. D'un autre côté, un lissage par filtre bilatéral plus robuste au bruit a été adapté à notre problème.

Du fait du caractère modulaire de notre algorithme, chacune des étapes de notre algorithme peut être remplacée par d'autres méthodes accomplissant la même tâche et qui peuvent être adaptées aux objectifs de l'application souhaitée (meilleur rendu visuel, contraste, couleur...).

Restauration d'images de pluie

Sommaire

5.1	Introduction	103
5.2	Généralités sur l'“inpainting”	104
5.3	État de l'art	105
5.3.1	“Inpainting” de régions contenant des structures	105
5.3.2	“Inpainting” rapide par diffusion isotropique	106
5.3.3	Synthèse de texture	108
5.3.4	“Inpainting” vidéo	111
5.4	Méthodes sélectionnées	112
5.4.1	Algorithme de Telea	113
5.4.2	Algorithme de Bornemann	115
5.5	Optimisation des paramètres sur des images routières	120
5.6	Conclusion	123

5.1 Introduction

Le problème qui nous intéresse dans cette partie est la restauration d'images dégradées par des gouttes de pluie focalisées présentes sur le pare-brise. Des travaux sont en cours pour la détection des gouttes défocalisées.

Après la détection des gouttes focalisées par un algorithme approprié [Cord 2011], le but de la restauration des images de pluie dans notre cadre d'étude des ADAS, est de reconstruire les parties masquées par les gouttes en utilisant les informations disponibles dans l'image. Nous supposons donc la position des zones à remplir connue.

Nous pouvons envisager l'utilisation du modèle optique afin de restaurer l'image, sauf que, dans notre cas, les gouttes sont de petites tailles et l'information contenue dans le bloc de pixels de la goutte est insuffisante pour reconstruire l'arrière plan. En revanche, les méthodes d'“inpainting”, que nous allons présenter dans la suite, sont bien adaptées pour reconstruire des zones peu étendues.

Le plan de ce chapitre est le suivant : nous commençons par expliquer ce qu'est l'“inpainting” et quels sont les différents types d'approches existantes. Ensuite, pour chaque type d'approches, nous allons présenter des exemples de méthodes de l'état de l'art. Nous avons ensuite sélectionné un type d'approche pour notre problématique. Pour ce type d'approches, 2 méthodes existantes ont été testées. La méthode

que nous avons jugée la plus pertinente a été retenue et adaptée au cas d'images routières.

5.2 Généralités sur l'“inpainting”

L'“inpainting” (ou retouche d'images en français) est la technique qui consiste à modifier une image de telle façon que la modification soit indétectable pour un observateur ne connaissant pas l'image d'origine.

Considérons le cas d'une image contenant des zones dégradées dont les positions sont connues. L'“inpainting” consiste à restaurer l'information manquante en utilisant les données spatiales et/ou temporelles voisines.

Notons la différence avec le débruitage d'images qui suppose que chaque pixel contient à la fois l'information et le bruit, alors que l'“inpainting” considère que la zone dégradée ne contient aucune information. Les méthodes de débruitages classiques ne peuvent donc pas s'appliquer.

Dans la littérature il existe trois types de techniques d'“inpainting”. Certaines méthodes utilisent les informations temporelles [Patwardhan 2005, Patwardhan 2007], d'autres les informations spatiales [Bertalmio 2000, Tang 2005, Elango 2009, Bornemann 2007], d'autres la synthèse de texture [Tang 2005, Criminisi 2004] et quelques méthodes utilisent une combinaison de ces 3 types d'approches [Sun 2005, Do 2006, Komodakis 2006]. Chacune de ces méthodes présente des points forts et des défauts. Par exemple, les algorithmes spatiaux préservent bien les structures mais ne restaurent pas bien les textures. Le problème avec les méthodes utilisant une séquence vidéo est qu'elles supposent que l'information est présente dans plusieurs images et ne fonctionnent que pour des zones de petites tailles. Les algorithmes de synthèse de textures permettent de remplir des régions larges mais l'utilisateur doit spécifier la texture utilisée en plus des régions à remplir.

Les applications de l'“inpainting” sont nombreuses : restauration de photos endommagées, élimination des écritures et des objets indésirables dans l'image...

La figure 5.1 montre un exemple illustrant cette technique.

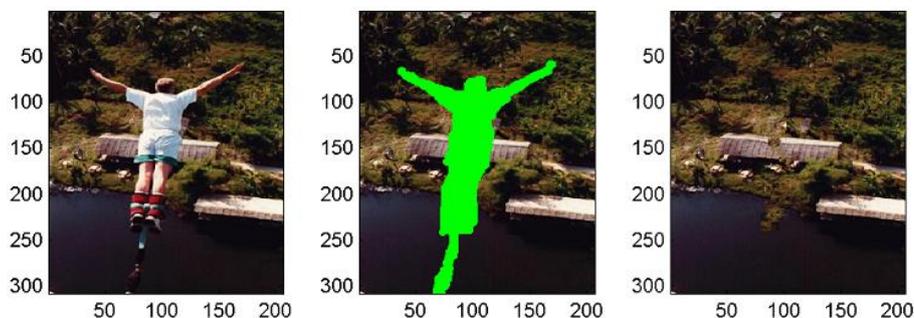


FIGURE 5.1 – Image originale, régions à remplir et résultat de l'“inpainting” [Criminisi 2004]

Dans la section suivante, nous allons présenter quelques exemples de méthodes d'“inpainting” (les plus classiques) pour chaque type d'approches.

5.3 État de l'art

5.3.1 “Inpainting” de régions contenant des structures

La méthode proposée par l'auteur [Bertalmio 2000] et que nous allons présenter dans cette partie est destinée à remplir des régions ayant comme fond des structures (contours tels que des lignes ou des courbes). Pour cela nous propageons d'une manière lisse l'information à partir des zones voisines. L'algorithme utilisé a la forme itérative suivante :

$$I_{t+1}(x, y) = I_t(x, y) + A\hat{I}_t(x, y), \forall (x, y) \in \Omega \quad (5.1)$$

avec t le temps, A le taux d'amélioration, \hat{I}_t est “l'amélioration” de $I_t(x, y)$ que nous avons besoin de déterminer et Ω la région à remplir.

Notons la ressemblance avec l'équation de diffusion vue précédemment (voir chapitre 2). En effet, cette équation correspond à une façon de mettre en œuvre l'équation de diffusion.

Notons que l'algorithme n'est appliqué que dans les régions à remplir.

Soit $L_{t(x,y)}$ l'information à propager et $\overrightarrow{N_t(x,y)}$ la direction de propagation telle que :

$$\hat{I}_t(x, y) = \overrightarrow{\delta L_{t(x,y)}} \cdot \overrightarrow{N_t(x,y)} \quad (5.2)$$

avec $\overrightarrow{\delta L_{t(x,y)}}$ une mesure du changement d'information (gradient par exemple).

Comme nous voulons que la propagation soit lisse, $L_t(x, y)$ doit être un estimateur lisse. Nous considérons un estimateur Laplacien : $L_t(x, y) := \Delta L_t(x, y)$.

La direction de propagation peut être définie comme la normale aux bords $\partial\Omega$ de la région Ω comme le montre la figure 5.2. Cependant, cela ne donne pas toujours de bons résultats à cause des formes arbitraires que peuvent prendre les bords $\partial\Omega$ (voir figure 5.2).

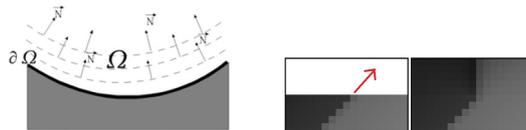


FIGURE 5.2 – Défauts induit par le choix de la direction de propagation normale aux bords. Le vecteur en rouge correspond à la direction des isophotes.

Une meilleure solution est de choisir comme direction $\overrightarrow{N_t(x, y)}$, la direction des lignes de contours arrivant à $\delta\Omega$ et ayant les mêmes valeurs de niveau de gris (flèche rouge sur la figure 5.2) ou ce que nous appelons des “isophotes”. Pour cela, nous considérons une rotation du gradient qui nous donne la direction de la plus petite variation spatiale qui correspond à la direction des lignes de contours :

$$\overrightarrow{N_{t(x,y)}} = \nabla^\perp I_t(x, y) \quad (5.3)$$

Notons que la direction varie dans le temps ce qui donne de meilleurs résultats que si la direction est fixée dès le départ en permettant de reconnecter les contours arrivant aux bords $\delta\Omega$.

Pour une meilleure détection des isophotes nous appliquons à toute l'image une diffusion (gaussienne ou Perona et Malik par exemple) afin de réduire la quantité de bruit. Après quelques itérations de l'algorithme d'“inpainting” décrit ci-dessus nous appliquons une nouvelle diffusion et ainsi de suite.

Dans le cas d'images couleurs l'algorithme est appliqué de façon indépendante sur chaque canal.

La méthode requiert environ 5 minutes pour effectuer l'“inpainting” d'images couleurs. Le temps de traitement dépend aussi de la taille des régions à retoucher.

Les figures 5.3 et 5.4 montrent les résultats que nous pouvons obtenir avec cette méthode :

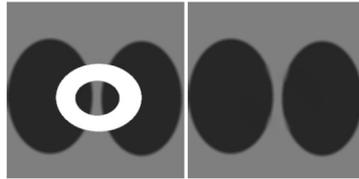


FIGURE 5.3 – Résultat d'“inpainting” avec la direction de propagation suivant les lignes de contours

Dans ce qui suit nous allons présenter une méthode utilisant la diffusion des informations spatiales permettant d'effectuer l'“inpainting” d'images d'une façon rapide.

5.3.2 “Inpainting” rapide par diffusion isotropique

L'algorithme [Richard 2001] que nous allons présenter dans cette section est destiné à l'“inpainting” de régions locales ayant de petites tailles en utilisant un modèle simple et rapide. L'hypothèse faite par l'auteur est que le système visuel

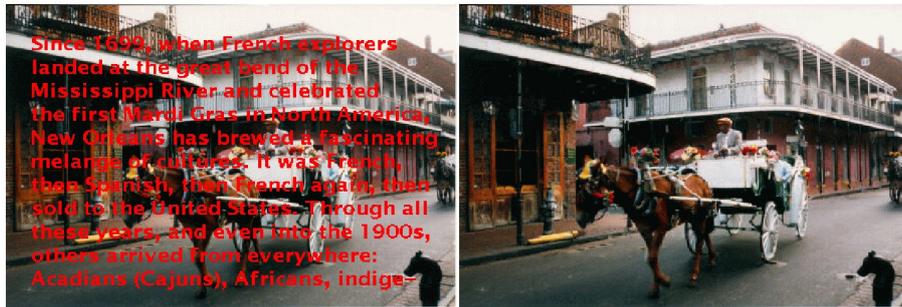


FIGURE 5.4 – Élimination du texte d'une image [Bertalmio 2000].

humain peut tolérer une certaine quantité de flou dans des régions ne contenant pas des contours à fort gradient.

Une version simple de la méthode est de convoluer les régions à remplir avec un noyau de diffusion de manière itérative. Nous pouvons utiliser une diffusion isotropique : noyau gaussien dont la valeur du pixel central est nulle. L'auteur [Richard 2001] propose les noyaux suivants :

$$\begin{pmatrix} a & b & a \\ b & 0 & b \\ a & b & a \end{pmatrix} \quad \begin{pmatrix} c & c & c \\ c & 0 & c \\ c & c & c \end{pmatrix}$$

FIGURE 5.5 – Noyaux de diffusion utilisées par l'auteur [Richard 2001].
 $a = 0,073235$ $b = 0,176765$ $c = 0,125$.

L'algorithme permet de restaurer l'images en quelques secondes (entre 300 ms et 4 sec) mais n'est pas aussi robuste aux régions contenant des structures (contours) que l'est le processus de diffusion anisotropique présenté dans la section précédente. La figure 5.6 montre le résultat obtenu par cette méthode. Nous remarquons que dans les zones à forts gradients nous introduisons du flou.

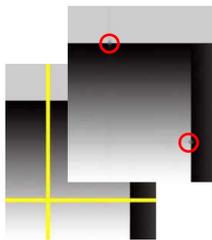


FIGURE 5.6 – Dans l'image à gauche les régions à remplir sont en jaunes et le résultat d'"inpainting" [Richard 2001] à droite après 100 itérations

Pour se rapprocher d'une diffusion anisotropique l'auteur propose que l'utilisateur spécifie de manière interactive la barrière des contours arrivant à $\delta\Omega$ afin de stopper la diffusion dans ces zones. Les figures 5.7 et 5.8 montrent le résultat obtenu

par cette méthode.

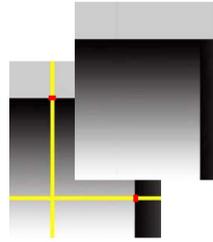


FIGURE 5.7 – Dans l'image à gauche les régions à remplir sont en jaunes et les barrières de diffusion en rouges spécifiées par l'utilisateur et le résultat d'"inpainting" [Richard 2001] à droite après 100 itérations.

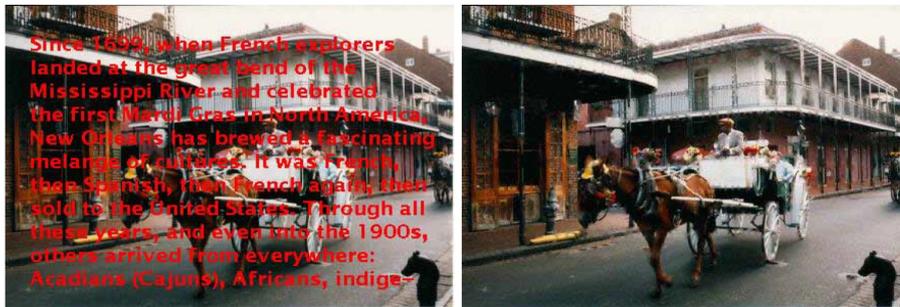


FIGURE 5.8 – Résultats d'"inpainting" [Richard 2001] sur l'image précédemment présentée dans la figure 5.4

Le problème avec les méthodes de diffusion est qu'elles ne sont pas efficaces dans le cas où la région à remplir est censée contenir des textures. Nous allons présenter maintenant des méthodes utilisant la synthèse de texture pour effectuer l'"inpainting".

5.3.3 Synthèse de texture

L'"inpainting" par synthèse de texture consiste en deux étapes :

1. Recherche du meilleur bloc correspondant à la région à remplir en utilisant la similarité des textures.
2. Coller le bloc sélectionné dans la région à remplir à une position précise.

Le problème de ce type de méthode est que la recherche du bloc en plus d'être coûteuse en temps de calcul peut parfois induire des erreurs qui vont se propager aux régions voisines et le résultat sera inacceptable.

La recherche des régions peut être effectuée en cherchant dans toute l'image les blocs similaires avec une grande priorité aux isophotes [Criminisi 2004] afin de préserver les structures (même hypothèse que la méthode d'"inpainting" par diffusion

anisotropique dans le chapitre 2). La figure 5.9 montre un exemple illustrant cette méthode.

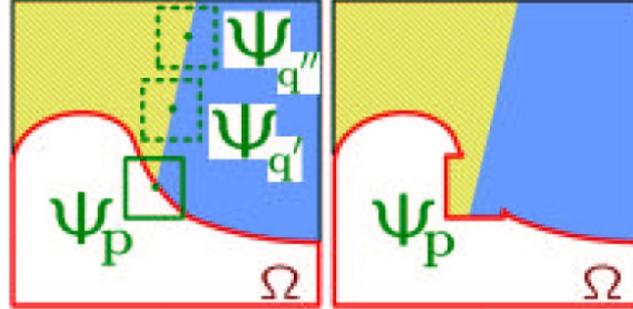


FIGURE 5.9 – Image de gauche montrant la recherche du bloc Ψ_p à remplir dans les régions de l'image contenant des isophotes. $\Psi_{q'}$ et $\Psi_{q''}$ sont deux blocs candidats et à droite le résultat de l'“inpainting”

La sélection du bloc utilisé est effectuée en mesurant une priorité pour chaque bloc de la façon suivante :

$$P(p) = C(p)D(p) \quad (5.4)$$

avec p la position du pixel central du bloc à remplir et

$$C(p) = \frac{\sum_{q \in \Psi_p \cap \bar{\Omega}} C(q)}{|\Psi_p|} \quad (5.5)$$

est la confiance qui indique la fiabilité du bloc candidat, $|\Psi_p|$ est l'aire du bloc Ψ_p , Ω la région à remplir. $C(p)$ est initialisé tel que $C(p) = 0$ dans les régions à remplir et $C(p) = 1$ dans le reste de l'image.

$$D(p) = \frac{|\nabla I_p^\perp \cdot n_p|}{\alpha} \quad (5.6)$$

est le terme des données qui donne la priorité aux isophotes, α un facteur de normalisation égal à 255 pour une image en niveaux de gris 8 bits, n_p est la normale aux bord $\delta\Omega$ et ∇I_p^\perp l'isophote comme le montre la figure 5.10.

Afin de réduire la zone de recherche l'auteur [2004] propose une méthode utilisant les champs de Markov aléatoire (MRF) permettant de limiter la recherche aux régions voisines.

Pour cela l'image est segmentée en plusieurs régions en fonction de la similitude de texture des régions. L'auteur [Tang 2005] propose d'utiliser un algorithme du “mean shift” [Comaniciu 2002]. Durant le processus d'“inpainting” la recherche est limitée aux régions qui intersectent avec le bloc à remplir. Cet algorithme restaure bien les blocs qui appartiennent à une seule région de segmentation. Dans le cas contraire la restauration peut induire des erreurs.

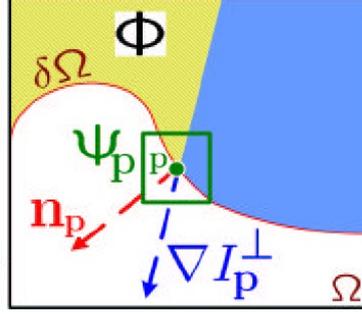


FIGURE 5.10 – Direction des isophotes ∇I_p^\perp , Φ étant la zone de recherche

Une mesure de cohérence permet de sélectionner le meilleur bloc parmi tous les blocs candidats. Cette mesure utilise les pixels voisins les plus proches en les faisant voter comme nous allons le montrer.

Considérons, comme sur la figure 5.10, Φ la région source. Nous mesurons la distance moyenne MD_Φ de la région Φ comme suit :

$$MD_\Phi = \sum_{i \in [0, N]} \frac{\|q_i, c_{q_i}\|}{N} \quad (5.7)$$

avec q_i un point de Φ , c_{q_i} la position du bloc à copier dans la région source, N le nombre de pixels dans Φ et $\|*\|$ la distance euclidienne.

D'un autre côté nous mesurons la distance moyenne de la région à remplir $\Psi - \Phi$ tel que :

$$MD_{\Psi_p - \Phi} = \sum_{i \in [0, M]} \frac{\|p_i, c_{p_i}\|}{M} \quad (5.8)$$

avec $p_i \in \Psi - \Phi$ et c_{p_i} son correspondant dans le bloc candidat et M le nombre de point dans $\Psi - \Phi$.

La confiance se calcule de la façon suivante :

$$C_{\Psi_p} = \frac{1}{|MD_\Phi - MD_{\Psi_p - \Phi}|} \quad (5.9)$$

Nous sélectionnons donc le bloc qui maximise cette confiance :

$$\operatorname{argmax}_{\Psi_q} C_{\Psi_q} \quad (5.10)$$

Afin de réduire les discontinuités qui peuvent encore exister, les Graph-cut [Efros 2001, Kwatra 2003] peuvent être utilisés afin de sélectionner le bloc qui pourra rejoindre les régions précédemment remplies de la meilleure façon.

La figure 5.11 montre les résultats obtenus par cette méthode.

Dans le cas où nous avons accès à une vidéo l'information temporelle peut être très utile. Nous allons présenter, dans la suite, des méthodes utilisant une séquence d'images afin de récupérer l'information manquante.

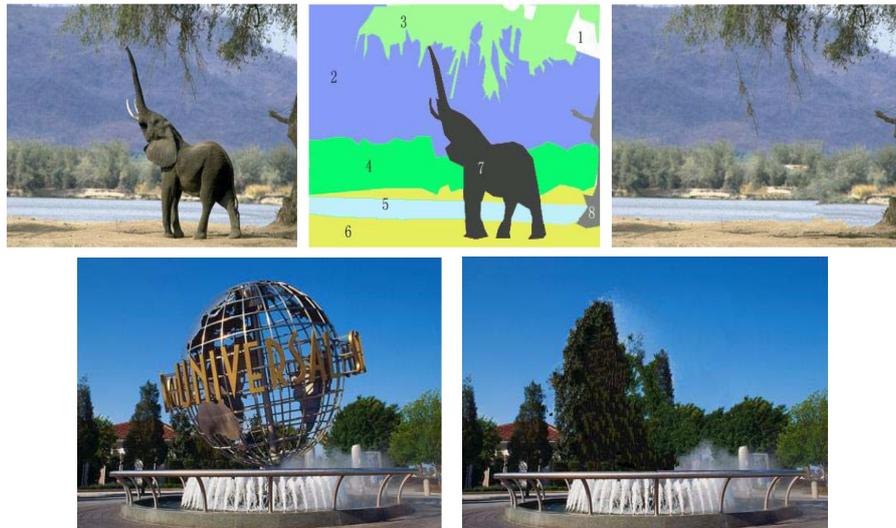


FIGURE 5.11 – Première ligne : image originale (à gauche), résultat de la segmentation en régions ayant des textures similaires (au milieu) et résultat de l’“inpainting” [Tang 2005] (à droite). Deuxième ligne : image originale (à gauche) et résultat de l’“inpainting”.

5.3.4 “Inpainting” vidéo

L’auteur [Patwardhan 2005] a proposé une méthode utilisant une caméra immobile, qui en supposant que la scène est essentiellement constituée de fond stationnaire et de quelques objets en mouvement, permet l’“inpainting” de régions appartenant au fond et cachées par les objets en mouvement. Une variante de cette méthode proposée par le même auteur permet de reconstruire des objets en mouvement partiellement cachés en supposant que le mouvement est globalement consistant.

Pour cela, nous détectons d’abord si un pixel est en mouvement ou non à l’aide du flot optique. Ceci va permettre d’éliminer automatiquement les objets en mouvement. Nous pouvons aussi spécifier la zone à remplir.

Les fonds stationnaires sont reconstruits dans un premier temps à partir des régions non endommagées (pixels considérés comme stationnaires) en les copiant dans les régions endommagées (pixels non stationnaires). Pour cela, la mesure de priorité [Criminisi 2004] que nous avons présentée dans la section précédente est utilisée. Toutefois, il reste des pixels endommagés formant un trou dans toute les trames de la vidéo. Ces pixels sont remplis en utilisant l’information spatiale à l’aide de la même mesure de priorité des blocs présentée dans [Criminisi 2004]. La figure 5.12 montre le résultat obtenu sur 4 trames d’une vidéo.

Dans la suite, nous allons présenter le type d’approches sélectionné (EDP de diffusion). Deux méthodes plus évoluées que celles que nous avons vues dans cette section seront présentées. La méthode la plus pertinente parmi ces deux méthodes a été retenue et adaptée à notre problème.



FIGURE 5.12 – Première ligne : 4 trames contenant des objets en mouvement et des objets stationnaires. Les objets stationnaires et le corps humain en mouvement à éliminer sont encadrés en rouge. Deuxième ligne : résultat de l’“inpainting” [Patwardhan 2005].

5.4 Méthodes sélectionnées

Dans le cas d’images routières, les parties à reconstruire sont constituées essentiellement de structures géométriques : courbe de la route, marquages et panneaux. Les zones texturées telles que les arbres nous intéressent moins. Nous avons donc testé 3 méthodes d’“inpainting” par EDP : la méthode [Richard 2001] présentée dans la section 5.3.2 et 2 autres méthodes plus évoluées que nous allons présenter dans la suite. Ces méthodes EDP ont l’avantage d’être plus rapides que des méthodes utilisant la vidéo et posent moins de problèmes lorsque la caméra et les objets de la scène sont mobiles. En plus, elles sont faciles à mettre en œuvre.

La méthode [Richard 2001] utilisant un lissage isotropique rapide (voir section 5.3.2), comme nous l’avons vu, génère du flou dans les parties à reconstruire (voir figure 5.13) .



FIGURE 5.13 – De gauche à droite : image originale avec les zones à remplir en blanc et image restaurée par diffusion isotropique rapide [Richard 2001] : convolutions successives par un noyaux gaussien

Nous avons donc sélectionné deux méthodes de diffusion par EDP plus récentes et plus évoluées. La première méthode que nous avons exploitée [Telea 2004] permet

de restaurer l'image sans générer de flou. La deuxième méthode [Bornemann 2007] est une amélioration de la première. Elle permet de mieux restaurer les structures géométriques. Nous allons présenter dans la suite ces deux méthodes.

5.4.1 Algorithme de Telea

Cette méthode [Telea 2004] utilise le principe de diffusion expliqué précédemment dans le chapitre 2 (débruitage). Considérons Ω la région à remplir, $\partial\Omega$ les bords de Ω , p le pixel à remplir et $B_\varepsilon(p)$ un voisinage circulaire de p de taille ε . L'“inpainting” de p consiste à diffuser l'information non dégradée de $B_\varepsilon(p)$ vers p . La figure 5.14 illustre le principe d'“inpainting”.

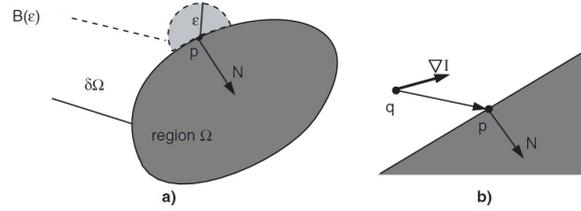


FIGURE 5.14 – Principe de l'“inpainting” : (a) Propagation de l'information de la zone B_ε/Ω vers le pixel p de la région Ω à remplir. (b) gradient $\nabla I(q)$ au point q , N normale au bords et $(p - q)$ le vecteur \mathbf{qp}

Pour un point $q \in B_\varepsilon(p)/\Omega$ sa contribution $I_q(p)$ dans le remplissage de p s'écrit, à partir de l'équation de diffusion en faisant une approximation du première ordre et en considérons ε assez petit, sous la forme suivante :

$$I_q(p) = I(q) + \nabla I(q)(p - q) \quad (5.11)$$

La moyenne de l'ensemble des points de $B_\varepsilon(p)/\Omega$ nous donne :

$$I(p) = \frac{\sum_{q \in B_\varepsilon(p)/\Omega} w(p, q)[I(q) + \nabla I(q)(p - q)]}{\sum_{q \in B_\varepsilon(p)/\Omega} w(p, q)} \quad (5.12)$$

avec $w(p, q)$ une fonction de poids qui doit être choisie de façon à propager les niveaux de gris ainsi que les détails des formes à reconstruire.

Les poids $w(p, q)$ sont construits de la façon suivante :

$$w(p, q) = \text{dir}(p, q) \text{dst}(p, q) \text{lev}(p, q) \quad (5.13)$$

tel que $\text{dir}(p, q)$ est une composante directionnelle qui garantit que la contribution des pixels proches du vecteur unitaire $N = \nabla T$ (normal au bord) pointant dans la direction de progression de l'“inpainting” est plus grande que pour les pixels les plus éloignés de N . Ce terme se calcule comme suit :

$$dir(p, q) = \frac{p - q}{\|p - q\|} N(p) \quad (5.14)$$

La distance géométrique $dst(p, q)$ augmente la contribution des pixels proches spatialement de p .

$$dst(p, q) = \frac{1}{\|p - q\|^2} \quad (5.15)$$

Enfin la composante $lev(p, q)$ assure que les pixels proches du contour traversant p contribuent plus que les pixels éloignés. En notant T la distance aux bords $\partial\Omega$ nous avons :

$$lev(p, q) = \frac{1}{1 + |T(p) - T(q)|} \quad (5.16)$$

Notons que pour des tailles de voisinage ε très petites (environ 6 pixels) la contribution de dst et lev est faible. Pour des tailles ε grandes l'utilisation des 3 composantes dir , dst et lev donne de meilleurs résultats visuels.

La valeur choisie pour ε doit être comprise entre 3 et 10 pixels, en considérant des régions à remplir ayant une épaisseur d'environ 15 pixels. Notons que plus ε est grand plus nous allons introduire du flou.

Dans le cas d'images couleur, l'algorithme est appliqué sur chaque canal séparément.

Afin d'effectuer l'"inpainting" rapidement, chaque pixel de Ω est parcouru une seule fois. L'ordre de parcours des pixels est déterminé par leur distance euclidienne aux bords $T(x) = dist(x, \partial\Omega)$ avec $x \in \Omega$. Nous commençons donc par remplir les pixels les plus proches des bords. La distance aux bords est estimée par un algorithme dit de marche rapide ou FMM [Sethian 1999].

La figure 5.15 montre les résultats obtenus avec cette méthode sur l'image précédemment présentée. Par rapport à la figure 5.13 la méthode de Telea introduit moins de flou mais dégrade les contours.



FIGURE 5.15 – "Inpainting" par la méthode de Telea

Nous appelons la courbe qui partage Ω en deux parties à distances aux bords égales le squelette (affiché en rouge sur la figure 5.16).

Le défaut de la méthode de Telea est qu'elle propage l'information seulement dans la direction donnée par le domaine à remplir Ω et par conséquent, elle ne permet pas le transport d'information à travers le squelette Σ de Ω sans introduire du flou (voir figure 5.16).

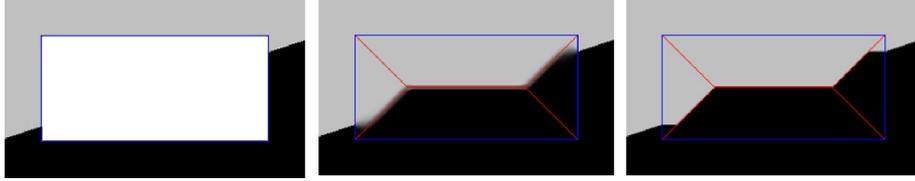


FIGURE 5.16 – Image originale (à gauche) et résultat obtenu avec l'algorithme de Telea tel que $\varepsilon = 6$ (image du milieu) et ensuite $\varepsilon = 1$ (image à droite). Le squelette est affiché en rouge et les bords $\partial\Omega$ en bleu. Nous constatons que la méthode introduit du flou lorsqu'elle tente de propager l'information à travers le squelette.

La méthode [Bornemann 2007], que nous allons présenter dans la suite, permet de résoudre ce problème en utilisant un tenseur de diffusion prenant en compte la direction des contours qui arrivent vers Ω et permet de contrôler en même temps la force de diffusion.

5.4.2 Algorithme de Bornemann

Dans cette section, nous présentons une méthode d'“inpainting” rapide [Bornemann 2007] qui est une version améliorée de la méthode précédente. Nous modifions les poids de la méthode de Telea de façon à obtenir un résultat de meilleure qualité visuelle : les contours arrivant vers Ω sont reconnectés d'une meilleure façon en diffusant à travers le squelette Σ de Ω .

Pour cela, un tenseur de diffusion nous donne une meilleure direction de diffusion ainsi qu'une force de diffusion qui permet de contrôler la quantité d'information à diffuser. Nous introduisons aussi deux paramètres de lissage (de l'image et des directions détectées) afin de mieux contrôler l'information propagée (moins de bruit mais dégrade les contours) et la direction de diffusion (diffusion à travers le squelette dès le départ). Une illustration des résultats obtenus pour les différents paramètres de l'algorithme, que nous allons présenter dans la suite, est en annexe A.1.

Pour résoudre le problème de la méthode de Telea nous devons prendre en compte les contours arrivant vers Ω . Ceci se fait en utilisant un vecteur c de Ω modélisant les isophotes. Le vecteur c peut être choisi comme étant aligné avec les lignes de niveau (isophotes) comme nous l'avons présenté dans la section 5.3.1 de ce chapitre (voir équation 5.3), nous avons donc :

$$\vec{c} = \nabla^\perp \Delta I_\sigma \quad (5.17)$$

avec $I_\sigma = K_\sigma * I$, K_σ étant un noyau gaussien : $K_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{|x|^2}{2\sigma^2}}$.

Flux de détection des contours

Nous appelons le flux généré par l'équation 5.17 (voir figure 5.17) le “flux de détection des contours” (“edge detection flow”). Notons que ce type de flux est utilisé pour des applications telles que la reconnaissance des empreintes digitales. Toutefois, cette méthode, comme le montre la figure 5.17, souffre d'un problème de perte de cohérence avec l'information contenue dans l'image originale.

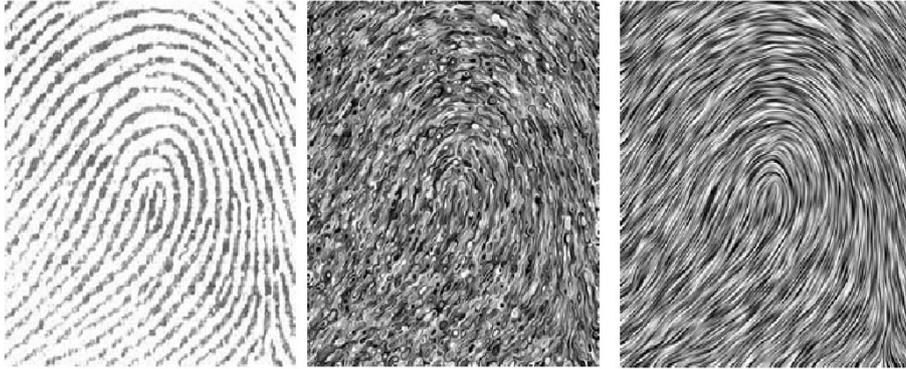


FIGURE 5.17 – De gauche à droite : image originale (empreintes digitales), flux de détection des contours tel que $\sigma = 1,5$ pixel et flux de cohérence tel que $\sigma = 1,5$ pixel et $\rho = 4$ pixels.

Le flux de détection de contours n'est pas très robuste : en utilisant le flux de détection des contours comme terme de la la fonction de poids w (équation 5.18) nous obtenons un résultat médiocre. Nous allons voir dans la suite une méthode permettant d'améliorer le résultat.

Rappelons tout d'abord la formule utilisée pour effectuer l'“inpainting” et qui est la même que dans la méthode de Telea :

$$I(p) = \frac{\sum_{q \in B_\varepsilon(p)/\Omega} w(p, q) [I(q) + \nabla I(q)(p - q)]}{\sum_{q \in B_\varepsilon(p)/\Omega} w(p, q)} \quad (5.18)$$

Nous montrons dans ce qui suit la fonction de poids utilisée.

Fonction des poids

Considérons un pixel y non dégradé (de valeur connue) se trouvant dans le voisinage d'un pixel x à remplir. La fonction de poids proposée par l'auteur [Bor-nemann 2007] est :

$$w(x, y) = \sqrt{\frac{\pi}{2}} \frac{\mu}{|x - y|} \exp\left(-\frac{\mu^2}{2\varepsilon} | -c^\perp \cdot (x - y) |^2\right) \quad (5.19)$$

avec μ un coefficient qui contrôle la force de diffusion. La valeur de chaque pixel de Ω sera modifiée à partir des équations 5.18 et 5.19.

Flux de cohérence

Il n'est pas conseillé d'utiliser le flux de détection des contours avec la fonction des poids à cause des défauts dont souffre cette méthode. Une façon plus robuste permettant de déterminer la direction de cohérence est l'approche proposée dans [Weickert 1998, Weickert 2003] qui consiste à utiliser un tenseur de structure (voir diffusion anisotropique dans le chapitre 2) qui permet d'obtenir, en plus de la direction, une mesure de la force de cohérence qui va nous permettre d'adapter le paramètre μ aux caractéristiques locales de l'image.

Le tenseur de structure J_ρ d'une image I est la matrice 2×2 symétrique semi définie positive tel que :

$$J_\rho(\nabla I_\sigma) = K_\rho * (\nabla I_\sigma \otimes \nabla I_\sigma) \quad (5.20)$$

avec K_ρ un noyau gaussien. Les valeurs propres du tenseur $0 \leq \lambda_1(x) \leq \lambda_2(x)$ correspondent aux vecteurs propres w_1 et w_2 .

La direction de cohérence est donc le vecteur propre correspondant à la plus petite valeur propre :

$$c = w_1 \quad (5.21)$$

Cette direction est une version lisse des directions des isophotes $\nabla^\perp I_\sigma$.

Nous appelons le flux correspondant "flux de cohérence" (coherence flow).

La figure 5.17 montre le résultat du flux de cohérence calculé sur l'image d'empreintes digitales. Nous constatons une grande différence par rapport au résultat obtenu par le flux de détection des contours.

La mesure de la force de cohérence μ se calcule selon l'équation 5.23 proposée par [Weickert 1998].

$$\mu = \begin{cases} 1, & \text{si } \lambda_1(x) = \lambda_2(x) \\ 1 + \kappa \exp\left(\frac{-\delta_{quant}^4}{(\lambda_1(x) - \lambda_2(x))^2}\right), & \text{sinon} \end{cases} \quad (5.22)$$

Le paramètre κ est un coefficient qui permet de contrôler la netteté, δ_{quant} est la résolution de la quantification qui correspond à la différence entre deux niveaux de gris successifs et qui, dans notre cas, est choisie égale à 1.

Une grande valeur de μ permet la diffusion d'une petite quantité d'information alors qu'une petite valeur permet la diffusion d'une grande quantité d'information.

Effets de bord et flux de cohérence modifié

Le problème avec le lissage par les gaussiennes K_σ et K_ρ est qu'il atteint les zones à remplir, alors que nous devons utiliser seulement les pixels à l'extérieur de Ω pour effectuer l'"inpainting". Une solution simple consiste à compléter l'image par des zéros dans les endroits à remplir. Nous obtenons alors le tenseur suivant :

$$J_\rho(\nabla(\mathbb{1}_{\Omega(x)}I)_\sigma)(x) \quad (5.24)$$

avec $\mathbb{1}_{\Omega(x)}$ la fonction indicatrice de $\Omega(x)$ qui prend la valeur 1 dans le domaine Ω et 0 sinon. Le problème, là encore, est que cela va aligner le flux de cohérence aux bords $\partial\Omega$ en les considérant comme des contours comme le montre la deuxième image de la figure 5.18. Nous allons donc retrouver le problème de diffusion suivant la normale aux bords comme avec l'algorithme de Telea.

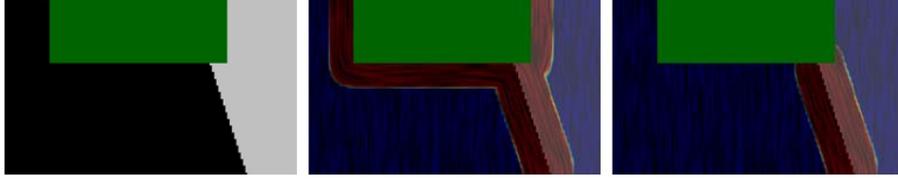


FIGURE 5.18 – Image originale avec la zone à remplir en vert, flux de cohérence (fortes cohérences sont affichées en marron et les faibles cohérences en bleu) et flux de cohérence modifié

L'auteur [Bornemann 2007] trouve expérimentalement que, en remettant à l'échelle la version lisse de $\mathbb{1}_{\Omega(x)}I$ (voir équation 5.26) avant le calcul des gradients nous arrivons à surmonter ce problème. Nous utilisons donc le tenseur de structure modifié suivant :

$$J_{\sigma,\rho}(x) = \frac{(K_\rho * (\mathbb{1}_{\Omega(x)}\nabla\nu_\sigma \otimes \nabla\nu_\sigma))(x)}{(K_\rho * \mathbb{1}_{\Omega(x)})(x)} \quad (5.25)$$

avec

$$\nu_\sigma = \frac{K_\sigma * (\mathbb{1}_{\Omega(x)}I)}{K_\sigma * \mathbb{1}_{\Omega(x)}} \quad (5.26)$$

La dernière image de la figure 5.18 montre le flux de cohérence obtenu avec cette méthode. Nous constatons une nette amélioration sur les bords par rapport au structure de tenseur initial.

Implantation

La partie de l'algorithme qui consomme le plus de temps de traitement est le calcul du tenseur de structure. Pour surmonter ce problème l'auteur [Bornemann 2007] propose une Implantation qui permet de mettre à jour le tenseur pour un pixel qui vient d'être rempli.

Nous commençons par mettre à jour ν_σ . Soit x un pixel qui vient juste d'être rempli et y le prochain pixel à remplir déterminé par l'algorithme FMM de la

même façon que dans la section précédente : nous commençons par remplir les pixels les plus proches du bord. Notons $\hat{\nu}_\sigma$ et χ_σ , respectivement les numérateur et dénominateur de ν_σ . Les formules de mise à jour permettant de calculer ces deux termes sont les suivantes :

$$\hat{\nu}_\sigma(y) = \hat{\nu}_\sigma(y) + K_\sigma(x - y)I(x) \quad (5.27)$$

et

$$\chi_\sigma(y) = \chi_\sigma(y) + K_\sigma(x - y) \quad (5.28)$$

avec y un pixel appartenant au voisinage quadratique de x .

De la même façon, en notant \hat{J}_ρ et χ_ρ , respectivement les numérateur et dénominateur de J , nous obtenons les nouvelles valeurs du tenseur :

$$\hat{J}_\rho(y) = \hat{J}_\rho(y) + K_\rho(x - y)(\nabla \nu_\sigma \otimes \nabla \nu_\sigma) \quad (5.29)$$

et

$$\chi_\rho(y) = \chi_\rho(y) + K_\rho(x - y) \quad (5.30)$$

Images couleurs

L'algorithme décrit ci-dessus est applicable aux images en niveau de gris. Une manière simple pour traiter les images couleurs est d'appliquer l'algorithme sur chaque canal séparément. Cependant nous allons avoir un problème de contours qui se forment à des endroits différents sur chaque canal. Une façon de surmonter ce problème est d'utiliser un tenseur commun aux trois couleurs calculé comme une combinaison des tenseurs de chaque canal R, G et B :

$$\hat{J}_\rho = 0,2999\hat{J}_\rho | R + 0,587\hat{J}_\rho | G + 0,114\hat{J}_\rho | B \quad (5.31)$$

La figure 5.19 montre le résultat d'"inpainting" sur une image précédemment présentée. Nous observons que, par rapport à la méthode de Telea 5.15, nous arrivons à mieux reconstruire les marquages et les différentes structures de la route.

Dans la suite nous allons voir comment nous choisissons les paramètres de la méthode.

Choix des paramètres

Nous rappelons les 4 paramètres de l'algorithme :

- ε la taille du voisinage considéré (rayon du cercle B_ε) du pixel à remplir.
- κ le paramètre de netteté qui permet de contrôler la quantité d'information à diffuser. Pour utiliser des grandes valeurs de κ nous devons utiliser un grand rayon ε .
- σ et ρ les écarts types des gaussiennes qui correspondent à l'échelle où les détails seront négligés pour le calcul de la direction du flux.

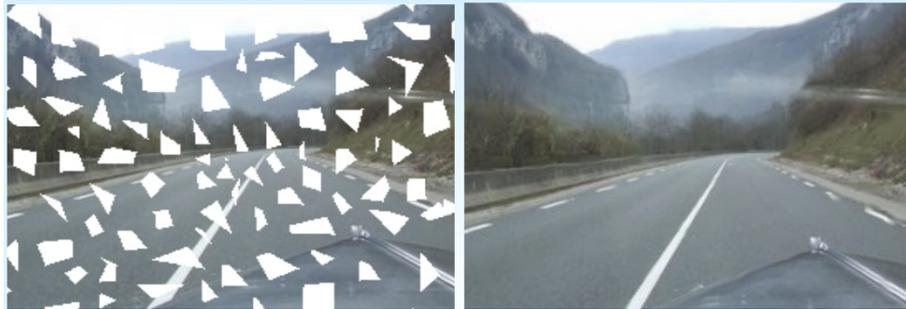


FIGURE 5.19 – Image originale avec les zones à remplir en blanc et le résultat d’“inpainting”. Par rapport à la méthode de Telea, nous obtenons une meilleure reconstruction des différentes structures de l’image

Dans notre cas nous avons ajouté un nouveau paramètre qui est la taille T_g des gaussiennes : au lieu d’utiliser une taille de $4 * \sigma$ ou $4 * \rho$ nous choisissons une taille différente. Nous pouvons aussi ne pas effectuer de lissage en choisissant $T_g = 0$.

Le choix des paramètres va dépendre du résultat final souhaité, du type de voisinage de la région à remplir (texture, structure ou uniforme), de la taille et de la forme du masque.

Plus les tailles des gaussiennes sont grandes, plus nous pouvons récupérer des informations loin du pixel à remplir.

La forme du masque joue aussi un rôle important dans le choix de la taille en fonction du résultat souhaitée (forme des structures à reconstruire).

Lorsque nous connaissons déjà la forme que doit avoir les structures dans l’image, nous faisons varier la taille du lissage de façon à avoir la direction et la quantité d’information diffusée permettant d’obtenir l’image désirée.

En annexe A.1, nous illustrons ceci par des exemples qui montrent l’impact des différents paramètres sur le résultat.

Notons que l’auteur [Bornemann 2007] a précisé que pour certaines images il a été obligé d’intervenir manuellement pour obtenir un résultat satisfaisant.

Comme, dans notre cas, nous n’avons aucun a priori sur les formes à reconstruire, nous avons appliqué l’algorithme du recuit simulé sur une base d’images de routes afin de trouver les paramètres optimaux.

5.5 Optimisation des paramètres sur des images routières

Dans notre cas, nous voulons que l’“inpainting” soit fait automatiquement sur une vidéo acquise à l’aide d’une caméra mobile. Le choix de la taille du lissage n’est pas une tâche facile à cause des grandes variations des gradients et des couleurs de l’image. De plus, nous n’avons pas d’a priori sur les formes à reconstruire (gouttes

de différentes tailles et formes).

Nous avons donc utilisé l’algorithme du recuit simulé, de la même façon que pour la méthode de restauration d’images de brouillard, afin de sélectionner les meilleurs paramètres qui minimisent l’erreur absolue moyenne entre l’image originale non dégradée (vérité terrain) et l’image restaurée. Après l’étude de sensibilité de chacun des paramètres nous trouvons que la meilleure configuration est la suivante :

$$\varepsilon = 3, \kappa = 7, 8, \sigma = 0, 5 \text{ et } \rho = 2, 7.$$

La figure 5.20 montre le résultat obtenu sur la base de donnée d’apprentissage en utilisant les paramètres déduits par le recuit simulé. Les images présentées sur cette figure ont une taille 384×288 , le temps de traitement est d’environ 160 ms par image.

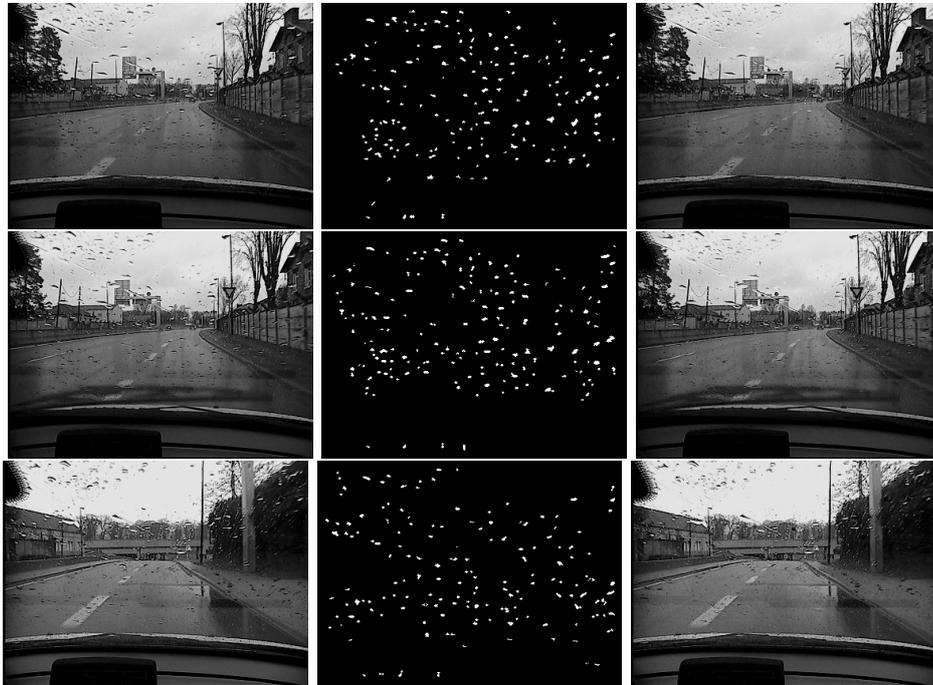


FIGURE 5.20 – De gauche à droite : image originale, masque des gouttes et résultats d’“inpainting” tel que $\varepsilon = 3, \kappa = 7, 8, \sigma = 0, 5$ et $\rho = 2, 7$

La figure 5.21 montre le résultat obtenu sur des images couleurs. La taille du masque est plus grande que dans le cas précédent. Nous sommes donc contraints d’adapter les paramètres à ce type d’images (les mêmes paramètres déduits par le recuit simulé donnent un résultat moins bon).

La figure 5.22 montre des images de pluie et le résultat de l’“inpainting” sur les gouttes détectées.

Une autre application de l’algorithme que nous avons trouvée intéressante est l’élimination de bruit poivre et sel lorsqu’il est important (voir annexe A.2).

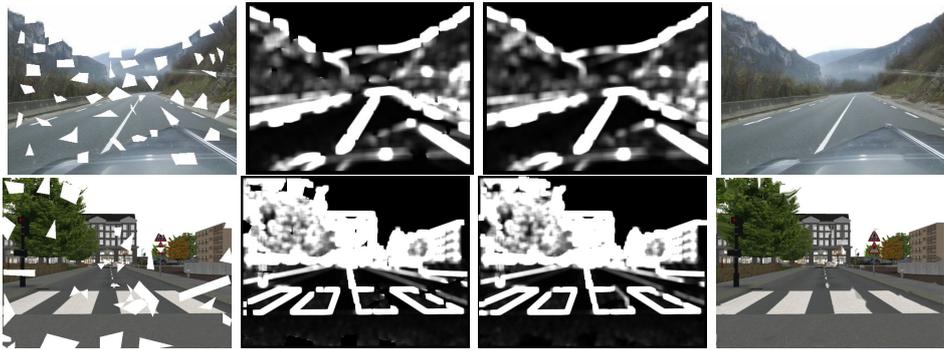


FIGURE 5.21 – Image originale, image avec zone à remplir en blanc, flux de cohérence initial, flux de cohérence final et restauration tel que $\varepsilon = 9$, $\kappa = 70$, $\sigma = 1,4$ et $\rho = 7$.



FIGURE 5.22 – Image de pluie et restauration tel que $\varepsilon = 9$, $\kappa = 70$, $\sigma = 1,4$ et $\rho = 7$. Première ligne : environ 60 % de données manquantes. Deuxième et troisième lignes : environ 10 % de données manquantes.

5.6 Conclusion

Afin de restaurer les images dégradées par les gouttes focalisées sur le pare-brise, nous avons proposé d'utiliser une méthode d'"inpainting" par diffusion anisotrope fondée sur les EDP. Comparé à d'autres types d'approches, l'"inpainting" par EDP est plus adapté à notre problématique. En effet, il restaure au mieux les structures de l'image en propageant localement l'information. Une méthode des plus robustes [Bornemann 2007] a été testée et retenue pour sa rapidité et sa facilité d'implantation. Cette méthode a été évaluée qualitativement à l'aide d'images réelles et d'images de synthèse. Les résultats obtenus sont de bonne qualité : restauration quasi-parfaite jusqu'à 10 % de zones dégradées dans l'image et résultat acceptable dans le cas où la dégradation atteint 60 % de l'information disponible dans l'image. En utilisant le recuit simulé l'algorithme a été optimisé sur des images routières acquises par caméra embarquée.

Notons que la qualité de la restauration dépend fortement de la méthode de détection des gouttes utilisée comme pré-traitement pour fournir au module de restauration le masque des gouttes.

Enfin l'algorithme peut s'appliquer de façon identique dans le cas de gouttes légèrement défocalisées. En revanche, les premières études sur la détection des gouttes fortement défocalisées nous ont montré que celles-ci couvrent de larges surfaces, mais surtout ne sont pas totalement opaques. Aussi, l'utilisation du modèle optique, par exemple, permettrait potentiellement de récupérer l'information masquée par les gouttes d'une meilleure façon que l'"inpainting".

Applications

Sommaire

6.1	Pré-traitement : détection des panneaux	125
6.1.1	Algorithme de détection	126
6.1.2	Paramètres de la méthode de restauration	128
6.1.3	Base de données de panneaux	128
6.1.4	Résultats expérimentaux	130
6.1.5	Conclusion	135
6.2	Assistance : afficheur tête haute dans le brouillard	136
6.2.1	Temps de réaction	136
6.2.2	Modélisation du brouillard et mesure du temps de perception	138
6.2.3	Amélioration du temps de perception dans le brouillard	144
6.2.4	Conclusion	148
6.3	Traitement : Détection de l'espace navigable	149
6.3.1	Estimation de la distance de visibilité	149
6.3.2	Discussion	150
6.3.3	Segmentation de la ROI fondée sur les cartes géodésique	151
6.3.4	Détection de l'espace navigable	152
6.3.5	Conclusion	154
6.4	Conclusion	154

Dans cette partie, nous allons présenter les résultats d'évaluation de la méthode de restauration d'images de brouillard sur des systèmes d'aide à la conduite. Dans un premier temps, nous avons évalué les performances d'un système de détection de panneaux dans le brouillard avec et sans restauration à l'aide d'une base d'images de panneaux. La deuxième application a consisté à évaluer l'utilité d'un système d'affichage tête haute utilisant la restauration d'images de brouillard en mesurant le temps de perception et le taux de détection d'un panel de 30 sujets. Enfin nous avons évalué et optimisé la détection de l'espace navigable en utilisant la méthode du monde plan.

Notons que les 3 applications sont représentatives, chacune, d'une famille d'application : pré-traitement, assistance et traitement.

6.1 Pré-traitement : détection des panneaux

La plupart des systèmes de vision en général et de détection d'objets par caméra en particulier utilisent des descripteurs déduits à partir des luminances, des

contrastes et des couleurs de l'image. Ces derniers sont fortement dégradés par le brouillard. Un algorithme capable d'améliorer la qualité de l'image peut être utilisé comme pré-traitement afin d'améliorer les performances de tels systèmes. Dans [Hautière 2010b] Hautière et al. ont illustré brièvement cette idée en testant 3 algorithmes ADAS classiques : détection des marquages, des obstacles et des panneaux circulaires.

Afin de démontrer l'intérêt d'utiliser la restauration comme pré-traitement ADAS, nous avons étudié dans le détail les impacts du brouillard et de la restauration sur un système de détection de panneaux. Pour cela, nous avons créé une base d'images de synthèse contenant des panneaux en présence et absence de brouillard. Cette base de données sera mise à disposition de la communauté scientifique.

Nous allons commencer par présenter la méthode de détection des panneaux, les paramètres de la méthode de restauration et la base de données dédiée à l'évaluation. Nous montrerons ensuite, de manière quantitative, l'impact de la densité de brouillard sur les performances de la détection des panneaux. Enfin l'amélioration obtenue grâce à la restauration est présentée en termes de taux de détection et de distance de détection.

6.1.1 Algorithme de détection

Pour la détection et localisation des panneaux à partir d'une seule image nous avons utilisé le modèle géométrique présenté dans [Belaroussi 2009] (sans le modifier). Ce modèle est une transformée bilatérale, fondée sur l'orientation des gradients, appelée la transformée chinoise bilatérale ou TCB. Afin de permettre la détection des panneaux, nous utilisons les positions de leurs centres de symétrie et leurs tailles.

Nous commençons par créer un tableau cumulatif de la même taille que l'image. Un indice de symétrie de deux pixels P_i et P_j est ajouté au centre P du segment $[P_i, P_j]$. Pour chaque point P un ensemble de pixels votants est défini par :

$$\Gamma_P = \left\{ P \mid \frac{P_i + P_j}{2} = P \right\} \quad (6.1)$$

Soit θ_i l'angle que fait le gradient n_i de P_i par rapport à l'horizontale et soit α_{ij} l'angle entre le segment $[P_i, P_j]$ et l'horizontale comme illustré sur la figure 6.1. Chaque paire (n_i, n_j) est plus ou moins proche d'une certaine symétrie axiale et centrale en fonction de l'orientation (θ_i, θ_j) . Cette symétrie est approximée par une fonction explicitée sur la figure 6.1 appelée fonction de phase Θ .

La fonction de phase Θ est ensuite pondérée par une fonction $D(i, j)$, décroissante, de la distance $\| P_i - P_j \|$ qui permet de donner plus de poids aux paires proches et par une fonction logarithmique de la norme des gradients $r_i = \log(1 + \|n_i\|)$ et $r_j = \log(1 + \|n_j\|)$. L'indice de symétrie total de la TCB est donc calculé comme suit :

$$Accu(P) = \sum_{(i,j) \in \Gamma(P)} D(i, j) \Theta(i, j) r(i) r(j) \quad (6.2)$$

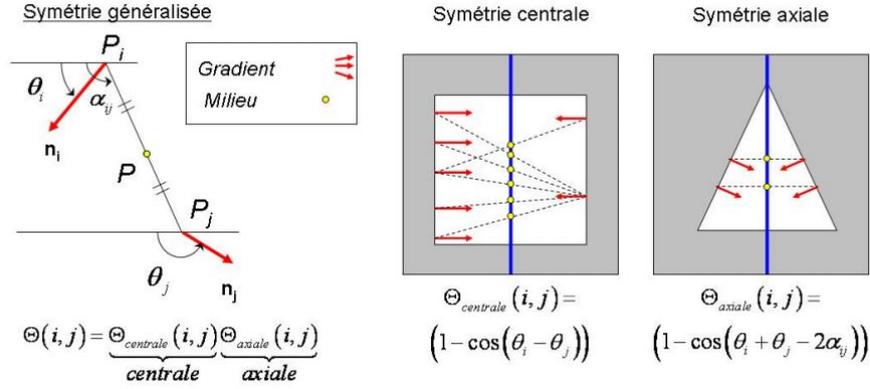


FIGURE 6.1 – Transformation en symétrie généralisée : l'indice de symétrie correspond au produit des deux termes $\Theta_{centrale}$ et Θ_{axiale}

La fonction de phase Θ modélise géométriquement les panneaux routiers tel que : n_i et n_j forment une symétrie centrale et le vecteur $P_i P_j$ est aligné avec n_i . Nous avons donc :

$$\Theta(i, j) = (W_\beta(\theta_i - \alpha_{ij}) + W_\beta(\theta_i - \alpha_{ij} - \pi)) \times W_\Delta(|\theta_i - \theta_j| - \pi) \quad (6.3)$$

Avec $W_R(x) = 1$ si $|x| < R$ et $W_R(x) = 0$ sinon. β est un angle de tolérance délimitant la région d'influence de P_i où P_j est recherché. Δ est une petite quantité qui assure que numériquement $|\theta_i - \theta_j| = \pi$: seuls les pixels avec des directions opposées ont voté à leur milieu.

La fonction de distance $D(i, j)$ est définie par :

$$D(i, j) = W_{R_{max}}(\|P_i - P_j\|) - W_{R_{min}}(\|P_i - P_j\|) \quad (6.4)$$

avec R_{max} et R_{min} respectivement le maximum et le minimum de la largeur des panneaux à détecter.

Dans le cas de formes polygonales avec quatre côtés ou plus, nous utilisons $\beta = \frac{\pi}{8}$ et $\Delta = \frac{2\pi}{N}$ avec $N = 8$: l'orientation du gradient est quantifiée sur 8 directions.

Les centres des objets sont détectés comme extremums locaux de la matrice *Accu* ayant une valeur supérieure à un seuil τ_{TCB} . Le seuil τ_{TCB} peut être utilisé pour générer les courbes ROC et DICE.

Dans notre expérience nous avons choisi $R_{min} = 0,04H$ et $R_{max} = 0,2H$ avec H la hauteur de l'image.

Pour une résolution d'image de 1400×600 l'intervalle de détection varie de 25×25 à 120×120 pixels. Dans le cas d'une caméra ayant une focale $f = 60$ mm et une taille de capteur 14×6 mm, cela correspond à une distance entre le panneau et la caméra comprise entre 50 et 250 mètres. La figure 6.2 montre un exemple de détection.

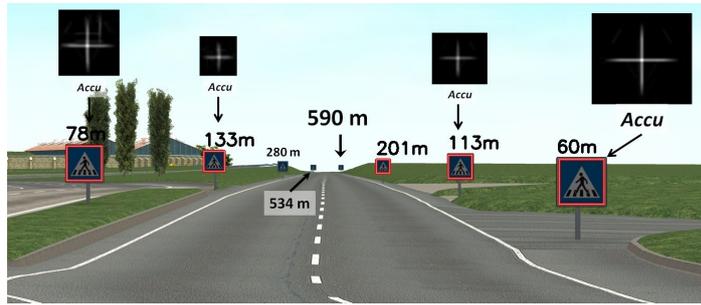


FIGURE 6.2 – Exemple de détection (rectangles rouges) avec la TCB : les panneaux les plus éloignés ne sont pas détectés à cause de leurs tailles dans l’image. Au dessus de chaque panneau détecté nous affichons le tableau *Accu* correspondant : les votes sont maximaux sur les axes de symétrie des panneaux.

6.1.2 Paramètres de la méthode de restauration

Afin d’effectuer la restauration des images de brouillard, nous avons utilisé l’algorithme proposé dans le chapitre 4 en utilisant toute la dynamique de l’image. Nous avons choisi les paramètres suivants : la taille de la fenêtre d’égalisation $W_s = 7$, la taille de la première gaussienne $G_{s1} = 11$ avec un écart type de 3, la taille de la deuxième gaussienne $G_{s2} = 7$ avec un écart type de 1 et un coefficient d’ajustement d’intensité $a = 0,7$.

6.1.3 Base de données de panneaux

À l’aide du logiciel Sivic, présenté dans le chapitre 3, nous avons créé une base d’images de panneaux de différents types, placés à différentes distances de la caméra, pour chaque image nous avons ajouté des brouillards uniformes de différentes densités.

Cette base de données contient un ensemble de 504 images originales couleurs (1400×600 pixels) contenant au total 2628 panneaux (vitesse, stop et passage piéton) avec la vérité terrain constituée de la distance des panneaux par rapport à la caméra noté $d_{sign} \in [36 - 590m]$, de leurs positions et leurs tailles dans l’image. La vérité terrain va nous servir pour faire l’évaluation de la détection comme nous allons le voir dans la suite.

Afin de créer une variabilité dans les images nous avons choisis 2 hauteurs de la caméra et 2 hauteurs de panneaux (4 configurations). La figure 6.3 montre les 3 types de panneaux utilisés et les 4 configurations.

La figure 6.4 montre la distribution du nombre des panneaux en fonction de leur distance à la caméra. Dans cet histogramme les distances sont quantifiées sur 20 *bins* tel que chaque *bin* a une largeur de 28 m. Par exemple le premier *bin* couvrant l’intervalle de distance $[36 - 64]$ m contient 228 panneaux. Le nombre total des panneaux tel que leur distance est inférieure à 280 m est égal à 1620 (61,6 % de la base de données), tandis que 1008 panneaux se trouvent plus loin. Donc environ

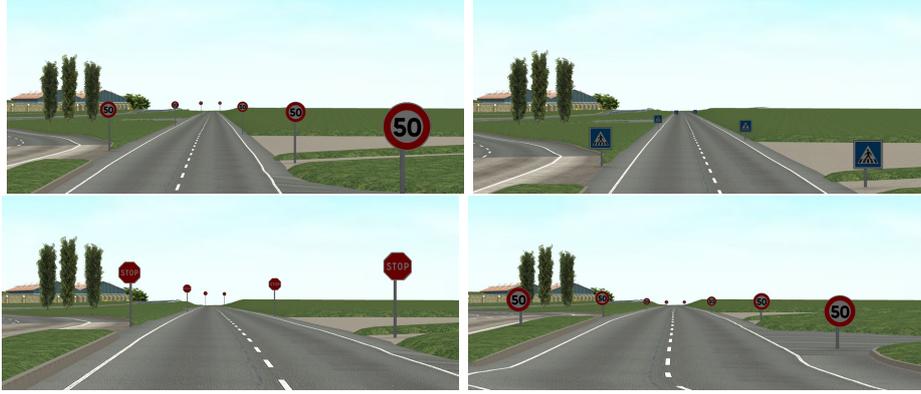


FIGURE 6.3 – Les trois types de panneaux utilisés dans la base de données. Les images montrent les différentes configurations de la hauteur de la caméra et des panneaux.

38 % des panneaux apparaissent très petits : sur la figure 6.2, les panneaux se trouvant à une distance de 280 m et 590 m ont respectivement des tailles de 20×20 et 10×10 pixels.

Notons que l'ensemble des panneaux de petites tailles n'a pas été exclu de notre base de données afin de permettre, dans le futur, l'évaluation d'autres méthodes capables de détecter cet ensemble.

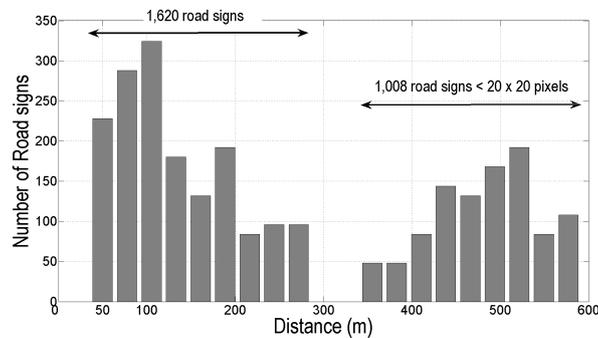


FIGURE 6.4 – Distribution de la distance des panneaux par rapport à la caméra. Un ensemble de 1620 panneaux sont proches de la caméra ($d_{sign} < 300m$) et un ensemble contenant le reste sont plus loin ($d_{sign} \geq 300m$).

En utilisant la carte de profondeur générée par Sivic et en utilisant le modèle de Koschmieder nous avons ajouté du brouillard dans chacune des images. Nous avons créé 7 types de brouillards uniformes tels que la distance de visibilité varie de 50 à 400 mètres. La première colonne de la figure 6.5 montre le résultat d'ajout du brouillard sur l'image de la figure 6.2.

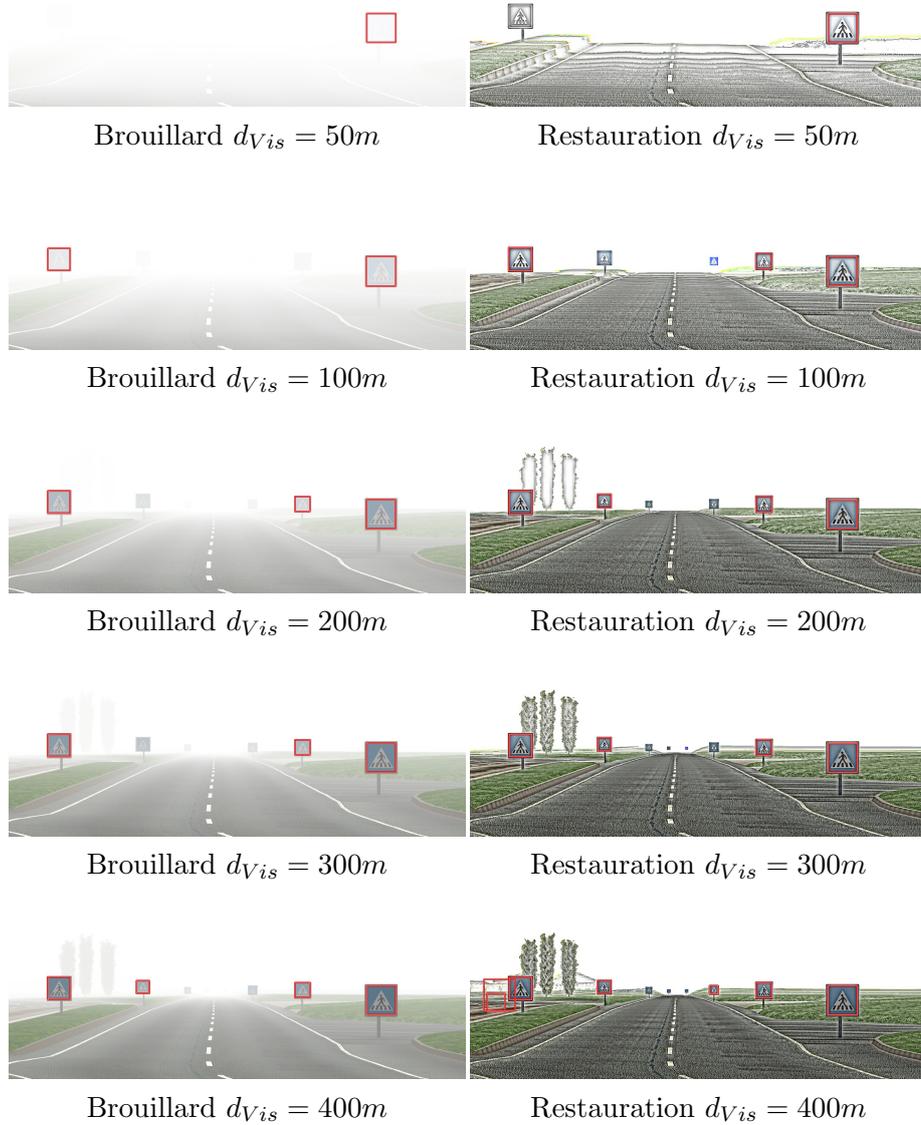


FIGURE 6.5 – Exemple d’images de la base de données. Les objets détectés par l’algorithme TCB encadrés en rouge : images de brouillards correspondant à différentes distances de visibilité d_{Vis} (colonne à gauche) et restaurations correspondantes (colonne à droite).

6.1.4 Résultats expérimentaux

6.1.4.1 Mesure des performances

Les performances de la TCB sont liées au nombre des vrais et des faux positifs. Une détection est considérée comme vrai positif lorsque deux conditions sont réunies : la distance entre son centre et le vrai centre est inférieure à 20 % de la

dimension du panneau (tolérance en fonction des dimensions du panneau) et l'erreur absolue relative de la largeur et la longueur du panneau est inférieure à 45 % (tolérance sur la taille).

La courbe ROC utilisée est le tracé du taux de détection correcte CDR (correct detection rate) en fonction des faux positifs par image $FPPI$:

$$CDR = \frac{TP}{P} \quad ; \quad FPPI = \frac{FP}{NbImg} \quad (6.5)$$

avec TP le nombre de vrais positifs, P le nombre total des panneaux, FP le nombre de faux positifs et $NbImg$ le nombre total des images.

La courbe du coefficient DICE est tracée en fonction du seuil τ_{TCB} :

$$Dice = \frac{2TP}{TP + FP + P} \quad (6.6)$$

6.1.4.2 Performances par temps clair

Dans la figure 6.6 la courbe ROC montre les performances de la TCB sur les images originales (courbe noire continue marquée en rouge). Nous constatons que le nombre maximum des panneaux détectés est d'environ 58 % des panneaux de la base de données : $CDR = 58\%$ pour $FPPI = 0,86$. Ce point de la courbe ROC montre que 1519 de 2628 panneaux sont détectés pour 436 fausses alarmes sur l'ensemble des 504 images. Comme les objets de taille inférieure à 20×20 pixels ne peuvent pas être détectés, le CDR ne peut pas excéder 62 %. Sur les 1620 panneaux tels que $d_{sign} < 300m$, 97 % sont détectés correctement.

La courbe du coefficient Dice (figure 6.7) est utilisée pour déterminer le seuil τ_{TCB} optimal : en choisissant le seuil qui maximise le coefficient Dice, nous obtenons un bon compromis entre les vrais et faux positifs. Le seuil trouvé sur l'ensemble des images originales est $\tau_{TCB} = 180$, la même valeur a été obtenue en utilisant la TCB avec d'autres base de données d'images réelles (voir [Belaroussi 2009, Belaroussi 2010]).

6.1.4.3 Impact de la densité du brouillard sur la perception des panneaux

Afin de caractériser l'influence de la densité du brouillard sur les performances de la détection, les courbes ROC et Dice des ensembles des images de brouillard ont été tracées en lignes pointillés sur les figures 6.6 et 6.7 : chaque densité du brouillard correspond à une couleur, avec $d_{vis} = \{50, 100, 150, 200, 300, 400\}$ (m). Pour montrer l'amélioration apportée par la restauration à la détection, nous avons tracé les courbes correspondantes à chaque densité de brouillard avec la même couleur en lignes continues. Le tableau 6.1 est extrait de la courbe ROC pour $FPPI = 3$.

Comme prévu, la présence de brouillard dégrade les performances de l'algorithme de détection : un brouillard léger tel que la distance de visibilité est dans l'intervalle $[250 - 400m]$ induit un maximum de détection autour de 44 % (comparé à 58 % par

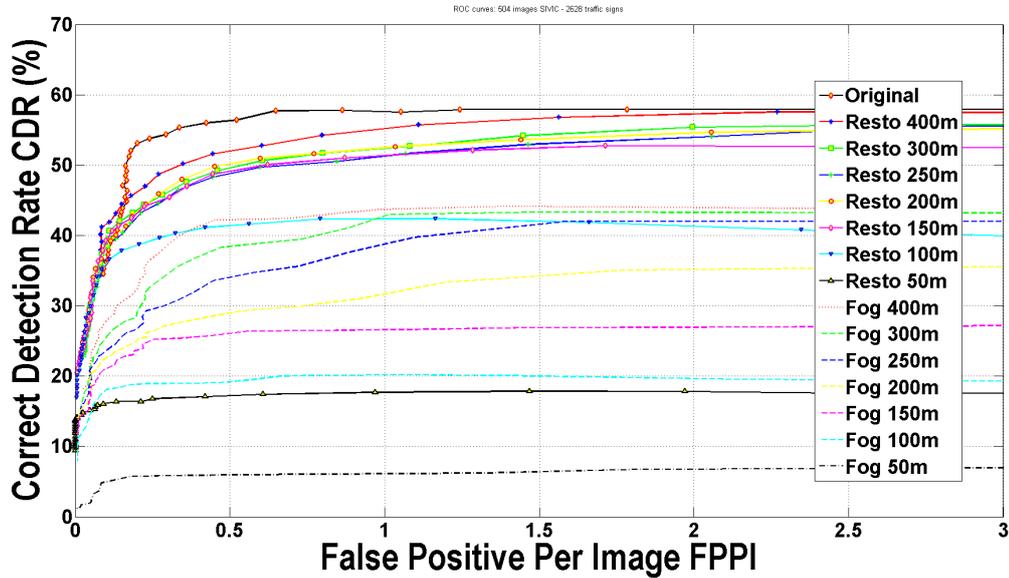


FIGURE 6.6 – Courbes ROC des images originales, images avec différents brouillards et images après restauration. Chaque ensemble est constitué de 504 images contenant 2628 panneaux

d_{Vis} (mètre)	250m-400m	200m	150m	100m	50m
Image de brouillard	44 %	35 %	27 %	20 %	7 %
Image restaurée	56 %	55 %	53 %	42 %	18 %

TABLE 6.1 – Maximum du CDR atteint en fonction de la distance de visibilité qui décroît (FPPI=3).

temps clair). Plus la distance de visibilité diminue, plus le taux de détection chute : 35 % à 200m, 27 % à 150m, 20 % à 100m et moins de 7 % à 50m. Ces résultats montrent que les conditions météorologiques influencent fortement les performances de l'ADAS.

Ces résultats sont confirmés par les courbes Dice, sur la figure 6.7, qui montrent que le coefficient Dice diminue en présence de brouillard pour continuer à assurer le bon fonctionnement de l'algorithme de détection. En effet le brouillard modifie les propriétés de l'image, en particulier le contraste et les gradients en fonction de la profondeur des objets et donc le point de fonctionnement de l'algorithme. Le point de fonctionnement optimal peut être réglé en utilisant les caractéristiques du brouillard comme suggéré dans [Hautière 2003]. Toutefois cela doit être fait pour chaque ADAS utilisant une caméra et devient rapidement un travail fastidieux.

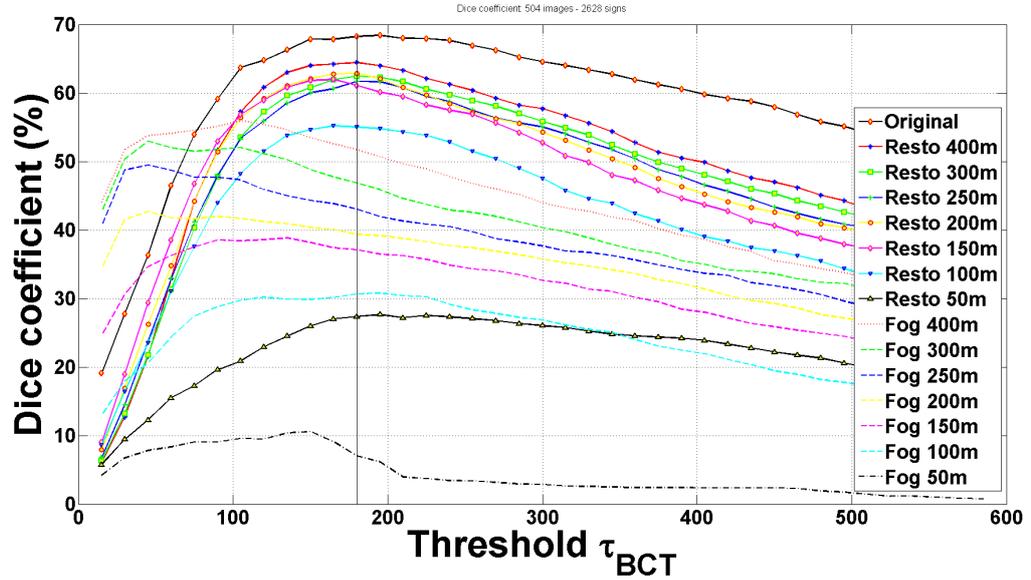


FIGURE 6.7 – Courbes Dice des images originales, images avec différents brouillards et images après restauration. Chaque ensemble est constitué de 504 images contenant 2628 panneaux

6.1.4.4 Amélioration par restauration d'images

Notre approche consiste à utiliser la restauration d'images comme pré-traitement sans modifier la configuration de l'ADAS.

Nous rappelons qu'en présence de brouillard, à partir d'une certaine distance les intensités de l'image se confondent avec l'intensité du ciel et nous avons une absence totale d'information au delà de cette distance (voir chapitre 3), en particulier pour $d_{Vis} = 50$ m. Ceci est illustré sur la figure 6.5 : Lorsque la distance de visibilité diminue, le nombre de panneaux détectés diminue, et ce dans les cas images de brouillard et images restaurées, alors que le nombre de panneaux détectés dans l'image originale reste assez élevé.

Statistiquement, les courbes ROC et Dice des images restaurées donnent de meilleurs résultats que celles des images de brouillard. En particulier pour des distances de visibilité supérieures à 200 m, la restauration permet de retrouver des performances de détection comparables aux performances de détection en absence de brouillard : 55 % de panneaux détectés après restauration (comparé à 58 % en absence de brouillard) comme le montre le tableau 6.1.

En plus, toutes les courbes Dice des images restaurées présentent une valeur maximale très proche du maximum de la courbe des images originales. Ceci montre que l'utilisation de la restauration comme pré-traitement permet de garder le même point de fonctionnement τ_{TCB} que dans le cas des images originales.

6.1.4.5 Amélioration de la gamme de fonctionnement des ADAS

Afin de mieux qualifier la dégradation de la détection dans le brouillard, sur la figure 6.8 nous avons tracé le CDR en fonction de la profondeur d_{sign} . Cette figure montre que, en absence de brouillard, l'algorithme TCB détecte tous les panneaux situés à une distance entre 64 m et 230 m.

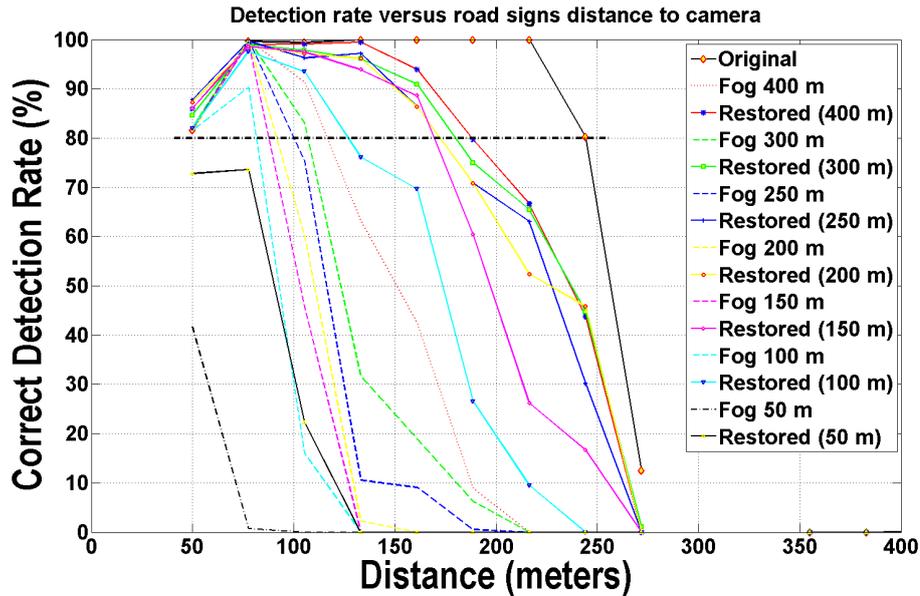


FIGURE 6.8 – Taux de détection des panneaux en fonction de leur distance par rapport à la caméra d_{sign}

Sur la figure 6.9 nous avons tracé la distance des panneaux détectés avec un taux de 80 % (ligne horizontale en pointillé sur la figure 6.8) en fonction de la distance de visibilité. Cette courbe montre que la distance de détection en utilisant la restauration augmente de 45 à 80 m.

La figure 6.8 montre que par temps de brouillard la détection varie de 80 m à 120 m pour une distance de visibilité d_{vis} allant de 100 m à 400 m. En utilisant la restauration nous améliorons l'intervalle de détection : $d_{sign} \in [125 - 190]$ m.

Considérant le cas d'un brouillard dense : sur la figure 6.5 à une distance $d_{vis} = 100$ m, la détection la plus éloignée est à $d_{sign} = 78$ m, après restauration le panneau le plus loin est détecté à $d_{sign} = 113$ m. En moyenne, sur les 504 images, le gain en détection est de 45 m dans le cas d'un brouillard dense, ce qui représente une augmentation de 55 % par rapport à la limite de 78 m sans restauration.

Pour des brouillard tel que $d_{vis} \in [150 - 200]$ m l'augmentation des détections est de 80 m en utilisant la restauration. Comparé à la détection dans le cas de brouillard, l'amélioration est de plus de 90 % : la distance de détection est presque doublée. Pour un véhicule ayant une vitesse de 60 km/h dans de telles conditions météorologique, un panneaux sera détecté 5 secondes plus tôt en utilisant la restauration.

Dans les cas où le brouillard est léger $d_{vis} > 250$ m, l'intervalle de distances

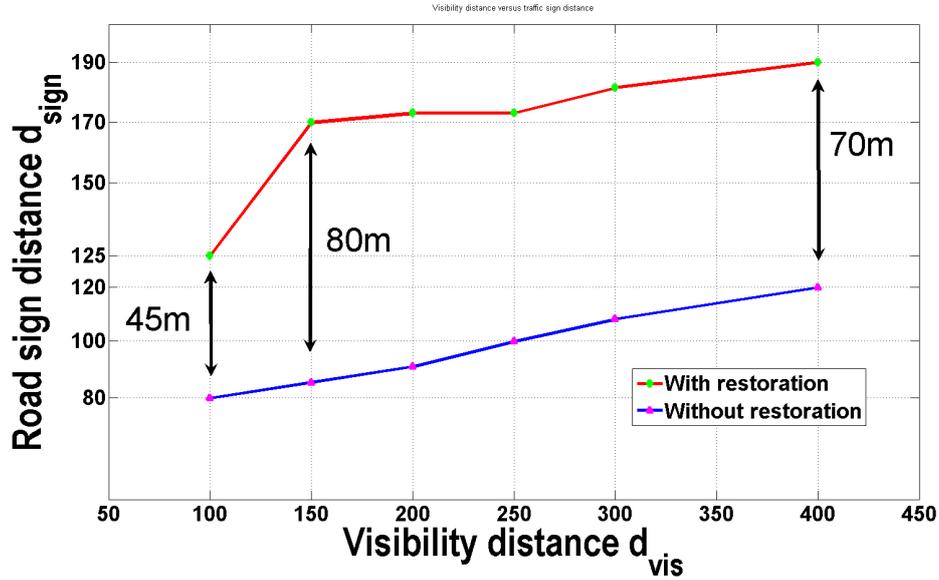


FIGURE 6.9 – Distance d_{sign} des panneaux détectées avec un taux de 80 % en fonction de la distance de visibilité d_{vis} .

de détection est $d_{sign} \in [100 - 120]$ m et après la restauration nous obtenons $d_{sign} \in [170 - 190]$ m. Le gain en distance de détection est de 70 m en utilisant la restauration : comparé à la distance de détection dans le brouillard, le gain est de 60 à 70 %. à une vitesse de 60 km/h les panneaux seront détectés 4 secondes plus tôt.

6.1.5 Conclusion

Le contraste et la couleur des panneaux les plus distants sont les plus dégradés par la diffusion atmosphérique. Notre système, dans le cas d'un brouillard léger, permet de retrouver des performances quasi-comparables à celles d'un système de détection par temps clair et dans le cas d'un brouillard dense, d'augmenter la distance de détection d'un taux non-négligeable. Cela va permettre au conducteur d'avoir un temps suffisant pour prendre sa décision et effectuer des manœuvres de manière sécurisée. Nous avons donc montré que l'utilisation de la restauration d'images permet d'étendre la gamme de fonctionnement d'un ADAS et potentiellement réduire le nombre d'accidents.

6.2 Assistance : afficheur tête haute dans le brouillard

Nous allons présenter dans cette partie une expérience que nous avons effectuée et qui vise à mesurer le temps de perception (que nous appelons, d'une façon plus générale, temps de réaction) d'un objet par un humain, par temps clair, dans le brouillard et en utilisant la restauration afin d'évaluer l'impact d'un afficheur tête haute ou HUD (Head up display), utilisant notre système d'amélioration de la visibilité, sur les performances d'un conducteur.

Menée sur un panel de 30 sujets, l'expérience s'est déroulée en 2 étapes. Dans un premier temps nous avons modélisé sur écran un objet noir sur un fond uniforme dans le brouillard en utilisant le modèle de Koschmieder et évalué le temps de perception et le taux de détection. Dans une seconde expérience nous avons utilisé des images Sivic pour modéliser des scènes routières de brouillard et évaluer le temps de détection d'un véhicule.

Nous allons commencer par montrer les différents outils et techniques utilisés pour mettre en place l'expérience et présenterons les résultats des 2 expériences.

6.2.1 Temps de réaction

Le temps de réaction est le temps écoulé entre la présentation d'un stimulus (auditifs, visuel ...) et la réponse du sujet (appui sur un bouton, mouvement des yeux, réponse vocale ...). La figure 6.10 illustre le principe de mesure de temps de réaction. Nous commençons par avertir le sujet du début de l'expérience et nous faisons apparaître le stimulus après une certaine durée aléatoire et mesurons le temps de réponse.

Dans notre cas le stimulus est un objet dans l'image que nous cherchons à détecter, le signal avertisseur est un message texte qui apparaît à l'écran pour signaler au sujet que le stimulus va bientôt apparaître et qu'il doit se préparer pour répondre. Nous allons voir dans la suite plus en détails comment l'expérience a été effectuée.

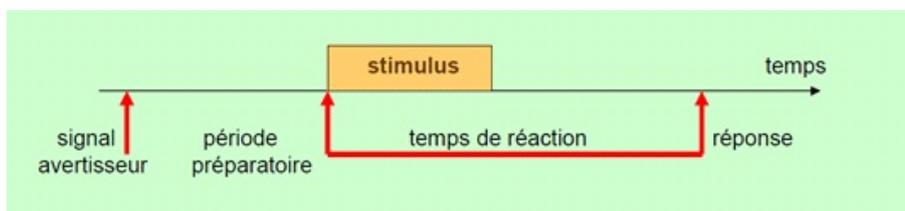


FIGURE 6.10 – Mesure du temps de réaction

Notons que pendant la période préparatoire il peut y avoir des erreurs d'anticipation, par exemple, le sujet va répondre avant l'apparition du stimulus. Par conséquent la variabilité des réponses ne va pas correspondre à une distribution gaussienne. Pour surmonter ce problème la même mesure est effectuée plusieurs fois et nous considérons la médiane de toutes les répétitions afin d'éliminer les valeurs

aberrantes.

Nous distinguons plusieurs types de temps de réaction suivant la tâche que nous demandons à l'utilisateur d'accomplir : temps de réaction de détection (simple), de reconnaissance, de choix. . . En 1868, le professeur Donders a été le premier scientifique à mesurer le temps de réaction en laboratoire et il a conclu que plus la tâche est complexe plus le temps de réaction est important.

Dans le cadre de notre étude nous nous sommes limités à la mesure du temps de réaction simple : Un objet apparaît et le sujet doit appuyer sur un bouton dès qu'il le perçoit. Nous appelons le temps mesuré "temps de perception". Nous faisons apparaître le stimulus (l'objet) toujours au même endroit afin de ne pas prendre en compte le temps de recherche et de n'avoir donc que le temps de perception.

6.2.1.1 Loi de Piéron

Cette loi exprime le fait que plus l'intensité d'un certain stimulus est faible plus le temps de réaction est important. L'équation 6.7 décrit le temps de réaction en fonction de l'intensité du stimulus I et des caractéristiques β et t_0 d'une modalité sensorielle donnée. α est un paramètre qui dépend des unités, suivant si le stimulus est une image, un son. . .

$$TR = \alpha I^{-\beta} + t_0 \quad (6.7)$$

Dans le domaine visuel la valeur de β est entre 0,3 et 0,35. α et t_0 sont à déterminer expérimentalement.

La figure 6.11 montre l'évolution du temps de réaction en fonction de l'intensité du stimulus selon la loi de Piéron.

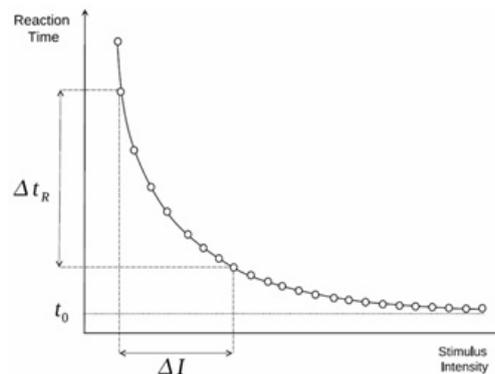


FIGURE 6.11 – Temps de réaction en fonction de l'intensité du stimulus

6.2.1.2 Temps de réaction dans le brouillard

Dans le brouillard, le contraste des objets s'atténue avec la distance. À cause d'une visibilité réduite, le conducteur n'aperçoit les objets avec un contraste suffisant que lorsqu'il est proche d'eux et il en résulte un freinage brusque qui peut augmenter

la gravité d'un accident. Le but des expériences que nous allons présenter dans la suite est de mesurer le temps de réaction par temps clair, dans le brouillard et en utilisant notre méthode de restauration afin d'évaluer les gains, en termes de performances du conducteur, apportés par notre système de restauration.

Les questions qui se posent sont donc :

- De combien le temps de perception augmente en présence du brouillard par rapport à un temps clair ?
- À partir de quelle distance un objet est détectable dans un brouillard d'une certaine densité ?
- Et de combien nous pouvons améliorer le temps de perception et le pourcentage de détection d'un objet dans le brouillard en utilisant la restauration ?

Pour répondre à ces questions nous avons mis en place deux expériences que nous allons présenter dans la suite.

6.2.2 Modélisation du brouillard et mesure du temps de perception

L'expérience a eu lieu dans les locaux d'IFSTTAR-Paris. Nous avons à notre disposition un écran avec une grande résolution et un PC placés une salle sombre (afin de n'afficher aux sujets que les luminances provenant de l'écran). La position de l'écran et du siège où le sujet devait s'asseoir n'ont pas changé durant toute l'expérience. La figure 6.12 montre une image de la salle Clovis et la configuration choisie pour cette expérience.

Cette expérience a consisté à présenter au sujet des images de disques, qui modélisent un objet noir d'une taille de 2 mètres plongé dans le brouillard, affichées sur un plan uniforme ayant une valeur fixe qui représente l'intensité du ciel.

Nous faisons varier l'intensité du disque selon Koschmieder : l'objet apparaît avec la même luminance qu'un objet noir plongé dans un brouillard d'une certaine densité.

La taille du disque sur l'écran est calculée en fonction de la distance de l'objet dans le brouillard : le disque apparaît à l'écran au sujet avec la même taille angulaire qu'un objet de 2 mètres observé en vrai à une certaine distance. Enfin, nous mesurons le temps de perception de chaque disque par le sujet.

La figure 6.13 montre des images de disques correspondants à un objet se trouvant à différentes distances dans le brouillard.

À chaque image, nous signalons au sujet le commencement, nous affichons une image de la scène sans le disque pendant une période $t_{preparatoire}$ qui varie à chaque fois aléatoirement entre 1 et 3 secondes, affichons l'image de la scène avec le disque pendant une durée de $5, 5 - t_{preparatoire}$ secondes et mesurons le temps de perception du disque.

Sans réponse du sujet nous considérons qu'il n'y a pas eu détection. Nous éliminons aussi les erreurs d'anticipation (réponse avant l'apparition du stimulus). Toutes les images sont affichées dans un ordre aléatoire.

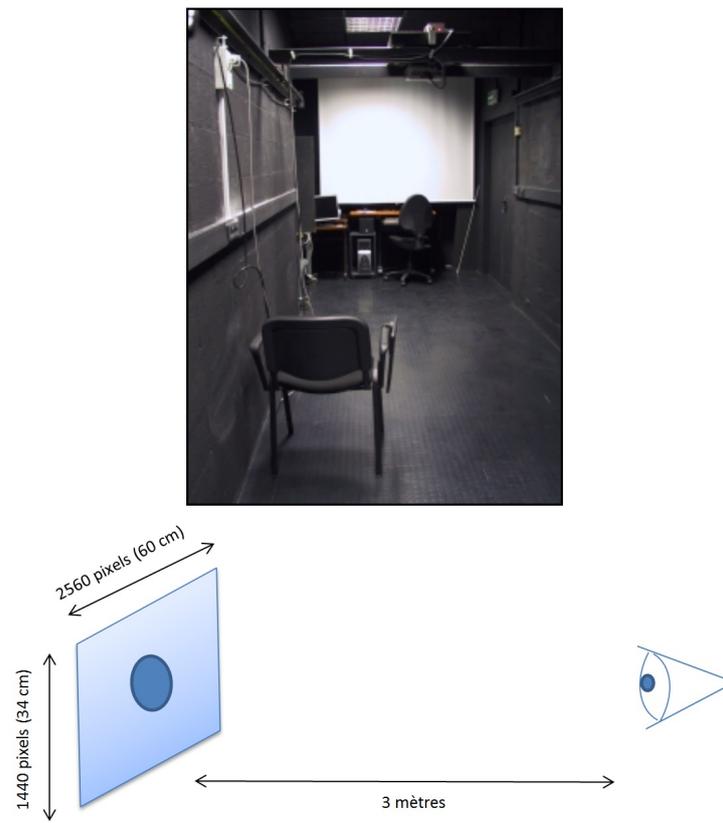


FIGURE 6.12 – Image du haut : salle Clovis. Image du bas : configuration de l'expérience. L'emplacement de l'écran par rapport à l'observateur ainsi que la résolution et les paramètres de luminance et de contraste n'ont pas changé durant toute l'expérience.

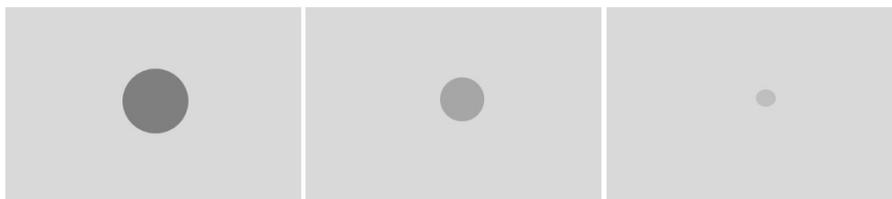


FIGURE 6.13 – Images des disques : dans le brouillard, plus l'objet est loin plus il devient petit et perd son contraste.

Comme le modèle de Koschmieder s'applique sur des luminances, afin d'afficher le disque avec la même luminance que dans le monde réel nous avons effectué un calibrage de l'écran utilisé à l'aide d'un luminance-mètre comme nous allons le montrer dans la suite.

6.2.2.1 Calibrage photométrique

Nous avons donc commencé par effectuer un calibrage de l'écran afin d'obtenir une relation entre les niveaux de gris des pixels de l'image et la luminance affichée à l'écran en cd/m^2 . Ceci va nous permettre d'utiliser le modèle de Koschmieder correctement en l'appliquant sur les luminances et de convertir ensuite les luminances en niveaux de gris pour l'affichage à l'écran. La figure 6.14 montre la courbe de calibrage.

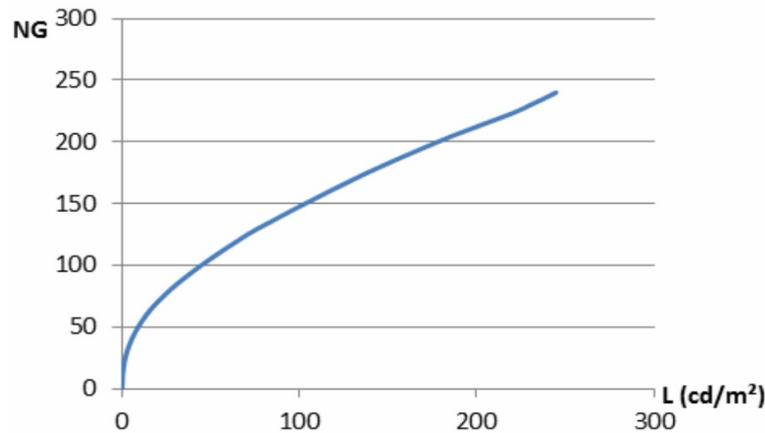


FIGURE 6.14 – Courbe du calibrage de l'écran à l'aide d'un luminance-mètre : en abscisse la luminance perçue et en ordonnée le niveau de gris correspondant.

6.2.2.2 Logiciel Presentation

Une fois le calibrage effectué, nous avons utilisé un logiciel nommé "Presentation" (Neurobs) utilisant un langage proche de "Python" qui nous a permis d'afficher les images des disques, de faire varier leur luminances et leurs tailles et de récupérer les temps de réaction des sujets avec une précision inférieure à une milliseconde.

6.2.2.3 Modélisation du brouillard

Pour modéliser le brouillard dans le cas des disques, nous avons utilisé le modèle de Koschmieder $L_{app} = L_o \exp(-\beta d) + L_f(1 - \exp(-\beta d))$, avec L_{app} la luminance apparente dans le brouillard que nous voulons attribuer au disque en fonction du brouillard considéré, L_o la luminance du disque, L_f la luminance du ciel (fond de l'image).

La figure 6.15 montre un schéma illustrant le principe de modélisation du brouillard. L'image de gauche montre un objet de taille T , dans le monde réel, observé à une distance d . L'image de droite montre ce que nous voulons afficher aux sujets : un objet observé à une distance de 3 m ayant la même taille angulaire et la même luminance que l'objet dans le monde réel.

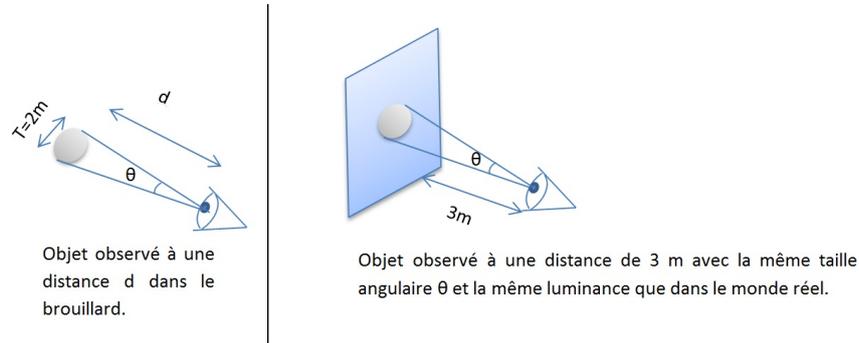


FIGURE 6.15 – Modélisation d'un objet dans le brouillard sur un écran.

Nous considérons que le disque est un objet noir de luminance L_o nulle et ayant une taille $T_o = 2$ m. Notons que L_o est choisi pour maximiser la visibilité de l'objet dans le brouillard. Si L_o augmente, la visibilité diminue.

La luminance du ciel est $L_f = 150 \text{ cd/m}^2$ afin d'avoir une scène assez lumineuse sans pour autant éblouir les yeux du sujet.

Nous avons utilisé 3 densités de brouillard correspondant aux distances de visibilité suivantes $d_{Vis} = 50, 80, 150(m)$. Pour chaque d_{Vis} l'objet est placé à 6 distances différentes afin de détecter le seuil de chute de visibilité (voir tableau 6.2).

Afin de caractériser la sensibilité du système visuel humain au contraste de luminance, un niveau de visibilité VL a été introduit par Adrian [Adrian 1989]. Afin d'anticiper le résultat et sélectionner les distances de l'objet, nous avons utilisé le modèle d'Adrian qui définit le niveau de visibilité VL en fonction d'un certain nombre de paramètres (taille, distance, hauteur, durée de l'observation ...).

Le tableau 6.2 montre les différentes distances de visibilité, les distances de l'objet, les luminances et les contrastes utilisées et les valeurs de VL déduites. Plus la distance augmente, plus le niveau de visibilité diminue.

La taille de l'objet sur l'écran est calculée, en considérant qu'il apparaît avec la même taille angulaire que dans le monde réel, à partir de la formule 6.8 déduite après un simple calcul géométrique à partir des images de la figure 6.15.

$$T_{ecran} = \frac{3T}{d} = \frac{6}{d} \quad (6.8)$$

Ensuite nous convertissons cette taille en pixels en connaissant la taille des pixels de l'écran.

Pour chaque sujet, nous affichons $3 \times 6 = 18$ images différentes (6 emplacements de l'objet pour chacune des 3 densités de brouillard) dans un ordre aléatoire. Le nombre de répétitions de chaque image est égal à 8.

6.2.2.4 Résultats

Nous avons fait passer l'expérience à 30 sujets, la figure 6.16 montre les courbes des temps de réaction, calculés comme la médiane des temps de réaction de tous

d_{Vis}	Contraste	$L_o(cd/m^2)$	$d(m)$	VL d'Adrian
150m	0,226	116	75	31,5
	0,126	131	105	16,7
	0,067	140	138	8,3
	0,033	145	171	3,9
	0,02	147	210	2,2
	0,013	148	228	1,4
80m	0,213	118	42	31,1
	0,12	132	57	17,4
	0,06	141	75	8,33
	0,033	145	93	4,49
	0,02	147	105	2,64
	0,013	148	120	1,72
50m	0,1867	122	28	27,7
	0,113	133	37	16,7
	0,06	141	47	8,7
	0,033	145	58	4,7
	0,02	147	67	2,8
	0,013	148	75	1,85

TABLE 6.2 – Paramètres de l'expérience en fonction de la distance de visibilité

les sujets en fonction de la distance d (6 distances), pour chaque type de brouillard et affichés en couleurs différentes : en rouge $d_{Vis} = 50$ m, en vert $d_{Vis} = 80$ m et en bleu $d_{Vis} = 150$ m. Sur chaque distance nous avons marqué le taux de détection moyen de tous les sujets.

Nous obtenons une courbe qui ressemble à celle de Piéron. Nous constatons que plus la distance augmente plus le temps de réaction augmente et plus le taux de détection diminue.

Nous montrons sur la figure 6.16 la même courbe avec les écarts-type des temps de réaction entre sujets.

Nous observons que plus la distance augmente plus l'écart type est grand, cela est dû au fait que lorsque le contraste est assez visible (objet proche) les sujets arrivent tous à le détecter avec un temps de réaction équivalent alors que lorsque le contraste est très faible (objet loin) certains sujets le détecte rapidement (bien que moins rapidement que lorsqu'il est proche) alors que pour d'autres cela prend plus de temps.

En fixant un seuil de détection à 500 ms, la distance d correspondante à ce seuil est de 58 m pour $d_{Vis} = 50$ m, de 93 m pour $d_{Vis} = 80$ m et de 171 m pour $d_{Vis} = 150$ m. Ces trois distance à partir desquelles nous considérons que l'objet est rapidement détectable correspondent à un contraste de 3 % et un niveau de visibilité d'Adrian VL d'environ 4. Ces résultats sont proches des normes CIE : un contraste de 5 % minimum est nécessaire pour qu'un objet soit visible.

Nous allons montrer dans la suite la deuxième expérience que nous avons ef-

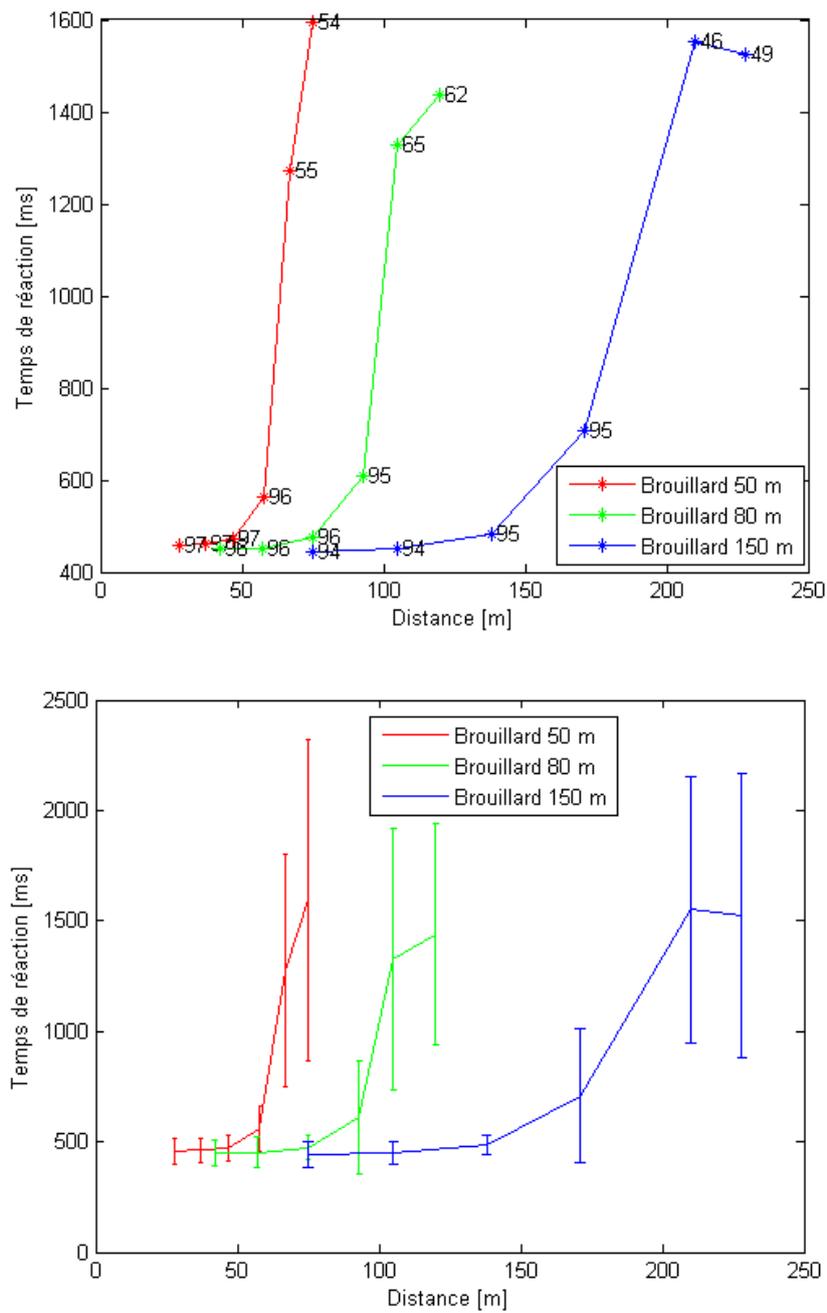


FIGURE 6.16 – Image du haut : temps de réaction en fonction de la distance et taux de détection pour chaque distance. Courbe en rouge $d_{Vis} = 50$ m, courbe en vert $d_{Vis} = 80$ m et courbe en bleu $d_{Vis} = 150$ m. Image du bas : temps de réaction en fonction de la distance et les écarts-type entre sujets pour chaque distance et chaque brouillard.

fectuée et dans laquelle nous avons remplacé les images de disques par des images routières de synthèse.

6.2.3 Amélioration du temps de perception dans le brouillard

Cette deuxième expérience, qui s’est déroulée en même temps que la première expérience en utilisant les mêmes outils (salle, écran, logiciel “Presentation”), a consisté aussi en une mesure des temps de perception (des mêmes 30 sujets) d’une voiture dans une scène routière affichée sur écran.

6.2.3.1 Modélisation d’images routières dans le brouillard

Pour créer les images routières, nous avons utilisé le logiciel SiVIC. Nous avons choisi une voiture de couleur noire (luminance presque nulle), l’arrière plan de l’image (la scène) est le même dans toute l’expérience (scène statique) et seule la position de la voiture sur la route varie (voir figure 6.17).

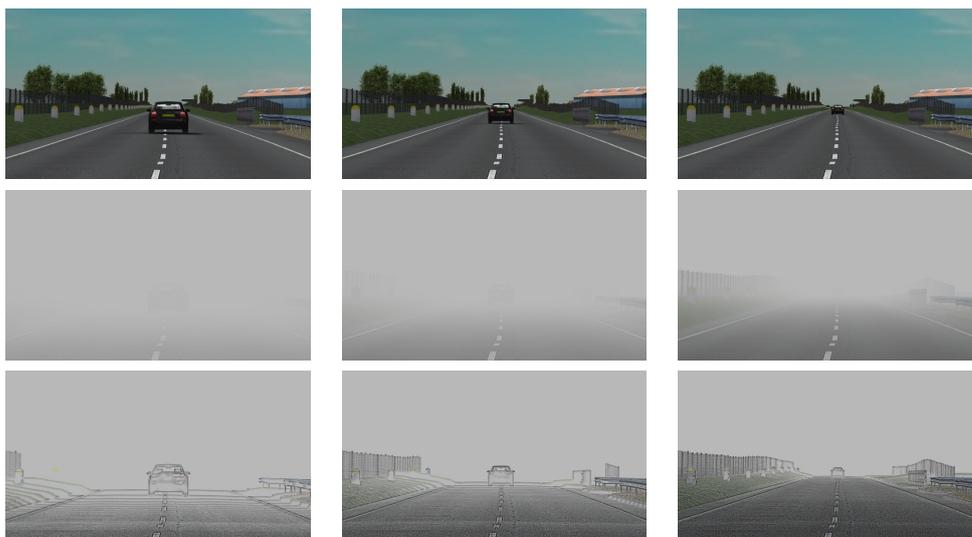


FIGURE 6.17 – Image originale avec la voiture placée à différentes distances (première ligne), images avec des brouillards $d_{Vis} = 50, 80, 150(m)$ (deuxième ligne) et images restaurées (troisième ligne).

La figure 6.18 montre la scène réelle (que nous avons simulé à l’aide de SiVIC) capturée par une caméra et l’image correspondante affichée sur l’écran pour le sujet.

Notre but ici est d’évaluer les performances d’un conducteur utilisant un système d’affichage tête haute : à la différence de la première expérience le but n’est pas de modéliser sur écran comment la scène est perçue dans le brouillard en réalité (même taille de l’objet et même luminance que dans le monde réel) mais plutôt comment la scène est perçue par un conducteur possédant un système d’affichage tête haute.

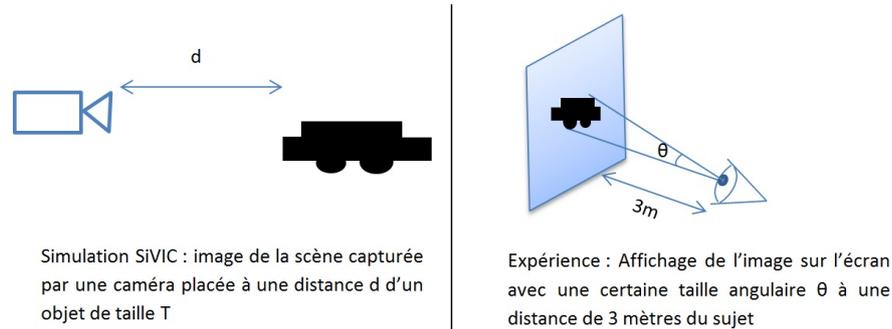


FIGURE 6.18 – Modélisation d'un objet dans le brouillard sur un écran.

La taille de l'objet et sa luminance vont donc dépendre des caractéristiques de la caméra et de l'écran.

Dans notre cas nous avons choisi une focale de la caméra $f = 128$ mm afin que l'objet ait une taille visible dans la gamme de distance dans laquelle nous voulons le placer ($[28 - 280]$ m).

Comme nous avons calibré notre écran et comme la réponse de la caméra SiVIC est linéaire, la luminance de l'objet sera affichée avec la même luminance que dans le monde réel.

Nous avons donc placé l'objet à 5 distances dans 3 cas : par temps clair, par temps de brouillard et après restauration. À chaque fois le temps de perception est mesuré.

En utilisant le modèle de Koschmieder et la carte de profondeur des images (fournie par SiVIC) nous avons ajouté dans les images originales 3 types de brouillard uniforme $d_{Vis} = 50, 80, 150(m)$.

Nous avons donc affiché pour chaque sujet $5 + (5 \times 2 \times 3) = 35$ images (5 images par temps clair, 5 images de brouillard pour chacun des 3 types de brouillards et les 5 images restaurées correspondantes). Chaque image est présentée 6 fois.

6.2.3.2 Déroulement de l'expérience

Comme précédemment, à chaque image nous signalons au sujet le commencement, nous affichons une image de la scène sans la voiture pendant une période $t_{preparatoire}$ qui varie à chaque fois aléatoirement entre 1 et 3 secondes, affichons l'image de la scène avec la voiture pendant une durée de $5,5 - t_{preparatoire}$ secondes et mesurons le temps de réponse depuis l'affichage de la voiture.

Les erreurs d'anticipation sont éliminées. Pour chaque groupe (images par temps clair, images des 3 types de brouillard et restaurations correspondantes) nous affichons aléatoirement les images (la distance varie aléatoirement).

d_{Vis} (mètre)	50	80	150
d_{Obj} (mètre)	93	138	280
TR Brouillard (sec)	2	2	2,4
TR Restauration (sec)	0,5	0,5	0,5
TD Brouillard	30 %	20 %	4 %
TD Restauration	100 %	98%	99 %

TABLE 6.3 – Exemple de temps de réaction TR et taux de détection TD , d'un objet placé à une distance d_{Obj} de la caméra, dans le cas d'images de brouillard (caractérisé par d_{Vis}) et dans le cas d'images restaurées. En utilisant la restauration, TR diminue et TD augmente.

6.2.3.3 Résultats

Pour chaque type d'images (sans brouillard, brouillard et restauration) et pour chacun des 30 sujets nous avons calculé la médiane des temps de réaction des 6 répétitions. Ensuite, nous avons considéré la médiane des mesures de tous les sujets.

La figure 6.19 montre le tracé du temps de réaction (de chaque groupe d'images) en fonction de la distance de la voiture. Nous affichons pour chaque distance le taux de détection, sur les 6 répétitions, calculé comme étant la moyenne des taux de détection de chaque sujet.

Nous constatons qu'en absence de brouillard le temps de réaction est autour de 500 ms, ce qui correspond au résultat prévisible. Dans le cas d'images de brouillard, le temps de réaction augmente en fonction de la distance de la même façon que pour les images des disques (expérience précédente). En utilisant la restauration nous arrivons à trouver une courbe très similaire à celle par temps clair : temps de réaction quasi-constant pour les différentes distances. La fin de chaque courbe des images de brouillard et de restauration correspond à une absence d'information visuelle à partir de cette distance.

Avec notre système nous arrivons donc à améliorer les performances visuelles du conducteur dans une certaine gamme de distance (tant que l'information est disponible).

Nous montrons sur le tableau 6.3, un exemple de taux de détection et taux de réaction extrait de la figure 6.19. Le taux d'amélioration, pour les trois types de brouillard, varie de 1,5 à 2 secondes pour un objet placé à environ 1,8 fois la distance de visibilité et ceci pour les différents types de brouillard.

Nous constatons aussi qu'en utilisant la restauration, le taux de détection est amélioré avec un taux entre 70 % et 95 % pour les objets les plus distants. Nous montrons aussi sur la figure 6.19, les courbes avec les écarts-type des temps de réaction entre sujets pour chaque distance et chaque groupe d'images. De la même façon que dans l'expérience précédente, nous constatons que l'écart type entre sujets augmente en fonction de la distance dans le brouillard, alors que par temps clair et en utilisant notre système tous les sujets arrivent à détecter la voiture en des temps

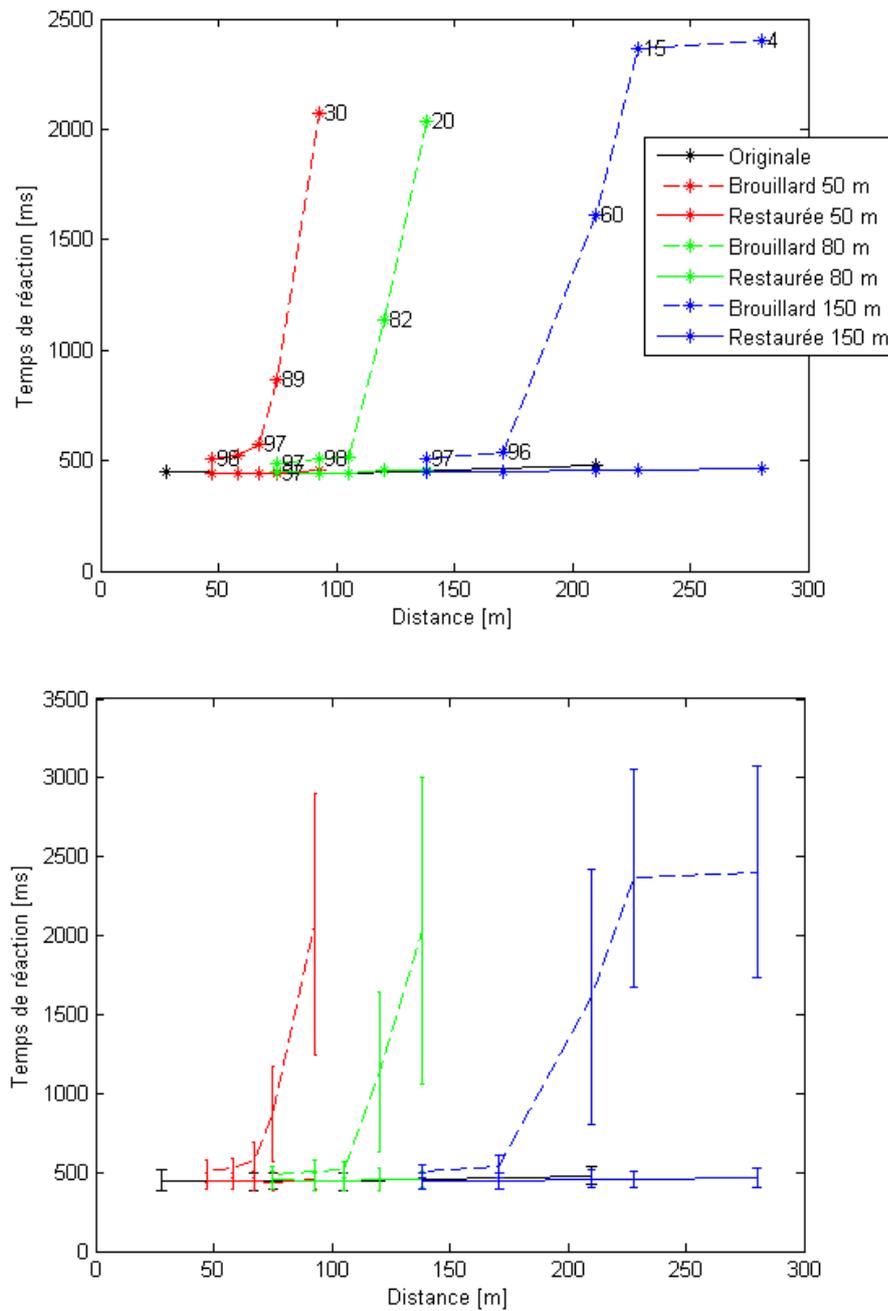


FIGURE 6.19 – Image du haut : temps de réaction en fonction de la distance. Courbe des images originales en noir, courbes des images de brouillard (en rouge $d_{Vis} = 50$ m, en vert $d_{Vis} = 80$ m et en bleu $d_{Vis} = 150$ m) en couleurs en lignes pointillées et courbes des images restaurées avec les mêmes couleurs en lignes continues. Le taux de détection moyen est affiché sur chaque distance de chaque courbe. Image du bas : temps de réaction en fonction de la distance avec les écarts types entre sujets pour chaque distance et chaque brouillard.

équivalents.

6.2.4 Conclusion

Nous avons présenté deux expériences de mesure de temps de réaction dans le brouillard.

Dans la première expérience, les résultats montrent que, dans le brouillard, le temps de perception d'un objet augmente en fonction de sa distance et que le taux de détection diminue. Nous constatons aussi qu'à partir de la distance de visibilité les performances de détection commencent à se dégrader. Cette expérience, bien que beaucoup plus simple, permet de retrouver les mêmes résultats qu'avec une scène plus réaliste (deuxième expérience). Aussi cette simplification permet d'envisager de nouvelles expériences en diminuant leurs temps de développement.

Dans la deuxième expérience, nous avons montré qu'en utilisant notre système de restauration, nous pouvons améliorer le temps de perception jusqu'à 2 secondes pour les différentes densités de brouillard utilisée, ainsi que le taux de détection jusqu'à 95 %. Notons que cette mesure correspond au gain maximum de temps de réaction qu'un HUD pourrait apporter. D'une part, nous avons étudié le cas idéal : les images de synthèse utilisées ne contiennent pas de bruit à la différence de la réalité (poussière et saleté sur le capteur ou le pare-brise, problème d'éclairage ...). L'utilisation d'images réelles dans une futur expérience, permettrait d'évaluer un gain qui sera plus proche de la réalité. D'autre part, de façon à mobiliser l'intégralité du gain potentiel en terme de temps de réaction, l'ergonomie du système HUD est critique. Il doit être conçu de façon à être facilement accessible au conducteur sans perturber sa tâche de conduite.

6.3 Traitement : Détection de l'espace navigable

Dans le chapitre 2, nous avons présenté la méthode [Hautière 2005] utilisant un seul capteur caméra et permettant de restaurer des images de scène routière de brouillard en supposant la route plane. Nous avons vu, dans le chapitre 4 la possibilité de combinaison de cette méthode avec notre méthode de restauration afin de mieux restaurer le plan de la route. Cette combinaison ayant été effectuée en segmentant la scène en objets verticaux et horizontaux. Une autre application de cette méthode, que nous allons présenter dans cette section, est l'utilisation de la segmentation de la route pour permettre la détection de la chaussée devant le véhicule. Cela permettra d'estimer les zones où le véhicule est capable de rouler de manière sûre (sans rencontrer des obstacles). Nous appelons cette zone l'espace navigable.

Notons que des méthodes de l'état de l'art utilisant la couleur [Crisman 1991], la texture [Zhang 1994], la stéréovision [Broggi 2005] ou une combinaison de ces approches [Soquet 2007], permettent de détecter l'espace navigable. Comme la luminance reçue par le capteur caméra s'atténue exponentiellement avec la distance dans le brouillard, ces méthodes risquent de donner des mauvais résultats. Une méthode permettant de restaurer les contrastes des images permettrait potentiellement d'améliorer les résultats de ces méthodes dans le brouillard.

L'approche proposée dans cette section permet la restauration et peut donc être combinée avec ses méthodes. En plus, elle permet la détection de l'espace navigable, dans le brouillard en exploitant les propriétés du brouillard et peut donc être assez complémentaire des méthodes classiques.

Le principe de la méthode est le suivant : nous commençons par estimer la distance de visibilité atmosphérique et la position de la ligne d'horizon. Ensuite l'image est restaurée et segmentée en utilisant l'hypothèse monde plan. Enfin, une segmentation des composantes connexes devant le véhicule nous permet de déduire l'espace navigable. La méthode a été optimisée en utilisant une base de données contenant 15 images de brouillard acquises à l'aide d'une caméra embarquée dans un véhicule.

6.3.1 Estimation de la distance de visibilité

Afin d'estimer la densité du brouillard, nous utilisons la méthode présentée dans le chapitre 2 qui consiste à chercher le point d'inflexion des intensités par rapport aux lignes de l'image. Nous rappelons la formule déduite de la densité de brouillard β :

$$\beta = \frac{2(v_i - v_h)}{\lambda} \quad (6.9)$$

avec v_i la position du point d'inflexion estimée sur une région d'intérêt de l'image (ROI) contenant une zone homogène de la route et la zone du ciel et possédant un gradient verticale minimum (inférieur à un seuil Δ_s). v_h est la position de la ligne d'horizon et λ un coefficient qui dépend des paramètres de la caméra (voir chapitre 2 pour plus de détails).

6.3.2 Discussion

Segmentation de la région d'intérêt

Notons que l'estimation de β est sensible à la présence d'obstacles tels que les véhicules qui empêchent la bonne détection de la bande verticale sur laquelle le point d'inflexion est estimé. Pour surmonter ce problème, un filtre temporel peut être utilisé. Dans [Hiramatsu 2009], un filtre de Kalman dédié à notre problème a été proposé.

Un autre problème de la méthode est le choix du seuil Δ_s . La méthode échoue dans le cas où la route est très texturée ou lorsque la transition entre la route et le ciel est très forte.

La figure 6.20 montre des exemples où la détection de la ROI échoue. Les images (a) et (b) montrent des scènes fortement texturées, l'image (c) correspond à une scène telle que la transition entre la route et le ciel est très forte et les images (d) et (e) montrent des images de synthèses contenant des objets verticaux (véhicules, arbres, pont).

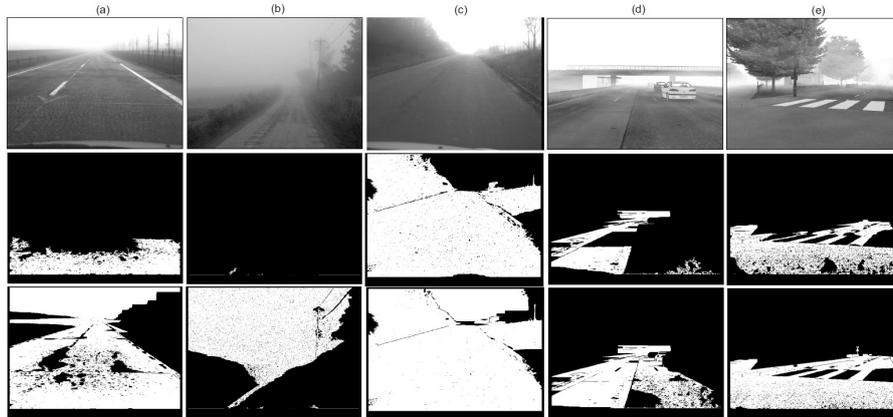


FIGURE 6.20 – Échec de la détection de la ROI où le point d'inflexion est estimé en utilisant un algorithme de croissance des régions. La première ligne montre des images de brouillard, la deuxième et la troisième lignes montrent les ROI détectées en utilisant respectivement des seuils $\Delta_s = 2$ et $\Delta_s = 3$.

Sensibilité au tangage

La méthode proposée est sensible à l'orientation du véhicule par rapport à la surface de la route. L'estimation de la distance de visibilité $V_{met} = \frac{3}{\beta}$ est correcte si l'estimation de v_i et v_h est correcte. Notons δ l'erreur entre v_i et v_h . Afin d'étudier l'influence de δ sur l'estimation de V_{met} , nous calculons l'erreur S entre la visibilité

météorologique estimée \hat{V}_{met} et la vraie valeur de V_{met} :

$$\begin{aligned}
 S &= V_{met} - \hat{V}_{met} \\
 &= V_{met} - \frac{3\lambda}{2} \frac{1}{v_i - v_h + \delta} \\
 &= V_{met} \left(1 - \frac{1}{1 + \frac{2\delta V_{met}}{3\lambda}} \right)
 \end{aligned} \tag{1.10}$$

Sur la figure 6.21 nous avons tracé la courbe de S en fonction de V_{met} pour $\delta \in [-4, 4]$. Nous constatons qu'une surestimation de δ est moins pénalisante qu'une sous-estimation. Afin d'avoir une mesure stable nous devons donc choisir la ligne d'horizon au dessus de sa valeur théorique.

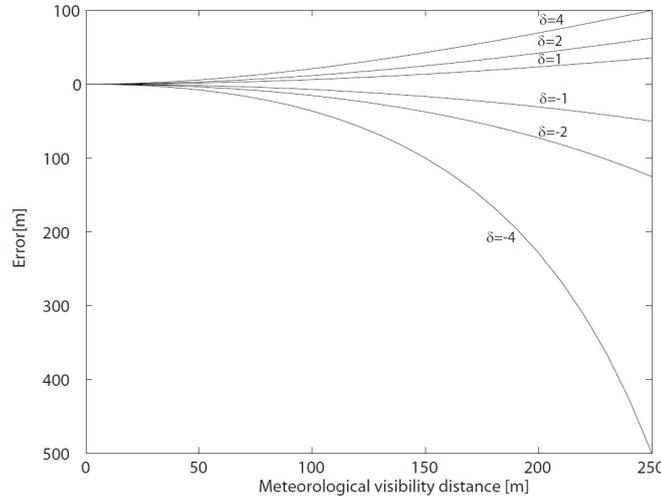


FIGURE 6.21 – Sensibilité de l'estimation de la visibilité météorologique V_{met} à l'erreur δ entre la position du point d'inflexion v_i et la position de la ligne d'horizon v_h . Une sous-estimation de δ est plus pénalisante qu'une surestimation.

Dans la suite nous allons voir comment les problèmes de la détection de la ROI sont surmontés.

6.3.3 Segmentation de la ROI fondée sur les cartes géodésique

La méthode de segmentation de la ROI de [Hautière 2006] peut être remplacée par d'autres méthodes de calcul de chemin optimal en assimilant l'image à un graphe. Dans l'état de l'art, il existe de nombreuses méthodes permettant de calculer le chemin le plus court tel que l'algorithme de Dijkstra [Dijkstra 1959] ou l'algorithme A^* [Hart 1968]. Cependant, elles sont coûteuses en temps de calcul. Dans notre cas nous avons utilisé l'algorithme [Brémond 1994, Ikonen 2005, Criminisi 2008] appelé transformée de distance géodésique efficace noté aussi WDTACS.

La distance géodésique $C(x, y)$ d'un pixel de coordonnées (x, y) est définie par l'équation sur la figure 6.22. Cette figure illustre le principe de de la WDTACS.

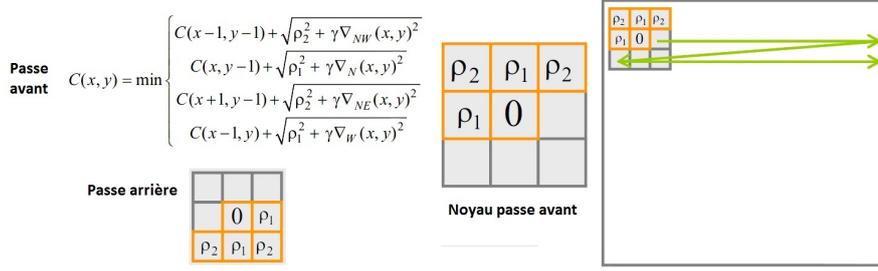


FIGURE 6.22 – L'image est parcourue de gauche à droite et de haut en bas et la distance géodésique $C(x, y)$ est calculée en utilisant un masque. Généralement, $\rho_1 = 1$ et $\rho_2 = \sqrt{2}$. γ est un coefficient de pondération des gradients calculé sur 4 directions.

L'algorithme requiert deux cartes de distance $\mathcal{F}_a^*(x)$ et $\mathcal{F}_b^*(x)$. L'extrémité a (respectivement b) du trajet est la caractéristique à partir de laquelle toutes les distances sont calculées.

La distance du chemin le plus court entre les extrémités a et b est :

$$\mathcal{D}_R(x) = \mathcal{F}_a^*(x) + \mathcal{F}_b^*(x) \quad (6.10)$$

Donc, les points correspondant à la distance minimal du chemin souhaité est :

$$\mathcal{R}(a, b) = \{x | \mathcal{D}_R(x) = \min_x \mathcal{D}_R(x)\} \quad (6.11)$$

Dans le cas d'images routière nous choisissons les extrémités A et B , respectivement, comme étant le bas de l'image (juste devant le véhicule) et le point le plus lumineux de l'image (le ciel). La figure 6.23 illustre l'application de cet algorithme sur une image de route.

La carte $\mathcal{R}(A, B)$ est calculée en fonction d'un paramètre de tolérance τ :

$$\mathcal{R}(A, B) = \{x | \mathcal{D}_R(x) \leq (1 + \tau) \min_x \mathcal{D}_R(x)\} \quad (6.12)$$

La segmentation finale est réussie si elle passe au dessus de la ligne d'horizon. Dans [Hautière 2011] une nouvelle méthode permettant l'estimation de la ligne d'horizon a également été proposée (nous ne la détaillerons pas ici).

Notons que la méthode rencontre encore des problèmes en présence de grands objets devant le véhicule. Pour plus de précision cette méthode peut être combiné avec un système GPS comme dans [Hautiere 2010a].

6.3.4 Détection de l'espace navigable

À partir du résultat de la segmentation de la ROI nous déduisons donc le point d'inflexion v_i . Ensuite, la densité du brouillard β est calculée à partir de l'équation 6.3.2. La restauration est effectuée en utilisant l'hypothèse monde plan et en inversant le modèle de Koschmieder. Enfin, la route est segmentée en objets verticaux et horizontaux comme dans le chapitre 4.

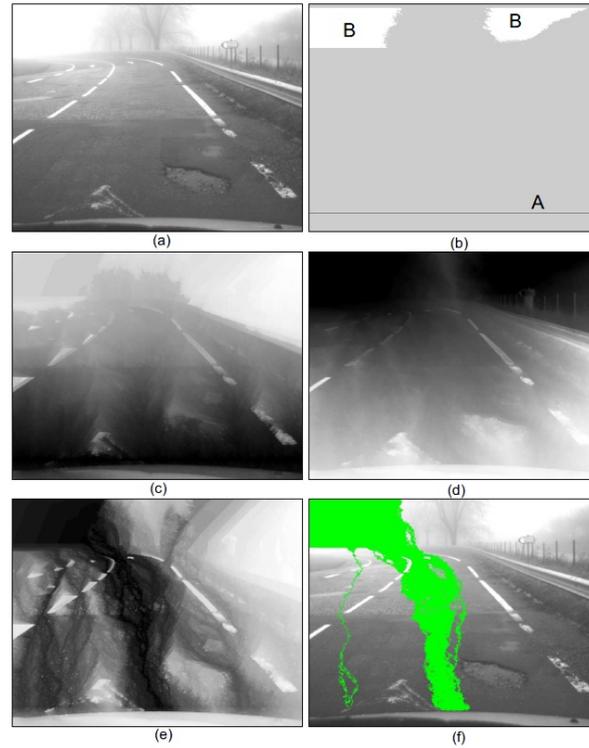


FIGURE 6.23 – Application de la WDTOCS sur une image de route. image originale (a), les ensembles des extrémités A et B (b), cartes de distance $\mathcal{F}_A^*(x)$ (c) et $\mathcal{F}_B^*(x)$ (d), distance minimale $\mathcal{R}(A, B)$ et le résultat de segmentation de la ROI en vert (f).

La figure 6.24 montre, en couleur verte, la détection de l'espace navigable considéré comme le résultat de segmentation des objets horizontaux. D'un point de vue qualitatif, nous obtenons un bon résultat, d'une qualité comparable à celle des approches stéréovision.

Afin d'évaluer quantitativement les résultats, nous avons créé manuellement, pour 15 images réelles de scènes routière, la vérité terrain correspondant à l'espace navigable. Ensuite, nous avons tracé la courbe ROC en fonction d'un coefficient ρ de pondération de la visibilité météorologique estimée.

Pendant la création de la vérité terrain, nous nous sommes demandé s'il fallait choisir l'espace navigable comme étant la route ou s'il fallait inclure d'autres zones telles que les trottoirs? Nous avons choisi la définition de l'espace navigable la plus restrictive : nous considérons seulement la surface de la route.

La figure 6.25 montre la courbe ROC que nous obtenons en traçant le taux de vrais positifs TPR en fonction du taux des faux positifs FPR . Le TPR est le taux des pixels appartenant à la route et détectés comme espace navigable et le FPR est le taux des objets verticaux détectés comme espace navigable.

Dans le cas où $\rho = 1$ (utilisation de V_{met} estimé) nous obtenons un TPR proche

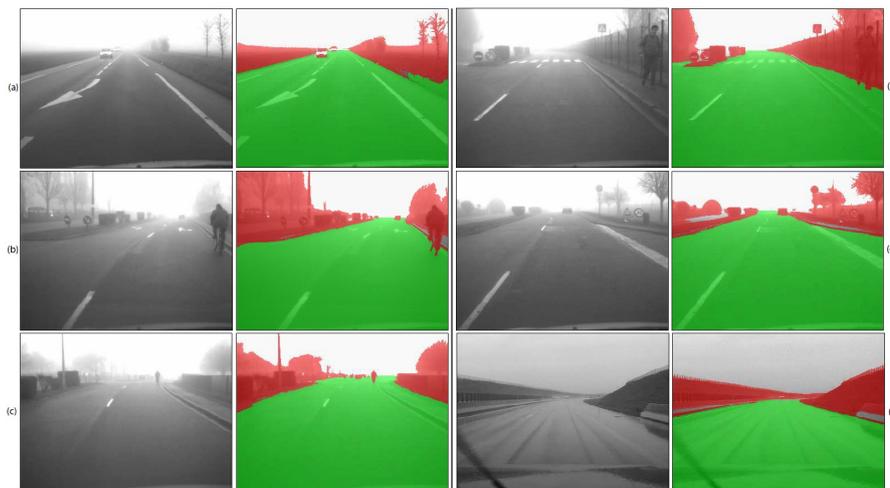


FIGURE 6.24 – Détection de l'espace navigable. Première et troisième colonnes : images originales. Deuxième et quatrième colonnes : segmentation en objets verticaux (zones rouge) et espace navigable (zones vertes). L'image en bas à droite correspond à une image de pluie (dans ce cas β a été choisi manuellement).

de 1 et un FPR d'environ 0,4. Nous avons donc un taux de faux positifs élevé.

Un coefficient $\rho < 1$ permet de réduire le FPR . En effet, en réduisant la visibilité météorologique la restauration du contraste est plus forte ce qui implique que certains zones telles que les trottoirs sont détectés comme obstacles.

Un coefficient $\rho = 0,6$ nous donne un $TPR = 95\%$ et un $FPR = 10\%$. Nous parvenant ainsi à optimiser la détection de l'espace navigable.

6.3.5 Conclusion

Nous avons présenté, dans cette section, une application de la restauration par hypothèse monde plan (en utilisant une version améliorée de l'estimation de la densité de brouillard [Hautière 2011]) permettant de détecter, dans le brouillard, l'espace navigable où le véhicule peut se déplacer d'une façon sécurisée sans rencontrer d'obstacle. La méthode a été évaluée et optimisée en utilisant une petite base de données de 15 images réelles. Sur cette base d'images, nous obtenons un taux de bonne détection assez élevé et un taux de mauvaise détection faible.

6.4 Conclusion

Au travers de ces trois expériences, nous avons pu montrer la pertinence de la restauration des images lorsqu'elles sont dégradées par du brouillard. Chacune d'entre-elles représente une famille d'application (pré-traitement, assistance, traitement). D'autres applications pourraient être testées pour confirmer ces résultats et inspirer, dans le futur, le développement de nouveaux systèmes d'assistance à la conduite.

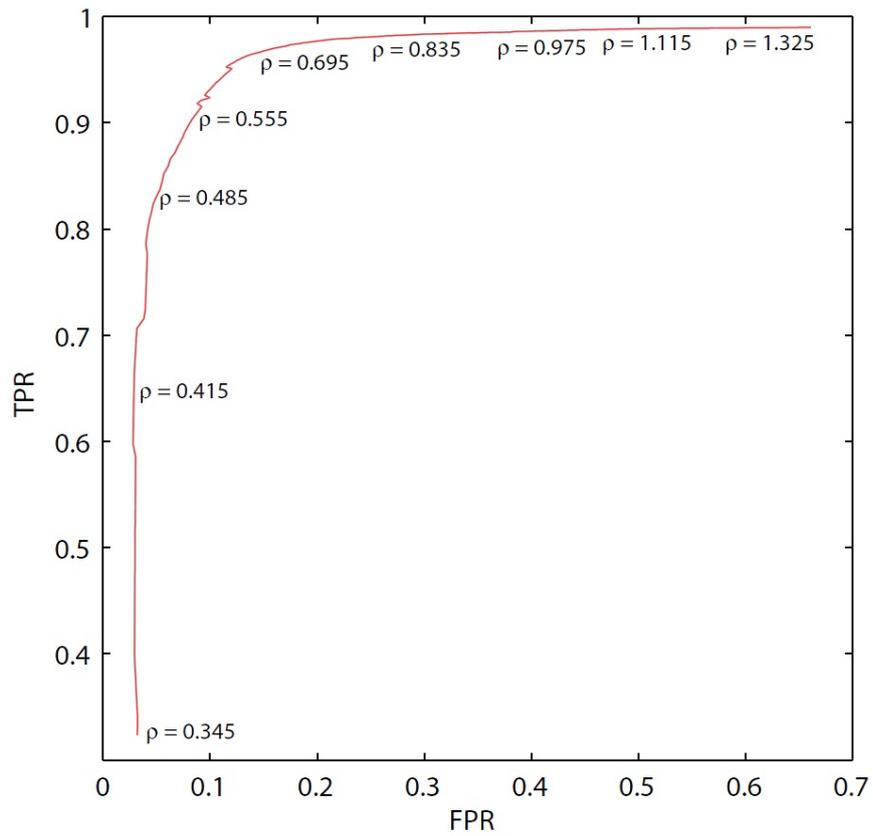


FIGURE 6.25 – Courbe ROC de la détection de l’espace navigable obtenue sur 15 images. Le taux de vrais positifs est tracé en fonction du taux des faux positifs en fonction d’un coefficient ρ qui pondère la visibilité météorologique.

Conclusion et perspectives

L'objectif principal des systèmes d'aide à la conduite est de pallier les défaillances du conducteur et augmenter la sécurité routière. Les nouveaux systèmes cherchent à anticiper au mieux pour prévenir les accidents, par l'exploitation de capteurs extéroceptifs permettant la perception de l'environnement. Les capteurs optiques tels que les caméras sont de plus en plus présents du fait de leur bas coût, de leur encombrement réduit et de la richesse des informations délivrées. Avec un seul capteur, la caméra, de nombreuses applications peuvent être réalisées. Or, les perturbations engendrées par des conditions météorologiques défavorables affectent la perception et, en particulier, le contraste et les couleurs dans la scène. Dans ces conditions, les algorithmes de traitement d'images peuvent fournir des résultats erronés.

Cette thèse traite le problème de la restauration en temps réel des images diurnes dégradées par le brouillard ou la pluie. Cette restauration permet ainsi, d'un point de vue technologique, d'étendre la gamme de fonctionnement des systèmes de vision actuels et d'un point de vue sociétal, de diminuer le nombre d'accidents, le tout en respectant la dimension économique qui consiste à diminuer le coût des systèmes embarquées en intégrant plusieurs fonctions au sein d'un seul capteur.

Contributions

Notre objectif méthodologique est le "deweathering", qui est, selon nous, en passe de devenir une discipline propre de la restauration d'images. Nous l'avons abordé en fusionnant des méthodes dédiées de détection et caractérisation de brouillard ou de pluie avec des méthodes classiques de traitement d'images adaptées à notre problème.

Dans l'état de l'art, le "deweathering" dans le cas du brouillard est un sujet qui a été abordé depuis une dizaine d'années. Nous avons présenté différentes méthodes (interactives et automatiques), s'appuyant sur le modèle optique du brouillard, permettant de résoudre le problème. Parmi ces méthodes, nous avons sélectionné celles qui sont les plus appropriées aux ADAS suivant des critères de temps de traitement, de qualité de la restauration et d'utilisation d'un seul capteur caméra. D'autres méthodes n'utilisant pas de modèle physique ont aussi été sélectionnées. Les différentes méthodes ont été évaluées qualitativement et quantitativement grâce à des bases d'images de synthèse et réelles.

Nous avons proposé une méthode de restauration du brouillard combinant approche physique, fondée sur le modèle optique du brouillard et approche signal, fondée sur l'égalisation d'histogramme locale. Afin de réduire les artefacts générés lors du traitement, différentes méthodes de débruitage ont été étudiées afin de sélectionner la méthode la plus adaptée à notre algorithme de restauration. Cette approche, évaluée sur des bases de données de référence, présente une amélioration significative par rapport aux méthodes existantes.

Concernant le “deweathering” des gouttes de pluie présentes sur le pare-brise, très peu d'études ont été effectuées à ce sujet. Nous avons proposé une nouvelle approche utilisant une méthode évoluée d’“inpainting” par diffusion anisotropique qui, après la détection des gouttes dans l'image, permet la restauration. Les paramètres de la méthode ont été optimisés par un algorithme de recuit simulé en utilisant une base d'images routières acquises par caméra embarquée.

Trois familles d'applications ADAS (traitement, pré-traitement et assistance) ont été proposées afin d'évaluer les méthodes de restauration d'images de brouillard. Dans chaque famille, nous avons testé une application. Nous avons évalué les gains apportés par notre système de restauration à un algorithme de détection de panneaux ainsi qu'à la perception du conducteur dans le brouillard. Des bases d'images de synthèse ont été utilisées pour permettre l'évaluation. Les résultats obtenus montrent une amélioration significative de la perception humaine et du traitement logiciel. Enfin, nous avons évalué et optimisé les performances d'un système de détection de l'espace navigable dans le brouillard en utilisant une base d'images réelles.

Les ADAS qui semblent le plus perturbé par le brouillard sont les systèmes de détection d'obstacles, de véhicules et de panneaux, puisque ces objets sont recherchés sur une large gamme de distance et que le brouillard dégrade fortement leur contraste et couleur lorsqu'ils sont éloignés. De même, le suivi de véhicule dans le brouillard risque de ne pas être efficace à cause de la variation d'intensité des objets avec la distance. D'un autre côté, la restauration risque de ne pas apporter des gains importants dans le cas de détection de marquages, puisque ces derniers ne sont recherchés que dans la partie inférieure de l'image qui est la moins dégradée par le brouillard, sauf dans le cas de brouillard extrême.

Perspectives

Différentes améliorations peuvent être apportées aux méthodes de restauration proposées et d'autres applications pour des ADAS ou, plus généralement, pour la vision par ordinateur sont envisageables.

L'algorithme de restauration d'images de brouillard a été développé de façon modulaire. Dans le futur, d'autres méthodes de lissage et de restauration de contraste et des couleurs qui seront plus efficaces et plus rapides pourraient remplacer les modules concernés. Ces méthodes devraient permettre de réduire le bruit et d'améliorer les couleurs afin d'obtenir une meilleure restauration sur les différentes zones

de l'image. En particulier, le module de lissage de la carte de transmittance, pourrait être remplacé par une méthode de diffusion anisotropique, telle que celle utilisée pour effectuer l'"inpainting".

D'un autre côté, nous avons bien noté que pour améliorer le contraste des objets de la scène, le choix de la taille de la fenêtre d'égalisation n'est pas aisé. Afin d'obtenir des meilleurs contrastes, nous pourrions adapter la taille de la fenêtre suivant les caractéristiques de la zone à traiter. Par exemple, utiliser une segmentation de l'image permettra d'adapter la taille en fonction du nombre des régions détectées : grande taille dans les zones uniformes et petite taille dans les zones texturées.

Pour une meilleure estimation de la carte de profondeur de la scène dans le brouillard, nous pourrions exploiter la segmentation et l'estimation de la profondeur de l'espace roulable afin de transférer la profondeur du plan de la route vers les objets verticaux. Pour cela, nous pourrions utiliser des méthodes de minimisation globale d'énergie, tels que les Graphcut, avec un a priori sur l'augmentation de la distance avec la hauteur de l'image.

L'évaluation sur des ADAS (détection de panneaux et mesure de temps de réaction) en utilisant la restauration sur des images ou des séquences réelles pourrait être encore approfondie. Cela permettra d'étudier la différence entre l'utilisation d'images de synthèses et les images réelles.

Dans le cas de mesure de temps de réaction, nous pourrions compliquer la tâche en mobilisant l'attention du sujet. Par exemple, il est possible d'ajouter un signal sonore ou de lui demander d'effectuer d'autres tâches (recherche de l'objet dans l'image, faire un calcul mental), afin de rendre l'expérience plus réaliste et d'analyser l'impact sur le temps de réaction.

Selon nous, il serait pertinent de mettre en place une expérience de mesure de la distance d'arrêt dans le brouillard en présence d'obstacles sur la voie, en utilisant un simulateur de conduite avec et sans HUD affichant l'image restaurée au sujet.

Une implantation sur des tablettes ou des Smartphone pourra contribuer à l'amélioration des ADAS existant sur ce type de plateformes telle que la détection de panneaux ou de véhicules. L'image restaurée pourra aussi être affichée au conducteur en utilisant ces systèmes nomades comme HUD.

Concernant le problème de restauration d'images de pluie, nous avons vu que, l'information que nous cherchons à restaurer n'est pas présente dans l'image. À la différence du brouillard, cette absence d'information est indépendante de la distance : des objets très proches de la caméra peuvent être masqués par les gouttes. Comme la méthode proposée utilise uniquement l'information provenant du voisinage spatial de la goutte cette information risque, dans les cas où la goutte couvre plusieurs objets, de ne pas être pertinente. L'utilisation du flux temporel pourra être exploité pour améliorer l'"inpainting".

La plupart des ADAS par caméra risquent d'être perturbés par la pluie. En effet comme c'est un phénomène qui a pour effet de masquer des parties de l'image, cela va dégrader les performances des algorithmes de traitement d'images à cause de l'absence d'information. La restauration proposée permet la reconstruction des

structures géométriques tels que les marquages et permettra donc potentiellement d'améliorer les systèmes de détection de marquages dans la pluie.

Pour des systèmes de détection de panneaux, la structure des panneaux (triangle, cercle...) pourra être reconstruite mais pas les détails inscrits dans les panneaux. La restauration pourra donc apporter un gain important pour la détection (si elle s'appuie sur la géométrie des structures) mais plus faible pour la reconnaissance.

Nous pourrons aussi tester la restauration sur des images dégradées simultanément par le brouillard et la pluie (pluie fine qui crée du brouillard) ainsi que sur d'autres types de dégradation tels que la neige.

La restauration peut aussi être utilisée pour d'autres applications hors ADAS : reconstruction stéréo dans le brouillard, segmentation, "structure from motion"... nous pourrons ainsi savoir quand la restauration apporte un gain aux systèmes et d'identifier les parties à améliorer suivant l'application souhaitée.

Dans le domaine routier, des applications telles que l'utilisation d'un HUD pour les voitures de patrouilles sur autoroute dans le cas de brouillard ou pour des caméras de détection de véhicule sur la route (système de transport intelligent) pourra potentiellement améliorer la sécurité routière.

Enfin, nous pouvons aussi envisager l'implantation de la restauration sur des appareils photos, des jumelles pour des fins militaires ou dans d'autres champs d'applications : surveillance des ports et des aéroports, cartographie terrestre, vidéo surveillance.

“Inpainting”

A.1 Illustration de la méthode d’inpainting retenue par des exemples

Nous commençons par le cas le plus simple où le masque partage l’image en parties symétriques comme le montre la figure A.1 qui affiche l’image avec la zone à remplir en blanc.

À partir de l’image dégradée toute seule nous ne savons pas quelle doit être la forme à restaurer. Nous allons commencer par montrer ce qui se passe sans effectuer de

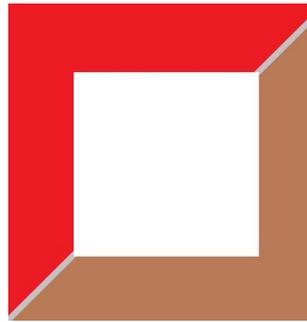


FIGURE A.1 – Exemple d’illustration

lissage. Nous fixons la taille du cercle tel que $\varepsilon = 3$ pixels (cercle de rayon 1,5 pixels), le coefficient $\kappa = 3$ et la taille du lissage à zéro (pas de lissage avec σ et ρ). La figure A.2 montre le flux de cohérence initial, le flux de cohérence final et le résultat de la restauration. Nous remarquons que nous avons le problème des bords cité dans le chapitre 5 à cause de l’absence de lissage et que nous négligeons certains détails (ligne grise) à cause de la petite taille de voisinage et à cause de leur faible cohérence. Choisissons maintenant une taille de cercle plus grande $\varepsilon = 9$, un coefficient $\kappa = 150$ pour récupérer une plus grande quantité d’information, un lissage tel que $\sigma = 1,4$ pour éviter le problème des bords sans dégrader les détails (σ petit) et $\rho = 50$ pour avoir un tenseur plus lisse permettant de récupérer une grande quantité d’information. La figure A.3 montre les résultats obtenus. Nous constatons que nous n’avons plus le problème des bords et que nous arrivons à joindre les deux lignes grises. Nous considérons maintenant la même image qui cette fois n’est pas parfaitement symétrique par rapport au masque qui est aussi de plus petite taille. La figure A.5 montre les résultats obtenus en utilisant les mêmes paramètres que précédemment. Nous constatons que, malgré que la taille de la zone à remplir

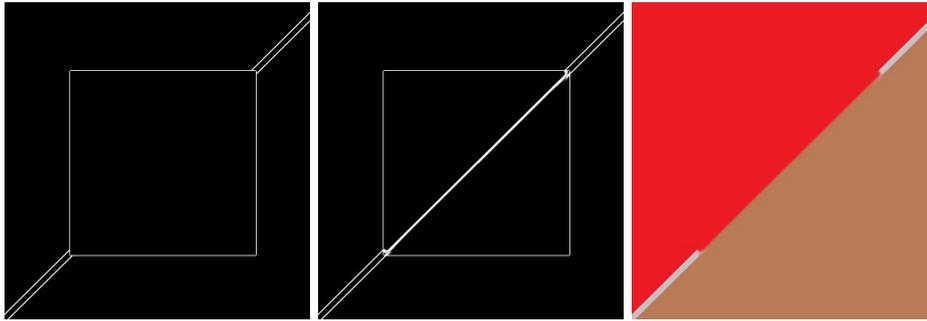


FIGURE A.2 – flux de cohérence initial, flux de cohérence final et restauration tel que $\varepsilon = 3$, $\kappa = 3$ et aucun lissage n'est effectué.

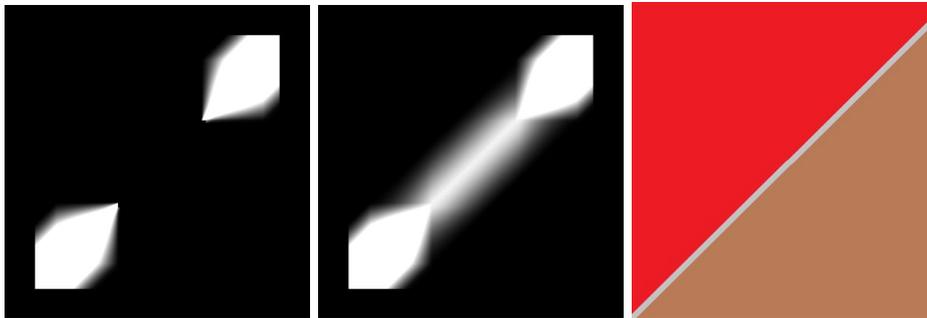


FIGURE A.3 – flux de cohérence initial, flux de cohérence final et restauration tel que $\varepsilon = 9$, $\kappa = 150$, $\sigma = 1,4$ et $\rho = 50$.

est plus petite qu'avant, les lignes grises ne se rejoignent pas. Pour surmonter ce problème il faut modifier les paramètres : taille de lissage plus grande pour une meilleure détection des directions de la diffusion. Dans le cas où le masque est plus grand il nous faudra encore agrandir la taille du lissage.

Nous voyons donc, par cet exemple, l'influence que vont exercer la forme et la taille du masque sur l'"inpainting". Nous montrons sur la figure A.5 le résultat

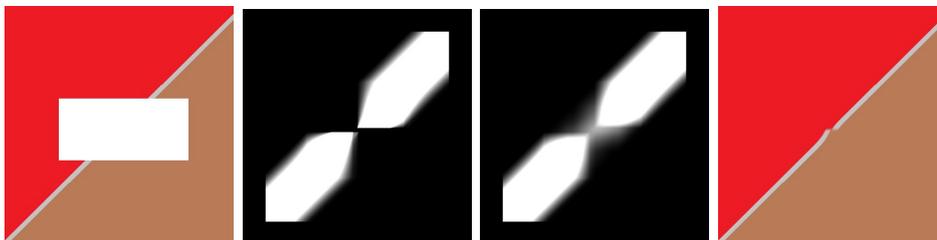


FIGURE A.4 – Image avec zone à remplir en blanc, flux de cohérence initial, flux de cohérence final et restauration tel que $\varepsilon = 9$, $\kappa = 150$, $\sigma = 1,4$ et $\rho = 50$.

obtenu sur une autre image. Sur la figure A.6 nous montrons un autre exemple de restauration. Nous constatons que nous arrivons à remplir les différentes zones de

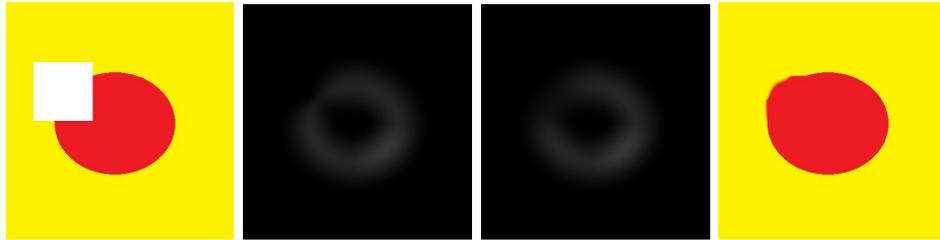


FIGURE A.5 – Image avec zone à remplir en blanc, flux de cohérence initial, flux de cohérence final et restauration tel que $\varepsilon = 9$, $\kappa = 150$, $\sigma = 1,4$ et $\rho = 50$.

l'image et nous obtenons donc une image proche de l'originale.

A.2 Inpainting pour le débruitage

L'image A.7 montre une image dans laquelle le bruit poivre et sel couvre 80% de l'image. Le résultat de l'"inpainting" [Bornemann 2007] montre que nous arrivons à retrouver les différentes structures de l'image et éliminer le bruit. Comparée à une méthode classique de dé-bruitage telle que le filtre médian, comme le montre la figure, l'"inpainting" permet le dé-bruitage sans altérer les contours.

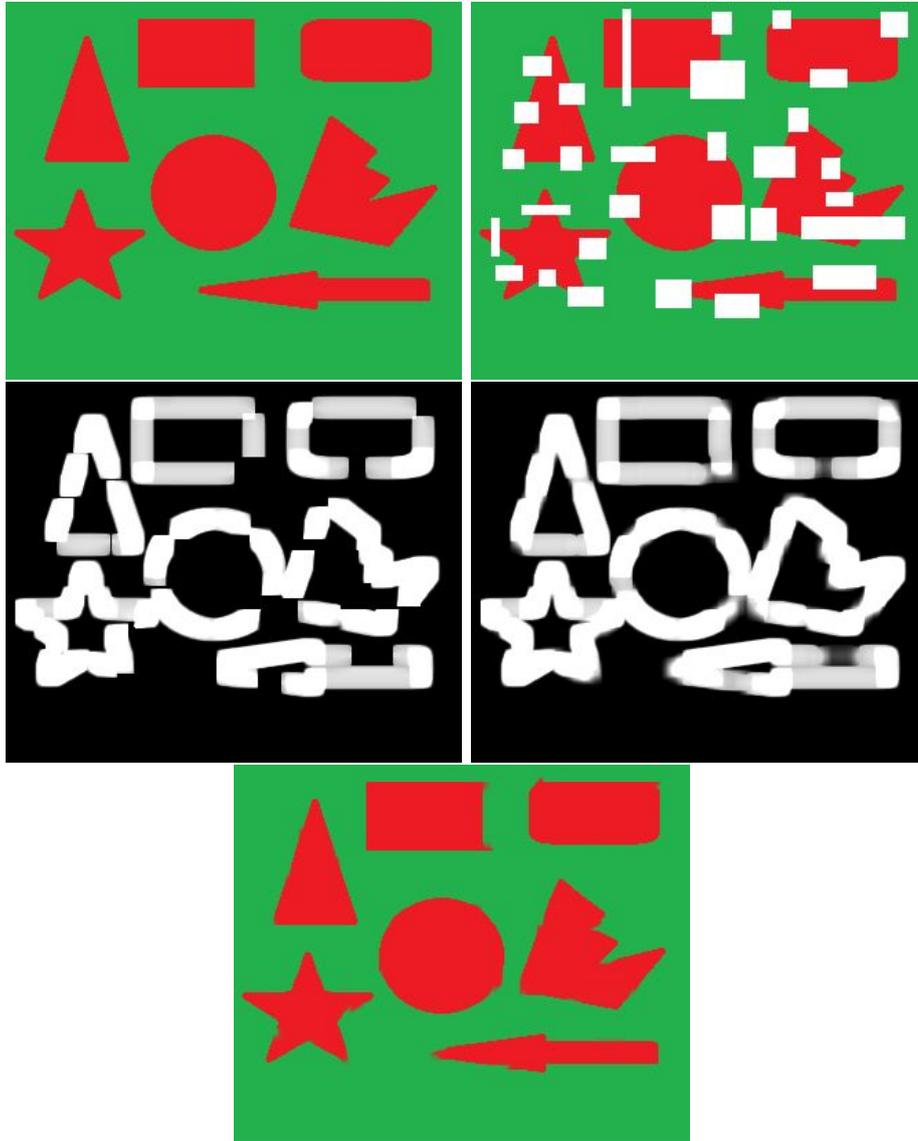


FIGURE A.6 – Image originale, image avec zone à remplir en blanc, flux de cohérence initial, flux de cohérence final et restauration tel que $\varepsilon = 7$, $\kappa = 150$, $\sigma = 1,4$ et $\rho = 7$.



FIGURE A.7 – Image originale bruitée à 80%, débruitage à l'aide d'un filtre médian et débruitage en utilisant l'"inpainting".

Bibliographie

- [Acharya 2005] T. Acharya et A.K. Ray. *Image processing : principles and applications*. Wiley-Interscience, 2005. (Cité en page 22.)
- [Adrian 1989] W. Adrian. *Visibility of targets : Model for calculation*. *Lighting Research and Technology*, vol. 21, no. 4, pages 181–188, 1989. (Cité en page 141.)
- [Barash 2002] D. Barash. *A Fundamental Relationship between Bilateral Filtering, Adaptive Smoothing, and the Nonlinear Diffusion Equation*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 844–847, 2002. (Cité en page 16.)
- [Barham 2001] P. Barham, L. Andreone, A. Toffetti, D. Bertolino et J. Eschler. *Changes to driving behaviour in conditions of reduced visibility when using an infrared vision support system-results of evaluations on a driving simulator*. In *International conference on traffic and transport psychology-icttp 2000*, held 4-7 september 2000, berne, Switzerland-keynotes, symposia, thematic sessions, workshops, posters, list of participants and word viewer-cd rom, 2001. (Cité en page 4.)
- [Beard 1987] K.V. Beard et C. Chuang. *A new model for the equilibrium shape of raindrops*. *Journal of the Atmospheric sciences*, vol. 44, no. 11, pages 1509–1524, 1987. (Cité en page 54.)
- [Belaroussi 2009] R. Belaroussi et J.P. Tarel. *A real-time road sign detection using bilateral chinese transform*. *Advances in Visual Computing*, pages 1161–1170, 2009. (Cité en pages 126 et 131.)
- [Belaroussi 2010] R. Belaroussi, P. Foucher, J.-P. Tarel, B. Soheilian, P. Charbonnier et N. Paparoditis. *Road Sign Detection in Images : A Case Study*. In *International Conference on Pattern Recognition (ICPR'10)*, pages 484–488, 2010. (Cité en page 131.)
- [Bertalmio 2000] M. Bertalmio, G. Sapiro, V. Caselles et C. Ballester. *Image inpainting*. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424. ACM Press/Addison-Wesley Publishing Co., 2000. (Cité en pages 104, 105 et 107.)
- [Bornemann 2007] F. Bornemann et T. März. *Fast image inpainting based on coherence transport*. *Journal of Mathematical Imaging and Vision*, vol. 28, no. 3, pages 259–278, 2007. (Cité en pages 104, 113, 115, 116, 118, 120, 123 et 163.)
- [Boubchir 2005] L. Boubchir et J.M. Fadili. *Bayesian denoising based on the map estimation in wavelet-domain using bessel k form prior*. In *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, volume 1, pages I–113. IEEE, 2005. (Cité en page 21.)

- [Brémond 1994] R. Brémond, D. Jeulin, P. Gateau, J. Jarrin et G. Serpe. *Estimation of the transport properties of polymer composites by geodesic propagation*. Journal of Microscopy, vol. 176, no. 2, pages 167–177, 1994. (Cité en page 151.)
- [Broggi 2005] A. Broggi, C. Caraffi, R.I. Fedriga et P. Grisleri. *Obstacle detection with stereo vision for off-road vehicle navigation*. In Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on, pages 65–65. IEEE, 2005. (Cité en page 149.)
- [Buchsbaum 1980] G. Buchsbaum. *A spatial processor model for object colour perception*. journal of the Franklin institute, vol. 310, no. 1, pages 1–26, 1980. (Cité en page 26.)
- [Catté 1992] F. Catté, P.L. Lions, J.M. Morel et T. Coll. *Image selective smoothing and edge detection by nonlinear diffusion*. SIAM Journal on Numerical Analysis, vol. 29, no. 1, pages 182–193, 1992. (Cité en page 18.)
- [Cavallo 2000] V. Cavallo, M. Colomb et J. Dore. *The overestimation of headways in fog*. Recherche Transports Securite, vol. 2000, no. 66, pages 81–99, 2000. (Cité en page 3.)
- [Chambolle 2010] A. Chambolle, V. Caselles, D. Cremers, M. Novaga et T. Pock. *An introduction to total variation for image analysis*. Theoretical Foundations and Numerical Methods for Sparse Recovery, De Gruyter, Radon Series Comp. Appl. Math, vol. 9, pages 263–340, 2010. (Cité en page 19.)
- [Charbonnier 1994] P. Charbonnier. *Reconstruction d'image : Régularisation avec prise en compte des discontinuités*. PhD thesis, 1994. (Cité en page 21.)
- [Comaniciu 2002] D. Comaniciu et P. Meer. *Mean shift : A robust approach toward feature space analysis*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 24, no. 5, pages 603–619, 2002. (Cité en page 109.)
- [Cord 2011] A. Cord et D. Aubert. *Towards rain detection through use of in-vehicle multipurpose cameras*. In Intelligent Vehicles Symposium (IV), 2011 IEEE, pages 833–838. IEEE, 2011. (Cité en pages 56, 58 et 103.)
- [Criminisi 2004] A. Criminisi, P. Pérez et K. Toyama. *Region filling and object removal by exemplar-based image inpainting*. Image Processing, IEEE Transactions on, vol. 13, no. 9, pages 1200–1212, 2004. (Cité en pages 104, 108 et 111.)
- [Criminisi 2008] A. Criminisi, T. Sharp et A. Blake. *Geos : Geodesic image segmentation*. Computer Vision–ECCV 2008, pages 99–112, 2008. (Cité en page 151.)
- [Crisman 1991] J.D. Crisman et C.E. Thorpe. *Unscarf-a color vision system for the detection of unstructured roads*. In Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on, pages 2496–2501. IEEE, 1991. (Cité en page 149.)

- [Dijkstra 1959] E.W. Dijkstra. *A note on two problems in connexion with graphs*. Numerische mathematik, vol. 1, no. 1, pages 269–271, 1959. (Cité en page 151.)
- [Do 2006] V. Do, G. Lebrun, L. Malapert, C. Smet, D. Tschumperlé et al. *Inpainting d’Images Couleurs par Lissage Anisotrope et Synthèse de Textures*. Actes de Reconnaissance des Formes et Intelligence Artificielle (RFIA’06), pages 35–42, 2006. (Cité en page 104.)
- [Efros 2001] A. Efros et W.T. Freeman. *Image quilting for texture synthesis and transfer*. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pages 341–346. ACM, 2001. (Non cité.)
- [Elango 2009] P. Elango et K. Murugesan. *Digital Image Inpainting Using Cellular Neural Network*. Int. J. Open Problems Compt. Math, vol. 2, no. 3, pages 439–450, 2009. (Cité en page 104.)
- [Fleishman 2003] S. Fleishman, I. Drori et D. Cohen-Or. *Bilateral mesh denoising*. In ACM Transactions on Graphics (TOG), volume 22, pages 950–953. ACM, 2003. (Cité en page 15.)
- [Garg 2007] K. Garg et S.K. Nayar. *Vision and rain*. International Journal of Computer Vision, vol. 75, no. 1, pages 3–27, 2007. (Cité en pages 54 et 56.)
- [Geiger 2010] Andreas Geiger, Martin Roser et Raquel Urtasun. *Efficient Large-Scale Stereo Matching*. In Asian Conference on Computer Vision, Queenstown, New Zealand, November 2010. (Cité en page 65.)
- [Geiger 2011a] A. Geiger, M. Roser et R. Urtasun. *Efficient large-scale stereo matching*. Computer Vision–ACCV 2010, pages 25–38, 2011. (Cité en page 65.)
- [Geiger 2011b] Andreas Geiger, Julius Ziegler et Christoph Stiller. *StereoScan : Dense 3d Reconstruction in Real-time*. In IEEE Intelligent Vehicles Symposium, Baden-Baden, Germany, June 2011. (Cité en page 65.)
- [Geman 1984] S. Geman et D. Geman. *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, no. 6, pages 721–741, 1984. (Cité en page 21.)
- [Ghael 1997] S.P. Ghael, A.M. Sayeed et R.G. Baraniuk. *Improved wavelet denoising via empirical Wiener filtering*. 1997. (Cité en page 15.)
- [Graham 1962] R. Graham. *Snow removal—A noise-stripping process for picture signals*. Information Theory, IRE Transactions on, vol. 8, no. 2, pages 129–144, 1962. (Cité en page 18.)
- [Gruyer 2009] D. Gruyer, S. Glaser et B. Monnier. *Simulation of vehicle automatic speed control by transponder-equipped infrastructure*. Proceedings of IEEE ITST, pages 20–22, 2009. (Cité en page 61.)
- [Halmaoui 2011] H. Halmaoui, A. Cord et N. Hautiere. *Contrast restoration of road images taken in foggy weather*. In Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, pages 2057–2063. IEEE, 2011. (Cité en pages 63, 64 et 72.)

- [Hart 1968] P.E. Hart, N.J. Nilsson et B. Raphael. *A formal basis for the heuristic determination of minimum cost paths*. Systems Science and Cybernetics, IEEE Transactions on, vol. 4, no. 2, pages 100–107, 1968. (Cité en page 151.)
- [Hautière 2006] N. Hautière, J.P. Tarel, J. Lavenant et D. Aubert. *Automatic fog detection and estimation of visibility distance through use of an onboard camera*. Machine Vision and Applications, vol. 17, no. 1, pages 8–20, 2006. (Cité en page 151.)
- [Hautière 2007] N. Hautière, J.P. Tarel et D. Aubert. *Towards fog-free in-vehicle vision systems through contrast restoration*. In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, pages 1–8. IEEE, 2007. (Cité en pages 43, 45, 71, 72, 76 et 89.)
- [Hautière 2008] N. Hautière, J.P. Tarel, D. Aubert, É. Dumont *et al.* *Blind contrast enhancement assessment by gradient ratioing at visible edges*. Image Analysis & Stereology Journal, vol. 27, no. 2, pages 87–95, 2008. (Cité en page 69.)
- [Hautiere 2010a] N. Hautiere, R. Belaroussi, J.P. Tarelet *et al.* *Assistance a la navigation automobile par temps de brouillard, In Projet DIVAS*. 2010. (Cité en page 152.)
- [Hautière 2010b] N. Hautière, J.P. Tarel et D. Aubert. *Mitigation of visibility loss for advanced camera-based driver assistance*. Intelligent Transportation Systems, IEEE Transactions on, vol. 11, no. 2, pages 474–484, 2010. (Cité en page 126.)
- [Hautière 2011] N. Hautière, J.P. Tarel, H. Halmaoui, R. Brémond et D. Aubert. *Enhanced fog detection and free-space segmentation for car navigation*. Machine Vision and Applications, pages 1–13, 2011. (Cité en pages 65, 152 et 154.)
- [Hautière 2003] N. Hautière et D. Aubert. *Driving Assistance : Automatic Fog Detection and Measure of the Visibility Distance*. In ITS World Congress (ITS'03), Madrid, Spain, November 2003. (Cité en page 132.)
- [Hautière 2005] N. Hautière et D. Aubert. *Contrast Restoration of Foggy Images through use of an Onboard Camera*. In IEEE Conference on Intelligent Transportation Systems (ITSC'05), pages 1090–1095, Vienna, Austria, September 2005. (Cité en page 149.)
- [He 2009] K. He, J. Sun et X. Tang. *Single image haze removal using dark channel prior*. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, pages 1956–1963. IEEE, 2009. (Cité en pages 49 et 52.)
- [Hiramatsu 2009] T. Hiramatsu, T. Ogawa et M. Haseyama. *A kalman filter-based method for restoration of images obtained by an in-vehicle camera in foggy conditions*. IEICE transactions on fundamentals of electronics, communications and computer sciences, vol. 92, no. 2, pages 577–584, 2009. (Cité en page 150.)

- [Hummel 1977] R. Hummel. *Image enhancement by histogram transformation*. Computer graphics and image processing, vol. 6, no. 2, pages 184–195, 1977. (Cité en page 22.)
- [Hunt 1977] B.R. Hunt. *Bayesian methods in nonlinear digital image restoration*. Computers, IEEE Transactions on, vol. 100, no. 3, pages 219–229, 1977. (Cité en page 20.)
- [Hwang 1988] C.R. Hwang. *Simulated annealing : theory and applications*. Acta Applicandae Mathematicae, vol. 12, no. 1, pages 108–111, 1988. (Cité en page 86.)
- [Ikonen 2005] L. Ikonen et P. Toivanen. *Shortest routes between sets on gray-level surfaces*. Pattern Recognition and Image Analysis, vol. 15, no. 1, pages 195–198, 2005. (Cité en page 151.)
- [Jobson 1995] D.J. Jobson et G.A. Woodell. *Properties of a center/surround Retinex : Part 2. Surround design*. NASA Technical Memorandum, vol. 110188, page 15, 1995. (Cité en page 23.)
- [Jobson 1997] D.J. Jobson, Z. Rahman et G.A. Woodell. *A multiscale retinex for bridging the gap between color images and the human observation of scenes*. Image Processing, IEEE Transactions on, vol. 6, no. 7, pages 965–976, 1997. (Cité en page 25.)
- [Kim 2001] J.Y. Kim, L.S. Kim et S.H. Hwang. *An advanced contrast enhancement using partially overlapped sub-block histogram equalization*. Circuits and Systems for Video Technology, IEEE Transactions on, vol. 11, no. 4, pages 475–484, 2001. (Cité en page 23.)
- [Komodakis 2006] N. Komodakis. *Image completion using global optimization*. In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, volume 1, pages 442–452. IEEE, 2006. (Cité en page 104.)
- [Kurihata 2005] H. Kurihata, T. Takahashi, I. Ide, Y. Mekada, H. Murase, Y. Tamatsu et T. Miyahara. *Rainy weather recognition from in-vehicle camera images for driver assistance*. In Intelligent Vehicles Symposium, 2005. Proceedings. IEEE, pages 205–210. IEEE, 2005. (Cité en page 56.)
- [Kwatra 2003] V. Kwatra, A. Schödl, I. Essa, G. Turk et A. Bobick. *Graphcut textures : image and video synthesis using graph cuts*. In ACM Transactions on Graphics (TOG), volume 22, pages 277–286. ACM, 2003. (Non cité.)
- [Lagendijk 1999] R.L. Lagendijk et J. Biemond. *Basic methods for image restoration and identification*. Academic, 2000).[Citation Context]... or image restoration include deconvolution, use of least-squares filters, and iterative approaches.[14]...[14], 1999. (Cité en page 15.)
- [Lam 2005] E.Y. Lam. *Combining gray world and retinex theory for automatic white balance in digital photography*. In Consumer Electronics, 2005.(ISCE 2005). Proceedings of the Ninth International Symposium on, pages 134–139. IEEE, 2005. (Cité en page 26.)

- [Loizou 2005] C.P. Loizou, C.S. Pattichis, C.I. Christodoulou, R.S.H. Istepanian, M. Pantziaris et A. Nicolaides. *Comparative evaluation of despeckle filtering in ultrasound imaging of the carotid artery*. Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on, vol. 52, no. 10, pages 1653–1669, 2005. (Cité en page 13.)
- [Middleton 1983] P. Middleton, T.R. Stewart et R.L. Dennis. *Modeling human judgments of urban visual air quality*. Atmospheric Environment (1967), vol. 17, no. 5, pages 1015–1021, 1983. (Cité en page 32.)
- [Narasimhan 2000] S.G. Narasimhan et S.K. Nayar. *Chromatic framework for vision in bad weather*. In Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on, volume 1, pages 598–605. IEEE, 2000. (Cité en page 40.)
- [Narasimhan 2002] S.G. Narasimhan et S.K. Nayar. *Vision and the atmosphere*. International Journal of Computer Vision, vol. 48, no. 3, pages 233–254, 2002. (Cité en page 29.)
- [Narasimhan 2003a] S.G. Narasimhan. *Models and algorithms for vision through the atmosphere*. PhD thesis, Citeseer, 2003. (Cité en page 42.)
- [Narasimhan 2003b] S.G. Narasimhan et S.K. Nayar. *Contrast restoration of weather degraded images*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 25, no. 6, pages 713–724, 2003. (Cité en page 36.)
- [Narasimhan 2003c] S.G. Narasimhan et S.K. Nayar. *Interactive (de) weathering of an image using physical models*. In IEEE Workshop on Color and Photometric Methods in Computer Vision. France, 2003. (Cité en page 40.)
- [Nashashibi 2010] F. Nashashibi, R. de Charrette et A. Lia. *Detection of unfocused raindrops on a windscreen using low level image processing*. In Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on, pages 1410–1415. IEEE, 2010. (Cité en page 56.)
- [Nayar 1999] S.K. Nayar et S.G. Narasimhan. *Vision in bad weather*. In Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, volume 2, pages 820–827. IEEE, 1999. (Cité en page 30.)
- [Patwardhan 2005] K.A. Patwardhan, G. Sapiro et M. Bertalmio. *Video inpainting of occluding and occluded objects*. In Image Processing, 2005. ICIP 2005. IEEE International Conference on, volume 2, pages II–69. IEEE, 2005. (Cité en pages 104, 111 et 112.)
- [Patwardhan 2007] K.A. Patwardhan, G. Sapiro et M. Bertalmío. *Video inpainting under constrained camera motion*. Image Processing, IEEE Transactions on, vol. 16, no. 2, pages 545–553, 2007. (Cité en page 104.)
- [Perona 1990] P. Perona et J. Malik. *Scale-space and edge detection using anisotropic diffusion*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 12, no. 7, pages 629–639, 1990. (Cité en page 17.)

- [Pizer 1987] S.M. Pizer, E.P. Amburn, J.D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J.B. Zimmerman et K. Zuiderveld. *Adaptive histogram equalization and its variations*. Computer vision, graphics, and image processing, vol. 39, no. 3, pages 355–368, 1987. (Cité en page 23.)
- [Pomerleau 1997] D. Pomerleau. *Visibility estimation from a moving vehicle using the RALPH vision system*. In Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on, pages 906–911. IEEE, 1997. (Cité en page 28.)
- [Richard 2001] M.M.O.B.B. Richard et M.K.Y.S. Chang. *Fast digital image inpainting*. In Appeared in the Proceedings of the International Conference on Visualization, Imaging and Image Processing (VIIP 2001), Marbella, Spain, 2001. (Cité en pages 106, 107, 108 et 112.)
- [Roser 2009] M. Roser et A. Geiger. *Video-based raindrop detection for improved image registration*. In Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on, pages 570–577. IEEE, 2009. (Cité en pages 56, 57 et 58.)
- [Rudin 1992] L.I. Rudin, S. Osher et E. Fatemi. *Nonlinear total variation based noise removal algorithms*. Physica D, vol. 60, pages 259–268, 1992. (Cité en page 19.)
- [Schechner 2001] Y.Y. Schechner, S.G. Narasimhan et S.K. Nayar. *Instant dehazing of images using polarization*. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, volume 1, pages I–325. IEEE, 2001. (Cité en pages 33 et 34.)
- [Sethian 1999] J.A. Sethian. *Fast marching methods*. SIAM review, vol. 41, no. 2, pages 199–235, 1999. (Cité en page 114.)
- [Snowden 1998] R.J. Snowden, N. Stimpson et R.A. Ruddle. *Speed perception fogs up as visibility drops*. Nature, vol. 392, page 450, 1998. (Cité en page 3.)
- [Soquet 2007] N. Soquet, D. Aubert et N. Hautiere. *Road segmentation supervised by an extended v-disparity algorithm for autonomous navigation*. In Intelligent Vehicles Symposium, 2007 IEEE, pages 160–165. IEEE, 2007. (Cité en page 149.)
- [Sumner 1977] R. Sumner, C. Baguley et J. Burton. *Driving in fog on the M4*. Rapport technique, 1977. (Cité en page 3.)
- [Sun 2005] J. Sun, L. Yuan, J. Jia et H.Y. Shum. *Image completion with structure propagation*. ACM Transactions on Graphics (ToG), vol. 24, no. 3, pages 861–868, 2005. (Cité en page 104.)
- [Tan 2007] R.T. Tan, N. Petterson et L. Petersson. *Visibility enhancement for roads with foggy or hazy scenes*. In Intelligent Vehicles Symposium, 2007 IEEE, pages 19–24. IEEE, 2007. (Cité en page 45.)
- [Tan 2008a] R.T. Tan. *Visibility in bad weather from a single image*. In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pages 1–8. IEEE, 2008. (Cité en pages 47 et 52.)

- [Tan 2008b] R.T. Tan, K. Ikeuchi et K. Nishino. *Color constancy through inverse-intensity chromaticity space*. Digitally Archiving Cultural Objects, pages 323–351, 2008. (Cité en page 46.)
- [Tang 2005] F. Tang, Y. Ying, J. Wang et Q. Peng. *A novel texture synthesis based algorithm for object removal in photographs*. Advances in Computer Science-ASIAN 2004. Higher-Level Decision Making, pages 3299–3300, 2005. (Cité en pages 104, 109 et 111.)
- [Tarel 2009] J.P. Tarel et N. Hautiere. *Fast visibility restoration from a single color or gray level image*. In Computer Vision, 2009 IEEE 12th International Conference on, pages 2201–2208. IEEE, 2009. (Cité en pages 52, 53 et 54.)
- [Tarel 2010] J.P. Tarel, N. Hautiere, A. Cord, D. Gruyer et H. Halmaoui. *Improved visibility of road scene images under heterogeneous fog*. In Intelligent Vehicles Symposium (IV), 2010 IEEE, pages 478–485. IEEE, 2010. (Cité en pages 63, 72 et 91.)
- [Tarel 2012] J.P. Tarel, N. Hautière, L. Caraffa, A. Cord, H. Halmaoui et D. Gruyer. *Vision Enhancement in Homogeneous and Heterogeneous Fog*. Intelligent Transportation Systems Magazine, IEEE, vol. 4, no. 2, pages 6–20, 2012. (Cité en pages 63, 65 et 72.)
- [Telea 2004] A. Telea. *An image inpainting technique based on the fast marching method*. Journal of graphics tools, vol. 9, no. 1, pages 23–34, 2004. (Cité en pages 112 et 113.)
- [Tenkink 1988] E. Tenkink. *lane keeping and speed choice with restricted sight distances—road user behavior. theory and research. papers presented at the 2nd international conference on road safety held in groningen, netherlands, august 1987*. Publication of : VAN GORCUM & COMP BV, 1988. (Cité en page 4.)
- [Tomasi 1998] C. Tomasi et R. Manduchi. *Bilateral filtering for gray and color images*. In Computer Vision, 1998. Sixth International Conference on, pages 839–846. IEEE, 1998. (Cité en page 15.)
- [Weickert 1998] J. Weickert. *Anisotropic diffusion in image processing*, volume 1. Teubner Stuttgart, 1998. (Cité en pages 18, 19 et 117.)
- [Weickert 2003] J. Weickert. *Coherence-enhancing shock filters*. Pattern Recognition, pages 1–8, 2003. (Cité en page 117.)
- [Whitaker 1993] R.T. Whitaker et S.M. Pizer. *A multi-scale approach to nonuniform diffusion*. CVGIP Image Understanding, vol. 57, pages 99–99, 1993. (Cité en page 18.)
- [Zhang 1990] KYJ Zhang et P. Main. *Histogram matching as a new density modification technique for phase refinement and extension of protein molecules*. Acta Crystallographica Section A : Foundations of Crystallography, vol. 46, no. 1, pages 41–46, 1990. (Cité en page 27.)

- [Zhang 1994] J. Zhang et H.H. Nagel. *Texture-based segmentation of road images*. In Intelligent Vehicles' 94 Symposium, Proceedings of the, pages 260–265. IEEE, 1994. (Cité en page [149](#).)
- [Zuiderveld 1994] K. Zuiderveld. *Contrast limited adaptive histogram equalization*. In Graphics gems IV, pages 474–485. Academic Press Professional, Inc., 1994. (Cité en page [23](#).)

Image Restoration During Foggy and Rainy Weather : Applications for Driver Assistance Systems

Abstract : Advanced Driver Assistance Systems (ADAS) are designed to assist the driver and in particular to improve road safety. For this purpose, various sensors are typically embedded in vehicles in order, for example, to alert the driver in case of imminent danger on the road. The use of camera type of sensor is a cost-effective solution and many ADAS based on camera are being created. Unfortunately, the performance of such systems decrease drastically in the presence of adverse weather conditions, especially in the presence of fog or rain, which could oblige to turn off the systems temporarily in order to avoid erroneous results. While, it is precisely in these difficult circumstances that the driver would potentially need the most to be assisted. Once the weather conditions detected and characterized by embedded vision, we propose in this thesis to restore the degraded image to provide a better signal to the ADAS and thus extend the operation range of these systems. In the state of the art, there are several approaches dealing with images restoration, some of which are dedicated to our fog and rain problem and others are more general : denoising, contrast or color enhancement, inpainting... We propose in this work to combine the two families of approaches. In the case of fog our contribution is to take advantage of both approaches (physical and signal) to propose a new automatic method adapted to the case of road images. We evaluated our method using ad hoc criteria (ROC curves, visible contrast to 5%, assessment on ADAS) applied to databases of synthetic and real images. In case of rain, once the drops present on the windshield are detected, we reconstruct the hidden parts of the image using a method of inpainting based on partial differential equations. The method parameters have been optimized on road images. Finally, we show that it is possible with this approach to build three types of applications : preprocessing, processing and assistance. In every family, we have proposed and evaluated a specific application : traffic signs detection during foggy weather ; detection of free space in fog conditions and display of the restored image to the driver.

Keywords : Signal and image processing, ADAS, Fog and Rain Detection, Image Restoration and Denoising, Deweathering, Inpainting, Traffic Signs Detection, Perception Time, Free Space Detection, Computer Vision, Onboard Camera (OBC).

Restauration d'images par temps de brouillard et de pluie : applications aux aides à la conduite

Résumé : Les systèmes d'aide à la conduite (ADAS) ont pour objectif d'assister le conducteur et en particulier d'améliorer la sécurité routière. Pour cela, différents capteurs sont généralement embarqués dans les véhicules afin, par exemple, d'avertir le conducteur en cas de danger présent sur la route. L'utilisation de capteurs de type caméra est une solution économiquement avantageuse et de nombreux ADAS à base de caméra voient le jour. Malheureusement, les performances de tels systèmes se dégradent en présence de conditions météorologiques défavorables, notamment en présence de brouillard ou de pluie, ce qui obligerait à les désactiver temporairement par crainte de résultats erronés. Hors, c'est précisément dans ces conditions difficiles que le conducteur aurait potentiellement le plus besoin d'être assisté. Une fois les conditions météorologiques détectées et caractérisées par vision embarquée, nous proposons dans cette thèse de restaurer l'image dégradée à la sortie du capteur afin de fournir aux ADAS un signal de meilleure qualité et donc d'étendre la gamme de fonctionnement de ces systèmes. Dans l'état de l'art, il existe plusieurs approches traitant la restauration d'images, parmi lesquelles certaines sont dédiées à nos problématiques de brouillard ou de pluie, et d'autres sont plus générales : débruitage, rehaussement du contraste ou de la couleur, "inpainting"... Nous proposons dans cette thèse de combiner les deux familles d'approches. Dans le cas du brouillard notre contribution est de tirer profit de deux types d'approches (physique et signal) afin de proposer une nouvelle méthode automatique et adaptée au cas d'images routières. Nous avons évalué notre méthode à l'aide de critères ad hoc (courbes ROC, contraste visibles à 5%, évaluation sur ADAS) appliqués sur des bases de données d'images de synthèse et réelles. Dans le cas de la pluie, une fois les gouttes présentes sur le pare-brise détectées, nous reconstituons les parties masquées de l'image à l'aide d'une méthode d'"inpainting" fondée sur les équations aux dérivées partielles. Les paramètres de la méthode ont été optimisés sur des images routières. Enfin, nous montrons qu'il est possible grâce à cette approche de construire trois types d'applications : prétraitement, traitement et assistance. Dans chaque famille, nous avons proposé et évalué une application spécifique : détection des panneaux dans le brouillard ; détection de l'espace navigable dans le brouillard ; affichage de l'image restaurée au conducteur.

Mots clés : traitement du signal et d'images, ADAS, détection du brouillard et de la pluie, restauration et débruitage d'images, "Deweathering", "Inpainting", détection de panneaux, temps de perception, détection de l'espace navigable, vision par ordinateur, caméra embarquée.
