



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA RECHERCHE  
SCIENTIFIQUE

**UNIVERSITE IBN KHALDOUN - TIARET**

# MEMOIRE

Présenté à :

FACULTÉ MATHÉMATIQUES ET INFORMATIQUE  
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

**MASTER**

Spécialité : [Génie informatique]

Par :

**BEKKAR Soumia**  
**BELMESSABIH Dalila**

Sur le thème

---

## **Déconvolution automatique des images floues**

---

Soutenu publiquement le .. / .. / 2019 à Tiaret devant le jury composé de :

Mr. CHENINE Aek

Grade Université MAA

Président

Mr. MEZZOUGE Karim

Grade Université MAA

Encadreur

Mr. BAGHDADI Mohamed

Grade Université MAB

Examineur

## **Remerciement :**

*Tout d'abord nous remercions **ALLAH** le Tout Puissant de nous avoir donné la force, la volonté, et le privilège d'étudier et de suivre le chemin de la science.*

*Mes sincères remerciements à mon encadreur **Mr. Mezzoug Karim** pour ces précieux conseils avec gentillesse et disponibilité pour son bon sens et son esprit vif pour la confiance qui a témoigné. Je remercie également les membres de jury et nous adressons également nos vifs remerciement à l'ensemble des enseignants de la faculté .A tout le personnel administratif.*

*Merci à tous*

*Dédicace*

***BEKKAR soumia***

*A mes très chers parents qui n'ont jamais  
cessé de m'encourager, que Dieu  
les protège*

*A mes très chers frères, et toute ma famille,*

*A tous mes amis*

*A tous ceux que j'aime, et tous ceux qui  
m'aiment,*

*Je dédie ce modeste travail*

*Dédicace*

***BELMESSABIH dalila***

*A mes très chers parents qui n'ont jamais  
cessé de m'encourager, que Dieu  
les protège*

*A mes très chers frères,  
A mes très chères sœurs, leur mari et leurs  
enfants.*

*A tous mes amis  
A tous ceux que j'aime, et tous ceux qui  
m'aiment,*

*Je dédie ce modeste travail*

## ***Summary :***

Images have an important place in the modern world. They are in the first place necessarily integrated into our organic life; blur images occupy a large center in image processing many researchers are finding solutions in this area to eliminate blur, improve images to see better image quality, and access image information.

Through the end of study project, we will present in the first part some methods of deconvolution in the form of a very practical utility to facilitate the task for researchers and future students in the field. In addition to a second part in our approach that was dedicated for a concrete application of this concept by using a toolbox for the segmentation of the blur compared to the image and then use this mask to correct and reverse the pace blurred through for example a photographic image wrongly taken.

This end of study project really represents a very modest platform for the processing of digital images in the field of deblurring while hoping that this application will be enriched by other treatment approaches.

## ***Résumé:***

Les images occupent une place importante dans le monde moderne. Ils sont en premier lieu nécessairement intégrés à notre vie organique ; Les images floues occupent une place centrale dans le traitement des images. De nombreux chercheurs ont trouvé des solutions dans ce domaine pour éliminer le flou, améliorer les images afin d'obtenir une meilleure qualité d'image et accéder aux informations relatives à l'image.

Dans le cadre du projet de fin d'étude, nous présenterons dans une première partie quelques méthodes de déconvolution sous la forme d'un utilitaire très pratique pour faciliter la tâche des chercheurs et des futurs étudiants du domaine. En plus d'une deuxième partie de notre approche qui était dédiée à une application concrète de ce concept en utilisant une boîte à outils pour la segmentation du flou par rapport à l'image, puis en utilisant ce masque pour corriger et inverser le flou.

Ce projet de fin d'étude représente en réalité une plate-forme très modeste pour le traitement des images numériques dans le domaine du deblurring, tout en espérant que cette application sera enrichie par d'autres approches de traitement.

## ملخص

الصور لها مكان مهم في العالم الحديث. إنها في المقام الأول مندمجة بالضرورة في حياتنا؛ تشغل الصور الضبابية مركزًا كبيرًا في معالجة الصور، ويعد العديد من الباحثين حلاً في هذا المجال للقضاء على الضبابية، وتحسين الصور لرؤية جودة صورة أفضل، والوصول إلى معلومات الصورة من خلال مشروع نهاية الدراسة، سنقدم في الجزء الأول بعض طرق فك الارتباط في شكل أداة عملية لتسهيل المهمة للباحثين وطلاب المستقبل في هذا المجال. بالإضافة إلى الجزء الثاني في منصفنا الذي تم تخصيصه لتطبيق ملموس لهذا المفهوم باستخدام مربع أدوات لتقسيم الترميز مقارنة بالصورة ثم استخدم هذا القناة لتصحيح وعكس وتيرة عدم وضوح صورة فوتوغرافية على سبيل المثال الصورة التي اتخذت خطأ يمثل مشروع نهاية الدراسة هذا حقًا منصة متواضعة جدًا لمعالجة الصور الرقمية في مجال إزالة الضبابية، معربًا عن أمله في إثراء هذا التطبيق من خلال أساليب علاجية أخرى

## Table des matières

Liste des figures

Liste des tableaux

Liste des équations

Introduction général ..... 1

### chapitre I :

Introduction ..... 3

2. Représentation des images ..... 3

A. Image binaire ..... 3

B. Image à niveaux de gris (intensité ou luminance) ..... 3

C. Image couleur ..... 4

3. système de traitement d'image ..... 4

a. L'acquisition ..... 4

- L'échantillonnage ..... 4

- La quantification ..... 4

b. Prétraitement ..... 4

a. Analyse d'image ..... 4

b. Interprétation ..... 4

4. Caractéristiques d'une image numérique ..... 5

. Dimension ..... 5

. Résolution ..... 5

. Bruit ..... 5

. Histogramme ..... 6

. Contours ..... 6

. Luminance ..... 6

. Contraste ..... 7

5. Techniques de traitement d'images ..... 7

A. Squelettisation ..... 7

B. Filtrage ..... 7

1. Filtres linéaires ..... 7

2. Filtres non linéaires ..... 10

C. Détection de contour ..... 10

D. Interpolation ..... 12

E. Objet de haut niveau (classification) :.....	12
F. Segmentation :.....	13
Conclusion :.....	14

## **Chapitre II :**

Introduction .....	16
1. Image floues.....	16
A. Raison de flou.....	16
B. Types de flou .....	16
1. Flou moyen .....	16
2. Flou gaussien .....	16
3. Flou de mouvement linéaire .....	17
4. Flou atmosphérique .....	17
5. Flou de focus : .....	17
2. Modèle de déconvolution.....	18
1. Techniques de déconvolution.....	19
A. Déconvolution non aveugle .....	19
Application aux cas des images .....	21
B. Déconvolution Aveugle .....	24
C. Comparaison des différences techniques de déconvolution :.....	26
2. Mesures de qualité d'image.....	26
Conclusion .....	27

## **chapitre III :**

Introduction .....	29
1. Travaux récents pour la déconvolution :.....	29
A. Méthode d'identification de paramètre proposée et algorithme de déconvolution d'image .....	29
a) Non-gaussianité et identification des paramètres :.....	29
b) Déconvolution d'images non aveugles .....	31
2. Travaux récents pour deblurring :.....	32
A. Évaluation des techniques de deblurring d'image .....	32
a. Techniques de deblurring : .....	32
b. Deblurring à l'aide d'une approche de réseau neural .....	34
3. Segmentation du flou de défocalisation basé sur LBP :.....	36
a. Métrique de flou proposée à base de LBP :.....	36

b.	Nouvel algorithme de segmentation de floue.....	38
1.	Génération de cartes de netteté à plusieurs échelles :.....	39
2.	Initialisation Alpha Matting : .....	39
3.	Calcul de la carte alpha :.....	40
4.	Inférence multi-échelle :.....	40
c.	Évaluation d'algorithme de segmentation de floue :.....	40
	Conclusion.....	42
<b>Chapitre IV :</b>		
	Partie I : conception et modélisation .....	44
1.	Définition d'UML .....	44
A.	Les modèles d'UML : .....	44
a.	Le diagramme de cas d'utilisation :.....	45
b.	Diagramme de séquence :.....	45
c.	Diagramme de classes : .....	45
B.	Entreprise Architect : .....	46
1.	Diagramme de cas d'utilisation :.....	46
2.	Diagramme de séquence :.....	48
3.	Diagramme de classes : .....	50
	Partie II : implémentation.....	51
1.	L'environnement de programmation MATLAB : .....	51
a.	Intérêts :.....	51
b.	Inconvénients : .....	52
2.	Analyse des besoins .....	52
3.	Notre approche.....	52
4.	Interface de l'application implémenté.....	53
5.	Résultat et discussion.....	56
	Conclusion.....	58
	Conclusion général .....	59
	Bibliographie.....	60

## Liste de figure

Figure 1 : image numérique I .....	3
Figure 2 : l'image original, l'image bruitée avec un bruit poivre et sel de 10% .....	5
Figure3 : l'image aérienne originale, l'image bruitée avec un bruit gaussien .....	6
Figure 4 : Exemple squelettisation .....	7
Figure 5 : Résultat du filtrage de l'image améliorée .....	9
Figure 6 : Principe du filtre médian .....	10
Figure 7 : Détection des contours d'une image par convolution de type Sobel .....	12
Figure 8 : Détection des contours d'une image par convolution de type Laplacien .....	12
Figure 9 : Segmentation d'une image .....	13
Figure10 : Segmentation, approche régions et l'approche contour .....	14
Figure 11 : image floue .....	16
Figure 12 : Modèle de dégradation de l'image .....	18
Figure13 : Processus montrant comment une image devient floue .....	18
Figure 14 : Classification des techniques d'image déconvolution .....	19
Figure 15 : déconvolution avec PSF connu .....	20
Figure 16 : déconvolution avec PSF inconnu .....	20
Figure 17 : schéma de déconvolution de Wiener .....	21
Figure 18 : image déconvolue avec filtre Wiener .....	22
Figure 19 : Image déconvolue avec le filtre régularisé .....	23
Figure 20 : Image de flou de mouvement à l'aide du filtre LR .....	24
Figure 21 : mouvement de masquage d'image à l'aide de Blind déconvoltion .....	25
Figure 22 : modèle de dégradation .....	29
Figure 23 : Modèles de flou et courbes non gaussiennes .....	31
Figure 24 : 8-bit LBP avec P=8,R=1 .....	37
Figure 25 : Distribution du code LBP dans les régions floues et pointues .....	37
Figure 26 : Histogramme de modèles de lombalgie .....	38

Figure 27 : Réponses métriques pour une image échantillon pour différentes métriques de netteté .....	39
Figure 28 : Mesures de précision, de rappel et F pour les seuils adaptatifs .....	41
Figure 29 : Comparaison de la carte de segmentation binaire avec Zhu et al .....	42
Figure 30 : organigramme des diagrammes de langage UML .....	45
Figure 31 : diagramme de cas d'utilisation .....	47
Figure 32 : diagramme de séquence .....	49
Figure 33 : diagramme de classe .....	50
Figure 34 : l'environnement Matlab .....	51
Figure 35 : présentation de l'approche de l'application .....	52
Figure 36 : interface menu principal de l'application .....	53
Figure 37 : fenêtre des filtres de déconvolution .....	54
Figure 38 : exemple de déconvolution avec le filtre winner .....	54
Figure 39 : le résultat de déconvolution avec filtre Lucy Richardson .....	55
Figure 40 : fenêtre de déconvolution avec filtre régularisé .....	55
Figure 41 : fenêtre de déconvolution avec filtre Blind .....	56
Figure 42 : fenêtre de focus .....	56
Figure 43 : fenêtre de résultat de traitement .....	57

## Liste des tableaux

Tableau 1 : Les différents types de masques de convolution utilisée pour la détection du contour d'une image.....	11
Tableau2 : tableau de comparaison des différentes méthodes de deblurring.....	26
Tableau 3 : comparaison de tous les algorithmes.....	34
Tableau 4 : Analyse de performance des méthodes de déblurring.....	36
Tableau 5 : Comparaison en temps d'exécution de différentes méthodes de segmentation du flou.....	41
Tableau 6 : administrateur.....	46
Tableau 7 : déconvolution d'image floue.....	48

*Introduction*

*Général*

### **Introduction général**

Il est évident que les images ont une place importante dans le monde moderne. Elles sont en premier lieu forcément intégrées à notre vie organique (la perception visuelle chez les êtres humains). Elles interviennent beaucoup dans notre vie quotidienne (téléphone, vidéo, jeux, TV, magazines ...) ; personnelle (images médicales, images biologiques, photographies, ...) ; professionnelle (bureautique, télésurveillance, visioconférence, vision industrielle, ...).

L'évolution continue dans le domaine informatique a énormément contribué au développement du domaine de traitement d'images, en bénéficiant de la grande augmentation de la puissance de calcul des nouveaux ordinateurs. Le traitement d'images qui est devenu une nécessité dans quasiment tous les aspects de la vie où on a recherché de manière automatique de l'information dans des images ; les images floues occupent un grand centre dans traitement d'image, plusieurs chercheurs trouvent les solutions dans ce domaine pour éliminer le flou, améliorer les images pour voir une meilleure qualité d'image et pour accéder aux informations de l'image.

Toutefois, l'exploitation des données dans les images floues nécessite une combinaison de prétraitement et de traitement d'informations brutes, afin d'exploiter les données et les objectifs.

Les problématiques abordées lors de ce mémoire sont les suivantes : déconvolution automatique d'une image floue, cette image contient le flou en avant ou arrière-plan.

A travers projet de fin d'étude nous allons présenté en première partie quelques méthodes de deconvolution sous forme d'un utilitaire très pratique afin de faciliter la tâche pour les chercheurs et futurs étudiants en la matière. En plus d'une deuxième partie dans notre approche qui a été dédié pour une application concrète de cette notion en utilisant un toolbox destiné à la segmentation du flou par rapport à l'image pour ensuite utiliser ce masque afin de corriger et inverser l'allure du flou à travers par exemple une image photographique mal prise.

Enfin ce mémoire se présente sous la décomposition suivante :

Le premier chapitre est consacré pour introduire au lecteur les notions de bases de l'image numérique.

Le deuxième chapitre est d'expliquer le flou et leur type ainsi que la déconvolution tout en citant les techniques les plus répandus dans le domaine en question.

Le troisième chapitre comprenant les travaux scientifiques les plus récents dans ce domaine.

Enfin on termine notre synthèse par un quatrième chapitre qui sera consacré à la description de notre application développée sous l'environnement MATLAB ainsi que sa modélisation et conception. En plus une partie sera destinée à quelques tests réalisés sur l'approche proposée.

# *Chapitre I*

## *L'image numérique*

## Introduction

L'image est partout. Elle est maintenant un des outils d'investigation les plus prisés de la recherche scientifique et technique. Son apport didactique, complémentaire au dialogue textuel et son caractère pluridisciplinaire ne sont en effet plus à démontrer.

Le traitement d'image est une discipline relativement jeune en pleine expansion donnant ainsi lieu chaque année à une multitude de travaux académiques, technologiques et industriels.

Par traitement d'image, on désigne l'ensemble des opérations appliquées sur les images numériques transformant une image en une autre, ou en une primitive formelle.

Nous allons exposer dans ce chapitre certains concepts de base élémentaires s'accordant au domaine de l'imagerie, images, et du traitement d'images.

### 1. Définition d'une image numérique :

Une image numérique est définie comme un signal fini bidimensionnel échantillonné à valeurs quantifiées dans un certain espace de couleurs. Elle est constituée de points (pixels). Autrement dit, une image est une matrice  $M \times N$  de valeurs entières prises sur un intervalle borné  $[0, N_g]$  où  $N_g$  est la valeur maximale du niveau de gris[1].

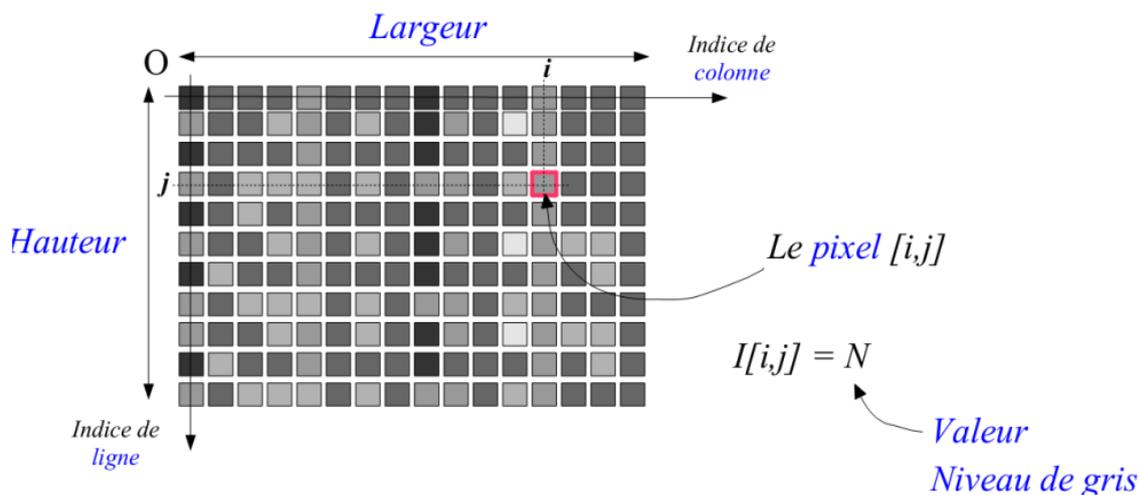


Figure 1 : image numérique I

## 2. Représentation des images

### A. Image binaire :

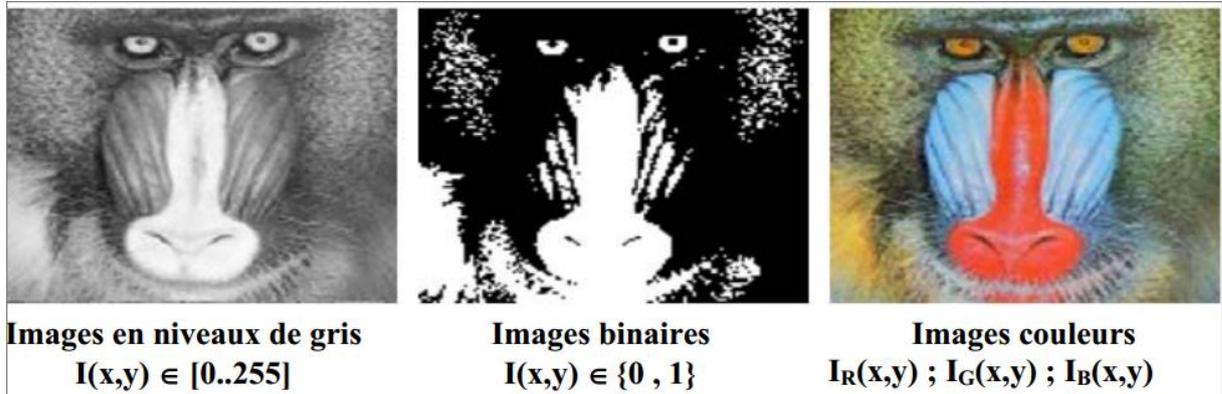
Dans une image binaire, les pixels sont représentés par deux états logiques 0 (noir) et 1 (blanc). C'est un codage de l'image sur 1 bit.

### B. Image à niveaux de gris (intensité ou luminance) :

Chaque pixel est codé sur  $N$  bits, ce qui lui confère des valeurs entières comprises entre 0 (noir) et  $2^N - 1$  (blanc). Et plus encore (image indexée, image 3D, image réelle,...).

**C. Image couleur :**

Une image couleur correspond à la synthèse additive de 3 images, rouge, vert et bleu. Chaque pixel est donc codé sur  $3 \times N$  bits.

**3. Système de traitement d'image :**

Un système de traitement d'image est une chaîne de quatre étapes : l'acquisition, le prétraitement, l'analyse d'image et l'interprétation.

**a. L'acquisition :**

L'acquisition est la première étape dans la chaîne du traitement d'images. Son rôle est l'acquisition et la numérisation de l'image l'en convertir d'une image analogique à une image numérique lisible et manipulable par l'ordinateur.

**- L'échantillonnage :**

L'échantillonnage consiste la discrétisation de coordonnées de l'image réelle. Son résultat est un plan à points discrets, ces points constituent l'ensemble des éléments composant l'image numérique (pixels).

**- La quantification :**

La quantification consiste la discrétisation d'intensités de l'image réelle. Son résultat est un intervalle de valeurs entières qui peut prendre un pixel. Ces valeurs définissent la nuance des niveaux de gris ou bien les niveaux de couleurs qui peuvent être prises par un pixel de l'image.

**b. Prétraitement :**

Le prétraitement est l'ensemble des opérations visant à améliorer les caractéristiques de l'image en vue d'un traitement postérieur.

**a. Analyse d'image :**

L'étape d'analyse d'image a pour but d'extraire des informations qualitatives ou quantitatives à partir d'une image ou d'un ensemble d'images.

**b. Interprétation :**

L'interprétation est l'étape finale du procédé de traitement d'image. Elle consiste à tirer des résultats à partir de l'étape précédente (analyse d'image). Une bonne interprétation requiert généralement des connaissances a priori sur le contenu de l'image, la structure des objets et la relation qui existe entre eux. [2]

#### 4. Caractéristiques d'une image numérique :

L'image est un ensemble structuré d'informations caractérisées par les paramètres suivants : [3]

- **Dimension**

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

- **Résolution**

C'est la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production d'images. Sur les moniteurs d'ordinateurs, la résolution est exprimée en nombre de pixels par unité de mesure (pouce ou centimètre).

- **Bruit**

Le bruit dans les images peut être appelé changement indésirable d'informations de couleur et de luminosité. Cela peut être visuel car les images contiennent des grains. Il existe différentes sources de bruit d'image. Il peut être généré au moment de prendre une image avec l'appareil photo ou pendant le transport. Les pixels d'une image bruyante apparaissent avec un taux d'intensité différent du taux d'intensité réel. Plusieurs mécanismes d'élimination du bruit utiles à la réduction de différents styles de bruit reposent sur l'exigence.

Les techniques couramment utilisées dans le traitement d'images pour le bruit sont :

- **Le bruit impulsif (Salt and Pepper)** : Le bruit « Poivre et sel » ou bruit impulsif est un bruit qui assigne à un certain nombre de pixels de l'image une valeur 0 ou 255 aléatoirement. Ce bruit est dû soit à des erreurs de transmission de données, soit à la défaillance d'éléments de capteur, soit à la présence de particules fines sur le capteur d'images.



a) Image original

b) bruit Salt and Pepper de 10%

Figure 2 : image originale, image bruitée avec un bruit poivre et sel de 10%

- **le bruit gaussien** : Le modèle standard de bruit d'amplificateur est additif, gaussien, indépendant de chaque pixel et indépendant de l'intensité du signal. Dans les caméras couleur où l'amplification est plus utilisée dans le canal de couleur bleu que dans le canal vert ou rouge, il peut y avoir plus de bruit dans le canal. canal bleu. Le bruit d'amplificateur est une partie importante du "bruit de lecture" d'un capteur d'image, c'est-à-dire du niveau de bruit constant dans les zones sombres de l'image.



Figure3 : A gauche l'image aérienne originale, à droite l'image bruitée avec un bruit gaussien.

- **Bruit Poisson** : Le bruit de Poisson ou bruit de tir est un type de bruit électronique qui se produit lorsque le nombre fini de particules transportant de l'énergie, telles que des électrons dans un circuit électronique ou des photons dans un dispositif optique, est suffisamment petit pour générer des fluctuations statistiques détectables dans une mesure.
- **Histogramme**  
L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris (couleur) dans l'image.
  - **Contours**  
Les contours représentent la frontière entre les objets de l'image, ou la limite entre deux pixels dont les niveaux de gris représentent une différence significative. Les textures décrivent la structure de ceux-ci. L'extraction de contour consiste à identifier dans l'image les points qui séparent deux textures différentes.
  - **Luminance**  
C'est le degré de luminosité des points de l'image.

- **Contraste**

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image.

## 5. Techniques de traitement d'images :

### A. Squelettisation [4]

La squelettisation donne une représentation compacte des objets, même dans le cas de formes et de topologies complexes. En dimension 2, les squelettes représentent des lignes interconnectées au centre d'un objet, comme le montre la figure (Figure 3).

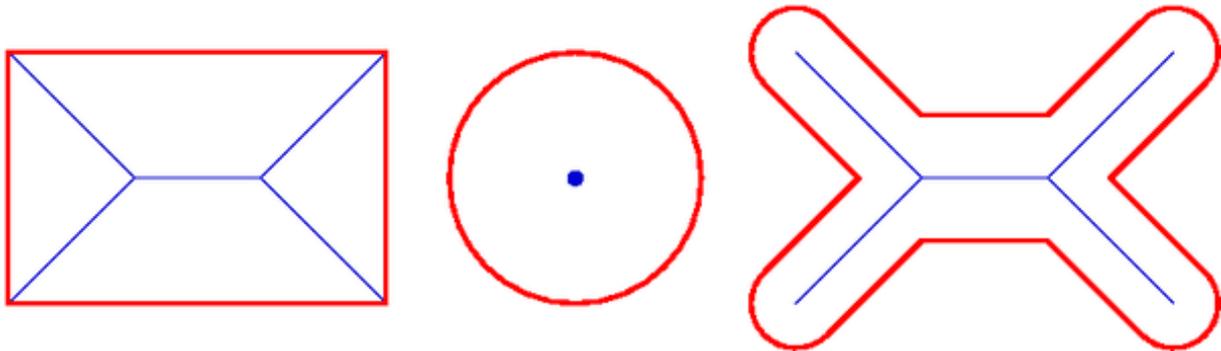


Figure 4 : Exemple squelettisation.

Ils sont très utilisés en analyse d'image et reconnaissance de forme, car ils permettent de décrire synthétiquement non seulement la forme, mais aussi certaines propriétés mathématiques des objets, comme par exemple la longueur ou la surface. Ils sont par ailleurs utilisés en compression d'image car ils décrivent les objets de façon compacte.

### B. Filtrage :

Le filtrage dans le traitement d'image est un processus qui permet de nettoyer les apparences et permet de mettre en surbrillance sélective des informations spécifiques. Un certain nombre de techniques sont disponibles et les meilleures options peuvent dépendre de l'image et comment elles seront utilisées. Le traitement d'image analogique et numérique peut nécessiter un filtrage pour produire un résultat final utilisable et attrayant. Cela peut être une partie courante du processus d'édition utilisé pour préparer les images à distribuer.

#### Pourquoi filtrer

Le filtrage d'une image est effectué pour assurer certaines fonctionnalités qui incluent :

- Eliminer ou réduire le bruit dans une image.
- Détecter les bords d'une image.
- Convolution entre une image et un filtre.

Les filtres peuvent être séparés en deux catégories, filtres linéaires et non linéaires.

#### 1. Filtres linéaires :

Un filtre linéaire transforme un ensemble de données d'entrée en un ensemble de données de sortie selon une opération mathématique appelée convolution. Il permet pour chaque pixel

de la zone à laquelle il s'applique, de modifier sa valeur en fonction des valeurs des pixels avoisinants.

Il existe plusieurs filtres linéaires, nous allons présenter quelques-uns :

**a. Les filtres passe-bas (lissage)**

Consistent à atténuer les pixels de l'image ayant une fréquence haute (pixels foncés). Ce type de filtrage est généralement utilisé pour atténuer le bruit de l'image, c'est la raison pour laquelle on parle habituellement de lissage. Les filtres moyens sont des types de filtres passe-bas dont le principe est de faire la moyenne des valeurs des pixels avoisinants. Le résultat de ce filtre est une image plus floue.

1	1	1
1	4	1
1	1	1

*Masque de convolution passe-bas.*

**b. Les filtres passe-haut (accentuation)**

À l'inverse des filtres passe-bas, les filtres passe-haut atténuent les pixels de basse fréquence de l'image et permettent notamment d'accentuer les détails et le contraste, c'est la raison pour laquelle le terme de "filtre d'accentuation" est parfois utilisé.

0	-1	0
-1	5	-1
0	-1	0

*Masque de convolution passe-haut.*

**c. Filtre Gaussien**

Le filtre Gaussien est un filtre isotrope spécial avec des propriétés mathématiques bien précises. La fonction gaussienne ( ) est souvent utilisée dans les distributions statistiques. Dans le traitement d'images, on traite des données à deux dimensions ( ), on introduit alors une fonction gaussienne à deux dimensions :

$$G(x, y) = \frac{1}{2\pi\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Etant données que les images numériques sont composées de valeurs discrètes, le noyau gaussien sera composé aussi de valeurs discrètes :

$$\frac{1}{6} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

Masque de convolution Gaussien

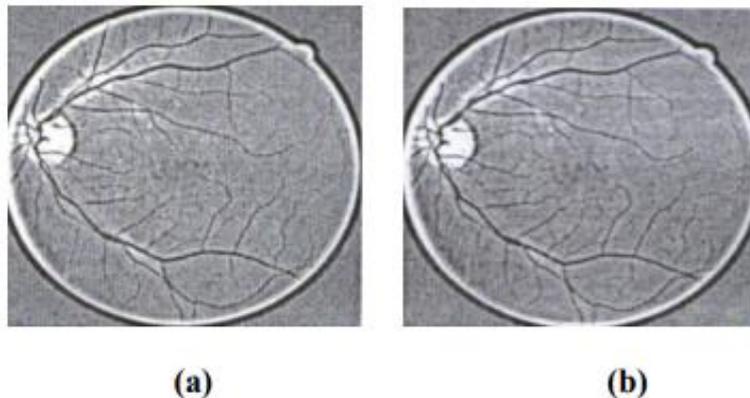


Figure 5 : Résultat du filtrage de l'image améliorée

(a) : image améliorée du canal vert ; (b) : le filtrage par la gaussienne de l'image (a).

D'après les résultats obtenus, on ne constate que le filtre de gauss permis d'atténuer le bruit mais ajouter un effet de flou.

**d. Filtre moyennneur**

Appelé également mean filtering, averaging ou Box filtering. Son principe est très simple : un pixel est remplacé par la moyenne de lui-même et de ses voisins. C'est dans la définition du voisinage que les filtres vont différer. Le filtre moyennneur est un filtre passe bas permettant ainsi d'éliminer les hautes fréquences, correspondantes au bruit. Son inconvénient est qu'il élimine également les hautes fréquences correspondantes aux détails de l'image : il rend ainsi l'image moins bruitée mais plus floue.

$$\frac{1}{9} * \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Masque de type 3\*3

## 2. Filtres non linéaires

Ils sont conçus pour régler les problèmes des filtre linéaires, surtout ce qui concerne la mauvaise conservation des contours. Leur principe est le même que les filtres linéaires, il s'agit toujours de remplacer la valeur de chaque pixel par la valeur d'une fonction calculée dans son voisinage. La différence est que cette fonction n'est plus linéaire mais une fonction quelconque. Les filtres non linéaires les plus connus sont :

### a. Le filtre médian

Le filtre médian appartient à la famille des filtres d'ordre. Les filtres d'ordre procèdent en remplaçant la valeur de chaque pixel par la valeur du pixel qui occupe un certain rang lorsqu'on trie les valeurs des pixels rencontrés dans un certain voisinage de ce pixel. La valeur du pixel central est remplacée par la valeur médiane de tous les pixels de la fenêtre d'analyse centrée sur ce pixel. Le filtrage médian est très robuste à différents types de bruit.

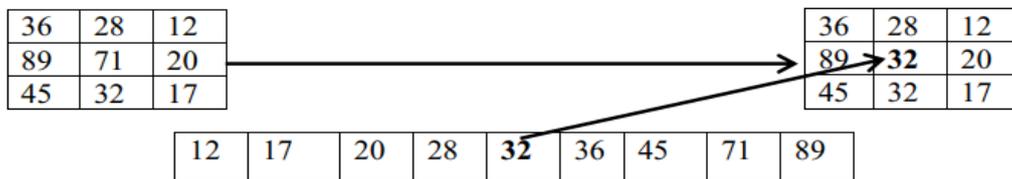


Figure 6 : Principe du filtre médian

Ce filtre - non linéaire - permet d'éliminer des pixels isolés ou les traînées fines noyées dans des zones assez homogènes

### b. Le filtre maximum

On applique le même traitement que celui du filtre médian mais la valeur du pixel du centre va être remplacée par le maximum.

### c. Le filtre minimum

On applique le même traitement que celui du filtre médian mais, cette fois, la valeur du pixel du centre va être remplacée par le minimum.

## C. Détection de contour

Le but de la détection de contours est de repérer les points d'une image numérique qui correspondent à un changement brutal de l'intensité lumineuse. La détection des contours d'une image réduit de manière significative la quantité de données et élimine les informations qu'on peut juger moins pertinentes, tout en préservant les propriétés structurelles importantes de l'image. Le tableau illustre les différents types de masques de convolution utilisables pour l'obtention du contour d'une image :

Type de masque	Gardien partiels	Extraction de l'amplitude	Extraction de la direction																								
<b>Masque de robert</b> <table border="1"> <tr><td colspan="2">Gardien selon x</td></tr> <tr><td>-1</td><td>0</td></tr> <tr><td>0</td><td>1</td></tr> </table> <table border="1"> <tr><td colspan="2">Gardien selon y</td></tr> <tr><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td></tr> </table>	Gardien selon x		-1	0	0	1	Gardien selon y		0	-1	1	0	$G_1 ; G_2$ Substitutions du pixel supérieur gauche	$A = \sqrt{G_1^2 + G_2^2}$	$\theta = \frac{\pi}{4} + \arctan\left(\frac{G_1}{G_2}\right)$												
Gardien selon x																											
-1	0																										
0	1																										
Gardien selon y																											
0	-1																										
1	0																										
<b>Masque de sobel</b> <table border="1"> <tr><td colspan="3">Gardien selon x</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> <tr><td>2</td><td>0</td><td>-2</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> </table> <table border="1"> <tr><td colspan="3">Gardien selon y</td></tr> <tr><td>1</td><td>2</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-2</td><td>-1</td></tr> </table>	Gardien selon x			1	0	-1	2	0	-2	1	0	-1	Gardien selon y			1	2	1	0	0	0	-1	-2	-1	$G_1 ; G_2$	$A = \sqrt{G_x^2 + G_y^2}$	$\theta = \arctan\left(\frac{G_y}{G_x}\right)$
Gardien selon x																											
1	0	-1																									
2	0	-2																									
1	0	-1																									
Gardien selon y																											
1	2	1																									
0	0	0																									
-1	-2	-1																									
<b>Masque de prewitt</b> <table border="1"> <tr><td colspan="3">Gardien selon x</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> </table> <table border="1"> <tr><td colspan="3">Gardien selon y</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-1</td><td>-1</td></tr> </table>	Gardien selon x			1	0	-1	1	0	-1	1	0	-1	Gardien selon y			1	1	1	0	0	0	-1	-1	-1	$G_1 ; G_2$	$A = \sqrt{G_x^2 + G_y^2}$	$\theta = \arctan\left(\frac{G_y}{G_x}\right)$
Gardien selon x																											
1	0	-1																									
1	0	-1																									
1	0	-1																									
Gardien selon y																											
1	1	1																									
0	0	0																									
-1	-1	-1																									
<b>Masque Laplacien</b> <table border="1"> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>-8</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	1	1	1	1	-8	1	1	1	1	$L$	$ L $																
1	1	1																									
1	-8	1																									
1	1	1																									

Tableau 1 : Les différents types de masques de convolution utilisée pour la détection du contour d'une image



Figure 7 : Détection des contours d'une image par convolution de type Sobel.

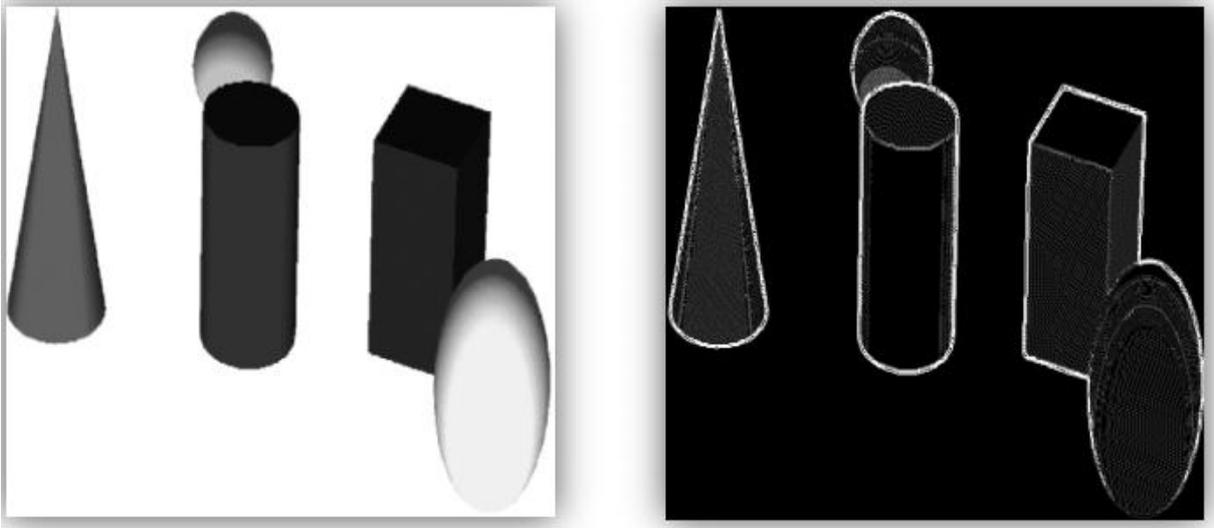


Figure 8 : Détection des contours d'une image par convolution de type Laplacien.

#### **D. Interpolation :**

Interpolation est (en traitement des images) le calcul d'un pixel en fonction de ses pixels voisins. Cette technique est utilisée lors d'une transformation d'image avec perte.

#### **E. Objet de haut niveau (classification) :**

Classification d'images est une tâche très courante. Pour pouvoir utiliser les images pour la cartographie ou pour des analyses complémentaires, il est souvent important de traduire l'information de fréquence contenue dans les images en information thématique portant sur l'occupation du sol ou la couverture végétale. Vous avez généralement le choix entre deux approches : la classification supervisée et non-supervisée.

**F. Segmentation :**

La segmentation est un traitement bas-niveau qui consiste à découper l'image en différentes régions et/ou contours. A pour but d'extraire les entités d'une image pour y appliquer un traitement spécifique et pour interpréter le contenu de l'image.

**- Objectifs de la segmentation**

L'extraction d'éléments et d'informations qualitatives de l'image.

- Fournir des régions homogènes
- Localiser les contours de région.
- Fournir une description de haut niveau : représentation par un graphe dont les nœuds sont des régions, chaque région est connectée à ses voisines et chaque région porte une étiquette qui donne des informations comme la taille, la couleur, la forme,...
- Réduire le bruit.

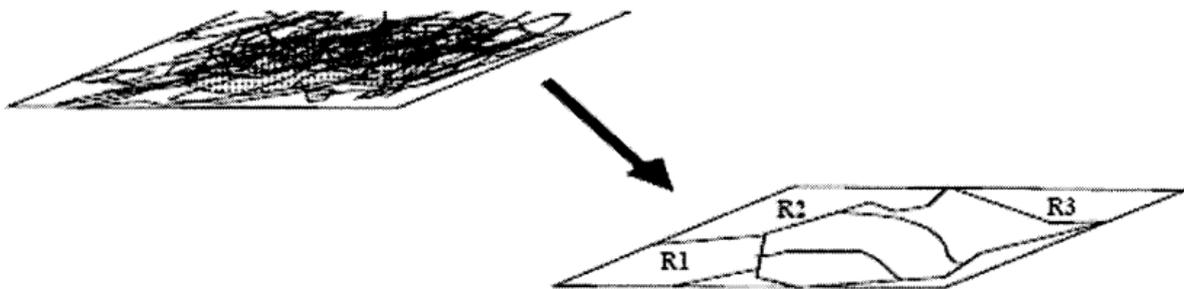


Figure 9 : Segmentation d'une image

**➤ La région :**

Une région est un ensemble de pixels connexes ayant des propriétés communes qui les différencient des pixels des régions voisines.

**➤ Le contour :**

Un contour est une variation brusque d'intensité ; on dit qu'un contour est équivalent à une région s'il est connexe et fermé.

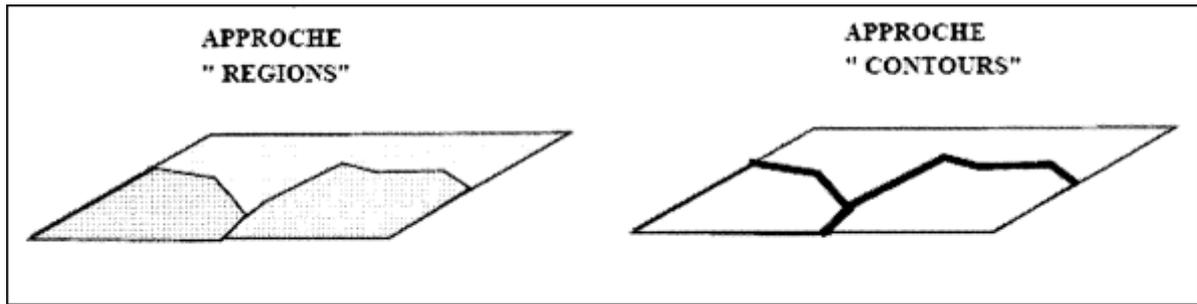


Figure10 : Segmentation, approche régions et l'approche contour

### Conclusion :

Dans ce chapitre nous avons présentés des outils de traitement d'image et des méthodes pour l'amélioration d'image rétinienne afin d'obtenir une image de meilleur qualité visuelle. Ces méthodes sont simples et efficaces pour analyser les images rétiniennes.

# *Chapitre II*

## *Déconvolution*

## Introduction

Les images prises par une caméra souffrent en général d'un certain niveau de représentation dégradée de l'objet d'origine au cours du processus de formation de l'image. Le flou est un artéfact indésirable commun associé à la formation d'images. Le flou est un processus déterministe, qui peut survenir pour diverses raisons, telles que le mouvement des distorsions atmosphériques, les aberrations optiques, le mouvement, etc.

Dans ce chapitre on présente le concept des images floues et les techniques de déconvolution.

### 1. Image floues

Le flou est une zone d'image floue provoquée par les mouvements de l'appareil photo ou du sujet, par une mise au point imprécise ou par l'utilisation d'une ouverture donnant une faible profondeur de champ. Les effets de flou sont des filtres qui adoucissent les transitions et réduisent le contraste en calculant la moyenne des pixels situés à côté des contours nets des lignes et des zones définies présentant une transition de couleur significative.



Figure 11 : image floue

#### A. Raison de flou

Les raisons de flou d'une image sont les suivantes :

- Un mouvement au moment de la procédure de capture d'image utilisant l'appareil photo où une durée de divulgation allongée est appliquée lors de la capture d'image.
- Objets flous, utilisant un objectif grand angle, des perturbations environnementales ou une durée d'exposition minime réduisant le nombre de photons capturés.
- Déformation optique étalée en microscopie confocale.

#### B. Types de flou

Pour mieux comprendre le flou de l'image, une connaissance plus approfondie des différentes raisons du flou est nécessaire.

1. **Flou moyen** : le flou moyen peut être dispersé dans les deux sens (horizontal, vertical). Le filtre moyen supprime ce type de flou et est utile lorsque le bruit affecte l'ensemble de l'image.
2. **Flou gaussien** : Flou simulé par la fonction gaussienne. Ce type de flou est impénétrable au centre et les flous sur les bords des flancs.

Flou gaussien en principe permet de calculer la moyenne pondérée des valeurs du voisinage. Les pixels du voisinage qui sont proches du pixel central et qui ont un poids plus fort.

Coefficients du noyau gaussien :  $A_{i,j} = \exp(-(i^2+j^2)/2\sigma^2)$ .

- 3. Flou de mouvement linéaire** : le flou de mouvement se produit en raison du mouvement comparatif entre la caméra et la scène. Les effets de flou de mouvement peuvent être simulés dans une image en utilisant un filtre de mouvement dans une direction spécifique, puis l'image résultante semblera bouger, et son modèle mathématique est le suivant :

$$h(x, y; M, \emptyset) = \begin{cases} \frac{1}{M}, & \tan \emptyset = -\frac{x}{y}, \frac{M}{2} \geq \sqrt{x^2 + y^2} \\ 0 & \text{sinon} \end{cases} \quad (1)$$

Où M et  $\emptyset$  représentent respectivement la longueur du mouvement et son orientation.

- 4. Flou atmosphérique** : Cela se produit généralement dans les domaines de la télédétection ou de l'exploration spatiale. En raison du mouvement irrégulier des molécules atmosphériques, qui ont une pression, une vitesse et une chaleur inégales dans différentes positions de l'air, il en résulte une mauvaise influence sur les caractéristiques de propagation de la lumière. En général, il est toujours modélisé par un filtre passe-bas gaussien et est également appelé flou gaussien.

$$h(x, y; \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad (2)$$

Où  $\sigma$  indique le degré de flou. Il convient de noter que depuis (2) est un filtre à réponse impulsionnelle infinie (IIR), une troncature et une normalisation appropriées sont nécessaires dans les applications pratiques.

- 5. Flou de focus** :

C'est l'ubiquité lorsqu'un appareil photo prend des scènes naturelles en images 2D, L'ouverture de l'appareil photo étant circulaire, un filtre passe-bas idéal peut le modéliser

$$h(x, y; R) = \begin{cases} \frac{1}{\pi R^2}, & x^2 + y^2 \leq R^2 \\ 0 & \text{sinon} \end{cases} \quad (3)$$

Où R est le rayon qui indique l'étendue du flou.

## 2. Modèle de déconvolution

Image floue basée sur le modèle de dégradation. Selon le modèle de dégradation représenté par la figure 1, l'image d'origine convolera avec une fonction dégradée, c'est-à-dire une fonction d'étalement de point utilisant un opérateur de convolution qui fonctionne comme un opérateur de multiplication. Ensuite, nous obtenons une image dégradée ou une image floue. Le bruit est également présent dans l'image dégradée.

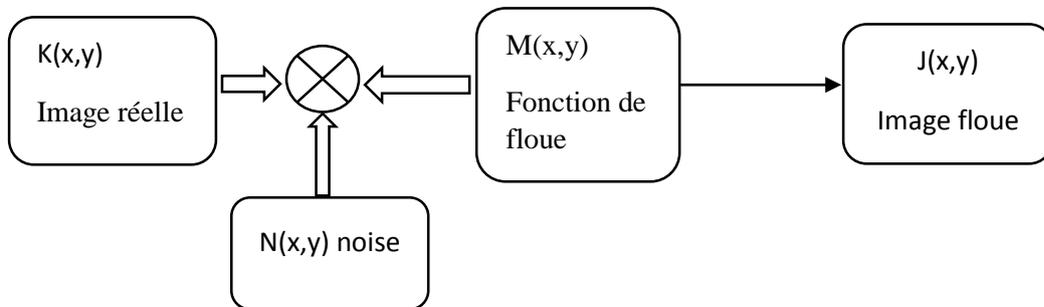


Figure 12 : Modèle de dégradation de l'image

L'image floue peut être représentée par l'équation suivante basée sur le modèle de dégradation :

$$J(x, y) = M(x, y) * K(x, y) + N(x, y) \quad (4)$$

Telle que,  $j(x, y)$  est une image dégradée,  $k(x, y)$  représente l'image d'origine non corrigée,  $m(x, y)$  représente le noyau flou qui a provoqué la dégradation,  $n(x, y)$  représente le bruit.

Représentation d'image de l'équation montrant comment une image devient floue :

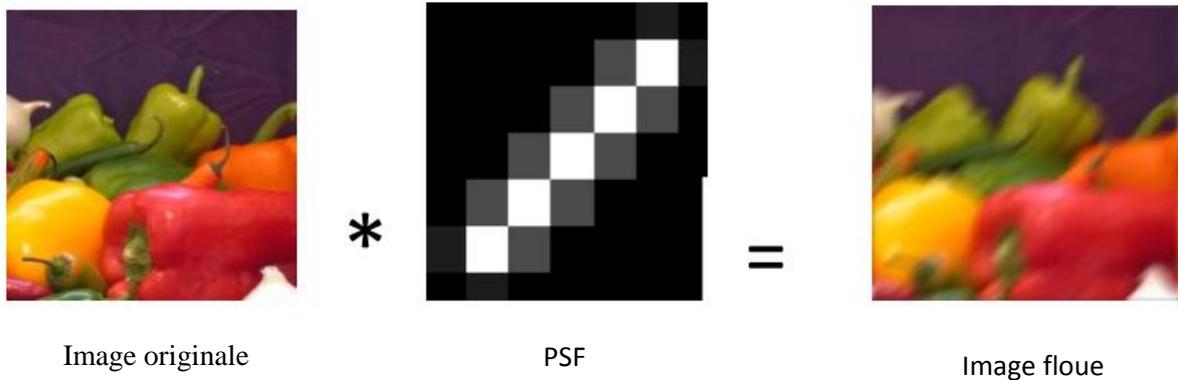


Figure13 : Processus montrant comment une image devient floue

Avec :

\*: convolution

PSF : La fonction d'étalement du point (point spread function)

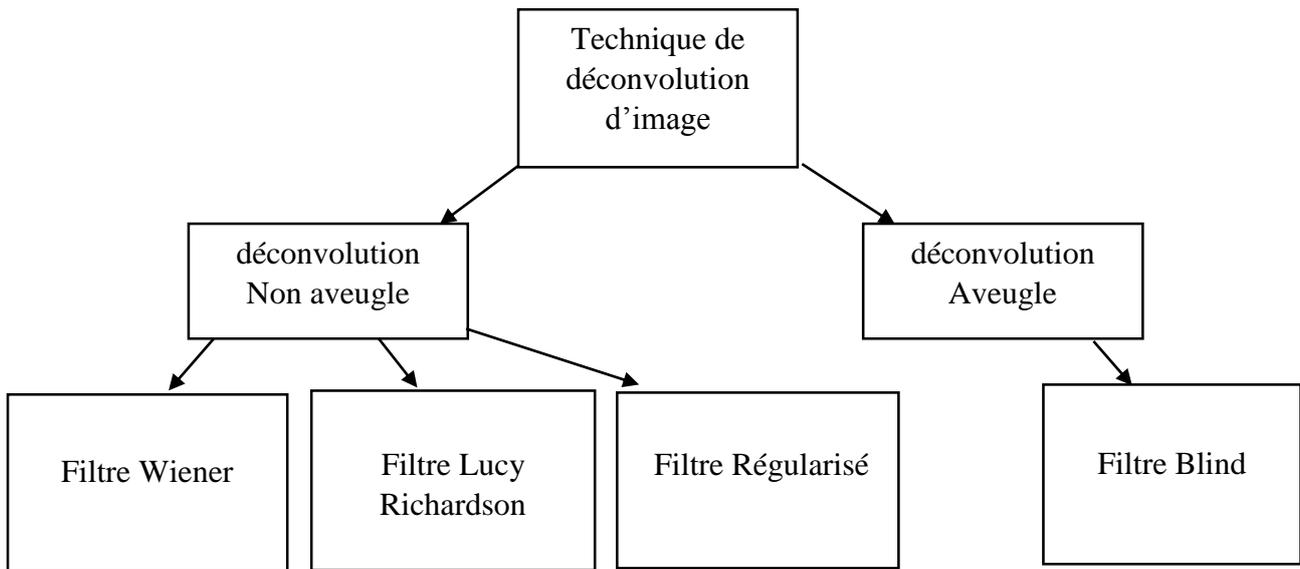


Figure 14 : Classification des techniques d'image déconvolution

## 1. Techniques de déconvolution

Il existe différentes techniques de suppression d'images dans le traitement des images. Parmi lesquelles quatre techniques de base sont classées comme :

### A. Déconvolution non aveugle

Dans cette technique, nous avons besoin des connaissances préalables sur les paramètres du noyau flou (PSF) pour effectuer la déconvolution.

- **Déconvolution : PSF Connu**

Les évaluations de performances de l'opération de suppression de flou avec une PSF connue peuvent être mises en œuvre dans deux cas: le premier cas a une quantité connue de flou, mais aucun bruit n'a été ajouté à une image, et le second cas est une quantité connue de flou et de bruit qui s'ajoute à l'image, puis l'image a été filtrée pour éliminer cette quantité connue de flou et de bruit à l'aide des méthodes de brouillage de Wiener, régularisées et Lucy Richardson. Dans le premier cas, les techniques régularisées et de Wiener ont donné les meilleurs résultats, mais il était surprenant que la technique de Lucy Richardson ait produit les plus mauvais résultats dans ce cas.

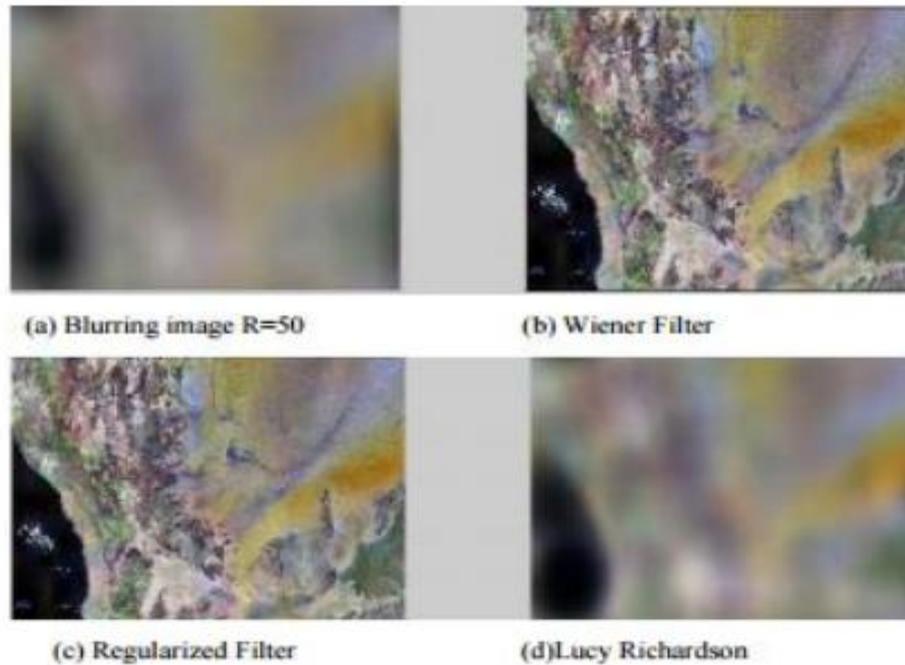


Figure 15 : déconvolution avec PSF connu.

- **Déconvolution PSF inconnu :**

Dans ce cas, appliqué avec une autre technique, on appelle technique de déconvolution en aveugle, qui est exécutée après l'estimation de la PSF, le nombre d'itérations et le seuil de poids de celle-ci. Après de nombreuses expériences, il s'est avéré que le seuil de poids devait être défini entre 0,10 et 0,25, la taille de la matrice PSF devait être définie sur 13x13 et le nombre d'itérations devait être un nombre supérieur à 30.

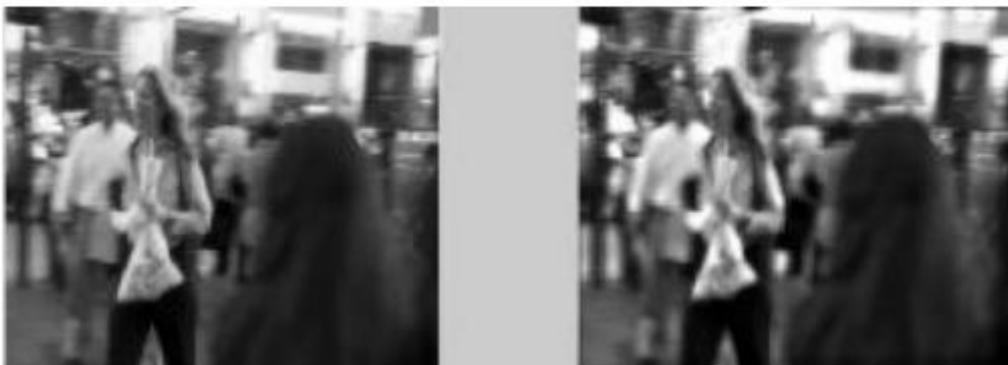


Figure 16 : déconvolution avec PSF inconnu.

- **Déconvolution utilisant le filtre de Wiener :**

Est une opération mathématique appliquant un filtre de Wiener pour éliminer ou atténuer une partie des bruits dans un signal. Elle opère dans le domaine fréquentiel en essayant de minimiser l'impact du bruit là où le rapport signal/bruit est mauvais. Cette méthode convient non seulement au son, mais aux images, car le spectre de fréquence de la plupart des images

visuelles est souvent bien conditionné et peut être estimé facilement. Elle tient son nom du mathématicien Norbert Wiener. le filtrage de WIENER ne permet que de restaurer des images qui ont été dégradées par l'ajout d'un bruit aléatoire. Nous allons voir qu'il est possible de combiner le filtrage de WIENER avec un procédé de déconvolution. Considérons le schéma de la figure 17. Comme pour le filtrage de WIENER, nous allons réaliser le raisonnement pour le cas à une dimension. La fonction idéale est ici dégradée par convolution avec un filtre dont la fonction de transfert est  $K(f)$  et par l'ajout d'un bruit  $n(t)$ . La fonction dégradée est toujours notée  $w(t)$ .

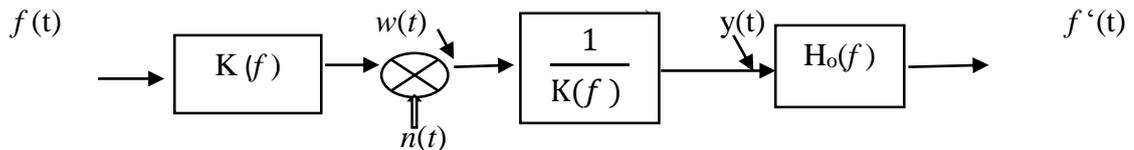


Figure 17 : schéma de déconvolution de Wiener

Avant d'appliquer la technique du filtrage de WIENER, il est nécessaire de réaliser la déconvolution par un filtre dont la fonction de transfert est égal à l'inverse de la fonction de transfert  $K(f)$ . Le filtre global de restauration est obtenu par combinaison du filtre de déconvolution et du filtre de WIENER. Vu que le filtre de déconvolution est connu, il reste simplement à déterminer la fonction de transfert  $H_o(f)$  du filtre de WIENER avant de combiner les deux filtres. Calculons-la transformée de FOURIER du signal dégrad

$$W(f) = K(f)F(f) + N(f) \quad (5)$$

Ainsi que celle du signal  $y(t)$  après déconvolution

$$y(t) = F(f) + \frac{N(f)}{K(f)} = F(f) + N'(f) \quad (6)$$

Tout se passe donc comme si  $y(t)$  était le résultat d'une dégradation de la fonction  $f(t)$  par un bruit aléatoire  $n'(t)$ . La détermination du filtre de WIENER devient alors évidente :

$$H_o(f) = \frac{\gamma f(f)}{\gamma f(f) + \gamma n'(f)} \quad (7)$$

Dans le cas de signaux non-corrélés. Finalement, la fonction de transfert du filtre global de restauration (ou de déconvolution de WIENER) est donné par :

$$g(f) = \frac{H_o(f)}{K(f)} \quad (8)$$

### Application aux cas des images

La déconvolution de WIENER peut aisément être adaptée aux cas des images à deux dimensions. Si l'on note  $y_f(u,v)$  la densité spectrale de puissance de l'image  $f(x, y)$ ,  $y_n(u,v)$  celle de bruit aléatoire additionnel et  $K(u,v)$  la fonction de transfert de filtre ayant dégradé l'image originale, la déconvolution WIENNER se caractérise par le filtre de déconvolution suivant :



Figure 18 : image déconvolue avec filtre Wiener.

- **Déconvolution à l'aide d'un filtre régularisé**

La restauration régularisée fournit des résultats similaires au filtrage de Wiener mais justifiés par un point de vue très différent. En outre, moins d'informations préalables sont nécessaires pour appliquer la restauration régularisée. (Remarque : la fonction Image Processing Toolbox qui implémente la restauration régularisée est appelée `deconvreg`).

La régularisation met en jeu deux objectifs souhaitables :

- 1) la proximité de l'ajustement du modèle.
- 2) la proximité du comportement du modèle par rapport à quelque chose qui serait attendu en l'absence d'une connaissance spécifique des paramètres ou des données du modèle.

Pour obtenir cette régularisation avec préservation des contours, ils ont utilisé une fonction  $\varphi$  non quadratique, appelée fonction de potentiel. Les propriétés des  $\varphi$  fonctions ont été étudiées dans un cadre variationnel pour préserver les discontinuités tout en interdisant l'amplification du bruit. Ces fonctions sont symétriques, positives, croissantes sur  $\mathbb{R}^+$ , elles sont quadratiques près de l'origine et ont une forme linéaire ou sous-linéaire en  $\pm\infty$  pour préserver les contours.[5]

La solution régularisée est calculée en minimisant l'énergie suivante :

$$U(\mathbf{X}) = \|\mathbf{Y} - \mathbf{X}\|^2 / 2\sigma^2 + \lambda\phi(\mathbf{X}) \quad (9)$$

$$\text{Ou } \phi(\mathbf{X}) = \sum_{ij} \left\{ \varphi \left( \frac{(\mathbf{D}_z \mathbf{X})_{ij}}{\delta} \right) + \varphi \left( \frac{(\mathbf{D}_y \mathbf{X})_{ij}}{\delta} \right) \right\} \quad (10)$$

où  $\mathbf{D}_x$  et  $\mathbf{D}_y$  sont les opérateurs de dérivation (différences de pixels voisins) par rapport aux colonnes et aux lignes. Les réels  $\lambda$  et  $\delta$  sont les paramètres du modèle de régularisation. Cela correspond à l'estimateur du Maximum A Posteriori (MAP). La contrainte de régularisation peut être exprimée comme la distribution a priori de l'image inconnu.

Le terme  $\|\mathbf{Y} - \mathbf{X}\|^2 / 2\sigma^2$  correspond à la vraisemblance de  $\mathbf{X}$  par rapport aux statistiques du bruit gaussien stationnaire :

$$P(\mathbf{Y}|\mathbf{X}) = \mathbf{K}^{-1} e^{-\|\mathbf{Y} - \mathbf{H}\mathbf{X}\|^2 / 2\sigma^2} \quad (11)$$

L'énergie  $U(\mathbf{X})$  est minimisée dans un cadre variationnel, afin de conserver un algorithme rapide. Cela est équivalent à maximiser la probabilité a posteriori de l'image  $\mathbf{X}$ , dans une approche stochastique [6]. La distribution a posteriori est donnée par l'équation suivante en utilisant la règle de Bayes

$$P(X|Y,\lambda) = Z^{-1}_{Y,\lambda} e^{-\|Y-HX\|^2/2\sigma^2 - \lambda\phi(X)} \quad (12)$$



Figure 19 : Image déconvolue avec le filtre régularisé.

- **Déconvolution à l'aide de Lucy-Richardson**

L' algorithme de Richardson – Lucy , également connu sous le nom de déconvolution de Lucy – Richardson , est une procédure itérative permettant de récupérer une image sous-jacente qui a été floue par une fonction de propagation de point connue. Il a été nommé d'après William Richardson et Leon Lucy, qui l'ont décrit de manière indépendante. [7] [8]

Lorsqu'une image est produite à l'aide d'un système optique et détectée à l'aide d' un film photographique ou d'un dispositif à couplage de charge, par exemple, elle est inévitablement floue, une source ponctuelle idéale n'apparaissant pas comme un point mais étant étalée dans la fonction dite de propagation par points. . Les sources étendues peuvent être décomposées en la somme de nombreuses sources ponctuelles individuelles, ainsi l'image observée peut être représentée sous la forme d'une matrice de transition  $p$  opérant sur une image sous-jacente :

$$d_i = \sum_j P_{i,j} U_j \quad (13)$$

Où  $U_j$  est l'intensité de l'image sous-jacente au pixel  $j$  et  $d_i$  est l'intensité détectée au pixel  $i$  . En général, une matrice, dont les éléments sont  $P_{i,j}$  , décrit la partie de la lumière provenant du pixel source  $j$  qui est détectée dans le pixel  $i$ . Dans la plupart des bons systèmes optiques (ou en général, des systèmes linéaires décrits comme invariants au décalage), la fonction de transfert  $P$  peut être exprimée simplement en termes de décalage spatial entre le pixel source  $j$  et le pixel d'observation  $i$  :

$$P_{i,j} = P(i-j)$$

Où  $P(\Delta i)$  est appelée une fonction d'étalement de points . Dans ce cas, l'équation ci-dessus .devient une convolution . Ceci a été écrit pour une dimension spatiale, mais bien sûr, la plupart des systèmes d'imagerie sont bidimensionnels, la source, l'image détectée et la fonction d'étalement du point ayant toutes deux indices. Ainsi, une image détectée en deux dimensions est une convolution de l'image sous-jacente avec une fonction d'étalement de points en deux dimensions  $P(\Delta x, \Delta y)$  plus un bruit de détection ajouté.

Afin d'estimer  $U_j$  compte tenu de l'observé  $d_i$  et un  $P$  connu ( $\Delta i_x, \Delta j_y$ ), nous utilisons la procédure itérative suivante dans laquelle l'estimation de  $U_j$  que nous appelons  $U_j^{(t)}$  pour le numéro d'itération  $t$  est mis à jour comme suit :

$$U^{(t+1)} = U^{(t)} \cdot \left( \frac{d}{U^{(t)} \otimes P} \otimes P^* \right) \quad (14)$$

Où la division et la multiplication sont des éléments, et  $P^*$  est la fonction de propagation du point de retournement.

l'algorithme de Richardson – Lucy a été proposée afin de réaliser la déconvolution à l'aveugle . [9]

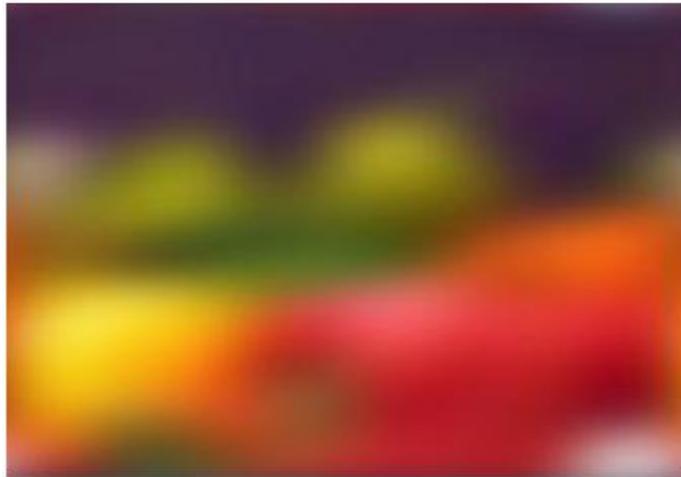


Figure 20 : Image de flou de mouvement à l'aide du filtre LR.

### B. Déconvolution Aveugle

La déconvolution à l'aveugle est une technique de déconvolution qui permet de récupérer la scène cible à partir d'un seul ou d'un ensemble des images floues en présence d'une fonction d'étalement du point (PSF) mal déterminée ou inconnue. [10] Les techniques classiques de déconvolution linéaire et non linéaire utilisent un PSF connu. Pour la déconvolution en aveugle, la PSF est estimée à partir de l'image ou du jeu d'images, permettant ainsi à la déconvolution d'être effectuée. Les chercheurs étudient les méthodes de déconvolution à l'aveugle depuis plusieurs décennies et ont abordé le problème sous différents angles.

La plupart des travaux sur la déconvolution à l'aveugle ont commencé au début des années 1970. La déconvolution aveugle est utilisée en imagerie astronomique et en imagerie médicale. La déconvolution en aveugle peut être effectuée de manière itérative, chaque itération améliorant l'estimation de la scène, ou de manière non itérative, lorsqu'une application de l'algorithme, basée sur des informations extérieures, extrait la PSF. Les méthodes itératives incluent l'estimation maximum a posteriori et les algorithmes de maximisation des attentes. Une bonne estimation de la PSF est utile pour une convergence plus rapide mais pas nécessaire.

SeDDaRA, [11] la transformation du cepstre et APEX sont des exemples de techniques non itératives. La transformée du cepstre et les méthodes APEX supposent que le fichier PSF a une forme spécifique et il est nécessaire d'estimer la largeur de la forme. Pour SeDDaRA, les informations sur la scène sont fournies sous la forme d'une image de référence. L'algorithme

estime la PSF en comparant les informations de fréquence spatiale dans l'image floue à celles de l'image cible.

La limitation de la déconvolution en aveugle est que l'image en entrée et le noyau flou doivent vivre dans un sous-espace fixe. Cela signifie que l'image d'entrée, représentée par  $w$ , doit être écrite sous la forme  $w = B \cdot h$ , où  $B$  est une matrice aléatoire de taille  $L$  sur  $K$  ( $K < L$ ) et  $h$  est de taille  $K$  sur  $1$ , tandis que le noyau flou, s'il est représenté par  $x$ , doit être écrit  $x = C \cdot m$ , où  $C$  est une matrice aléatoire de taille  $L$  sur  $N$  ( $N < L$ ) et  $m$  de taille  $N$  sur  $1$ . Image observée, si représentée par  $y$ , donnée par  $y = w * x$ , ne peut être reconstruite que si  $L \geq K + N$ .



Figure 21 : Mouvement de masquage d'image à l'aide de Blind Déconvolution

La méthode de déconvolution aveugle est principalement de deux types :

La première est basée sur la projection et le second est la méthode du maximum de vraisemblance.

L'approche basée sur la projection restaurera simultanément le fichier PSF résultant et l'image réelle. Cette technique est de nature cylindrique et se répète continuellement jusqu'à ce qu'elle réponde à un critère prédéfini. Ils ont d'abord essayé d'estimer le PSF, puis l'image réelle. L'un des avantages de cette méthode est qu'elle peut facilement gérer les imprécisions de la taille du support et qu'elle n'est pas sensible au bruit.

La seconde approche, comme son nom l'indique, permet de déterminer l'estimation du maximum de vraisemblance de paramètres de flou tels que PSF et les matrices de covariance. Étant donné que le PSF estimé n'est pas unique, nous pouvons prendre en considération d'autres facteurs, tels que la taille, la symétrie, etc. du PSF estimé. L'un des principaux avantages de cette méthode est que son calcul et sa complexité sont faibles et permettent également de déterminer les spectres de flou, de bruit et de puissance de l'image réelle.

## C. Comparaison des différences techniques de déconvolution :

Méthode	Types de flou	Performance
déconvolution aveugle	flou gaussien	Efficace et bon
lucy Richardson	flou gaussien	Efficace
Filtre Wiener	flou gaussien	pire performance
Filtre régularisé	flou gaussien	efficace

Tableau2 : tableau de comparaison des différentes méthodes de deblurring

## 2. Mesures de qualité d'image

La qualité de l'image est estimée à l'aide d'une formulation mathématique visant à attribuer la valeur de distorsion de l'image.

- 1. Erreur quadratique moyenne (MSE) :** Le plus simple paramètre de la mesure de qualité d'image est le MSE (L'erreur quadratique moyenne). Il est défini comme suit :

$$MSE = \frac{1}{M*N} \sum_{(i,j)=1}^{M,N} [g(i,j) - f(i,j)]^2 \quad (15)$$

Où g et f représentent respectivement, l'image originale et l'image filtrée, la grande valeur de MSE signifie que l'image est de mauvaise qualité.

- 2. RMSE (The root MSE) :** RMSE est la racine carrée de l'erreur quadratique moyenne MSE sur une fenêtre M \* N.

$$RMSE = \sqrt{\frac{1}{M*N} \sum_{(i,j)=1}^{M,N} [g(i,j) - f(i,j)]^2} \quad (16)$$

- 3. Rapport signal sur bruit crête (PSNR) :** PSNR (Peak Signal to Noise Ratio) est l'une des métriques les plus populaires et les plus utilisées à ce jour. Cela est dû essentiellement à sa simplicité, il est utilisé comme critère d'évaluation de la qualité de débruitage en plus de l'évaluation subjective. Ce dernier mesure la distorsion entre l'image originale g et l'image débruitée f suivant l'équation :

$$PSNR = -10 \log_{10} \frac{MSE}{g_{max}^2} \quad (17)$$

Où  $g_{max} = 255$ . PSNR maximale devrait indiquer que le filtrage est de meilleure qualité.

**Conclusion :**

Dans ce chapitre, nous avons présenté les images floues et les différents types de floue et on a cité les différentes techniques de déconvolution. Le chapitre suivant fera l'objectif d'une discussion de quelques travaux parmi les plus récents dans le domaine en question sur le déconvolution automatique des images floues.

# *Chapitre III*

## *L'état de l'art*

## Introduction

Le traitement des images numérique trouve des applications dans un nombre croissant de domaine aussi divers que le tri automatique du courrier. La croissance exponentielle des capacités des processeurs est pour une grande part dans le développement du traitement d'images, ce domaine étant, par nature, très gourmand en puissance de calcul. Les informations que l'on souhaite extraire d'une image, les applications, les types de données, sont si variés qu'il est difficile, lorsque l'on doit traiter une nouvelle application, de savoir à quel modèle qu'on doit appliquer et quelle méthodologie suivre. A travers ce chapitre nous avons choisi et sélectionné les travaux scientifiques parmi les plus récents pour donner aux lecteurs une vue générale sur l'actualité du domaine en question à savoir la déconvolution des images numériques

### 1. Travaux récents pour la déconvolution :

Les images numériques souffrent toujours de différents types de dégradations dans le système d'imagerie et le flou de l'image en est un phénomène courant, qui réduit considérablement la qualité des images. Il est donc urgent de récupérer une version approximative de l'image d'origine à partir d'une observation.

#### A. Méthode d'identification de paramètre proposée et algorithme de déconvolution d'image

##### a) Non-gaussianité et identification des paramètres :

Selon le théorème central limite, la sommation de variables aléatoires ayant une distribution identique tend toujours vers une distribution gaussienne dans la plupart des situations. Au vu du modèle de dégradation d'image (22), la procédure de flou peut être considérée comme une combinaison linéaire des décalages d'une image avec des poids différents de ceux définis par le noyau de flou. Par conséquent, une image d'observation floue serait plus gaussienne que l'original, et il est possible de restaurer des images latentes sur la base d'une mesure de non-gaussianité. La négentropie est l'une des méthodes classiques de mesure de la non-gaussianité, qui est robuste aux valeurs éloignées. Cependant, il est complexe dans le calcul et nécessite une approximation dans la pratique.

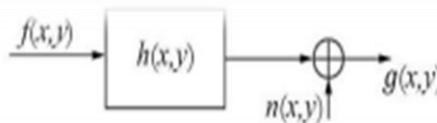


Figure 22 : model de dégradation

Le kurtosis est une autre mesure possible, qui est liée au quatrième moment de la donnée d'entrée  $x$  lorsqu'il est prétraité d'avoir une variance unitaire. Kurtosis est très simple en calcul et permet de mesurer efficacement l'épaisseur d'une distribution.

$$\text{Kurt}(x) = E\{x^4\} - 3(E\{x^2\})^2 \quad (18)$$

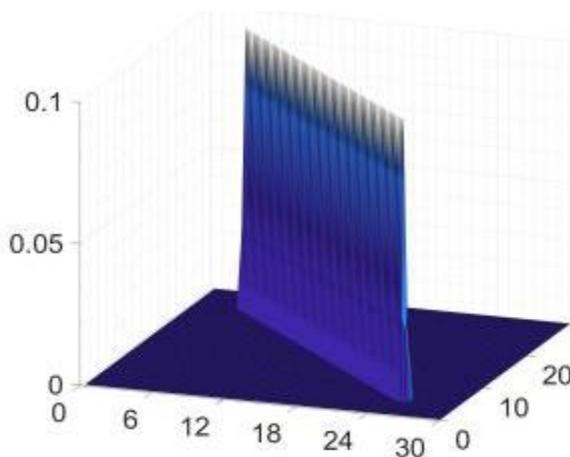
L'équation ci-dessus montre que le kurtosis est 0 lorsque la variable aléatoire  $x$  soumet une distribution gaussienne et qu'elle est supérieure à 0 pour les distributions super-gaussiennes et

inférieure à 0 pour les distributions sous-gaussiennes. Dans la plupart des cas, les images naturelles ont des valeurs de kurtosis positives. Les résultats de la simulation à droite de la figure 21 montrent que le kurtosis de l'image «Lena» augmente avec le degré croissant de flou artificiel. L'axe des ordonnées représente la non-gaussianité mesurée par le kurtosis et l'axe des ordonnées représente la longueur  $M$  d'un flou directionnel sur la figure 23(d), la turbulence  $\sigma$  d'un flou gaussien sur la figure 23(e) et le rayon flou sur la Fig. 23(f).

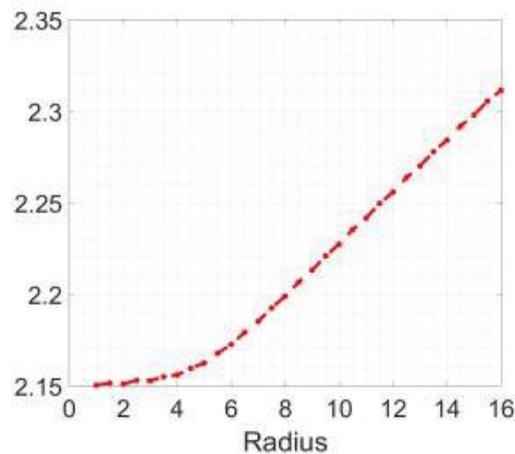
Combinés à des algorithmes heuristiques, les paramètres des modèles peuvent être identifiés à partir des images d'observation floues. Dans cette thèse, ils utilisent un algorithme génétique pour faire évoluer la population la plus adaptée dans l'espace paramétrique.

La méthode proposée est la suivante :

- 1) **Initialisation** : Générez la population de paramètres  $\lambda$  de manière aléatoire en codant et en définissant le taux de croisement  $X$  et le taux de mutation  $\mu$  appropriés. Calculez ensuite la non-gaussianité mesurée par le kurtosis de l'image floue d'entrée.
- 2) **Génération** : Effectuez une itération pour la population du paramètre  $\lambda$  de manière à obtenir une matrice de noyau flou  $H(\lambda)$  par le modèle de dégradation.
- 3) **Evolution** : Utilisez un algorithme de déconvolution non aveugle pour restaurer une image latente temporaire avec  $H(\lambda)$  et faire évoluer la population de la génération suivante à partir des parents en fonction de la mesure de la non gaussianité en tant que fonction de mise en forme.
- 4) **Répéter** : Revenez à l'étape 2 jusqu'à ce que la convergence ou les itérations dépassent un temps tolérant. Le schéma proposé ci-dessus est simple dans le calcul et facile dans la mise en œuvre. L'espace paramétrique  $\lambda$  peut la longueur  $M$  dans le flou en mouvement, l'intensité de la turbulence  $\sigma$  dans le flou de turbulence atmosphérique et le rayon  $R$  dans le flou.



(a)



(d)

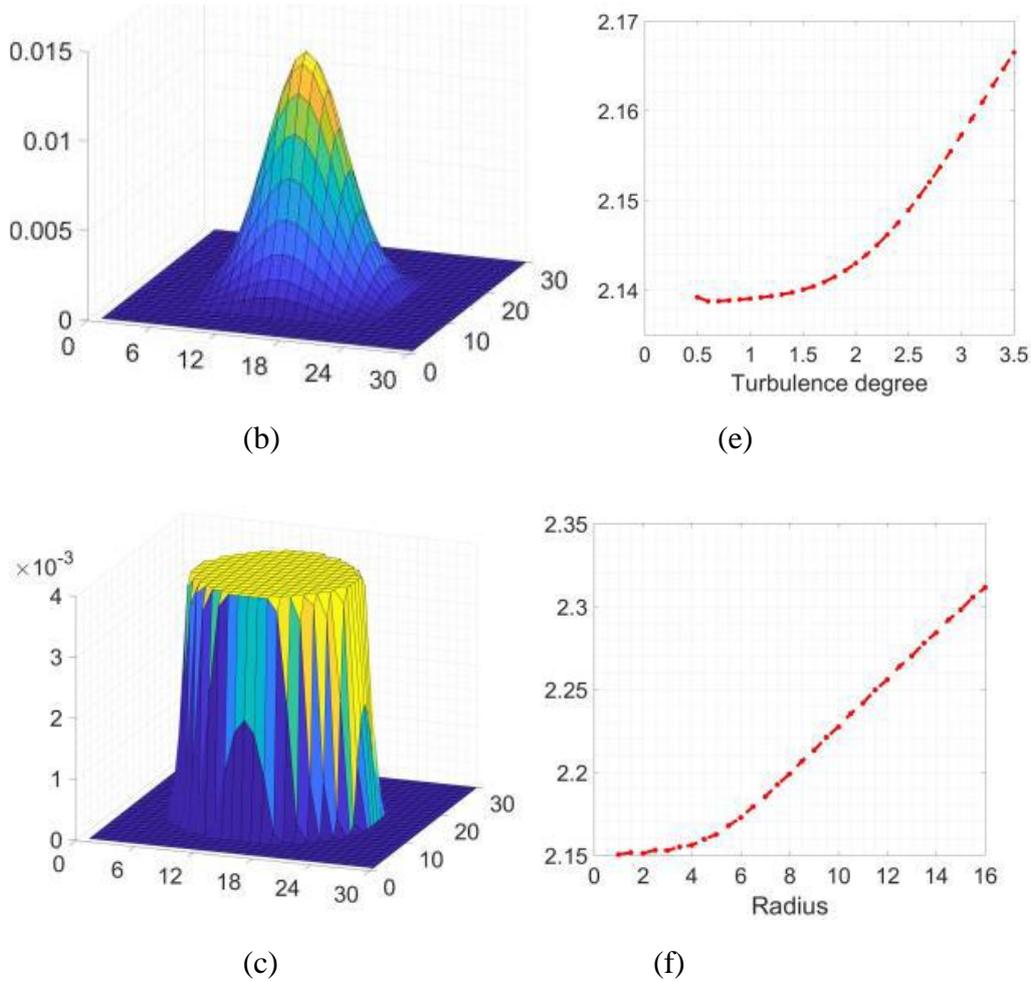


Figure 23 : Modèles de flou et courbes non gaussiennes.

**b) Déconvolution d'images non aveugles**

Pendant ce temps, la méthode de déconvolution non aveugle à la troisième étape peut être une méthode de restauration classique telle que le filtrage de Wiener ou la méthode RL. Dans cette thèse, nous proposons une optimisation sous contrainte algorithmique basée sur  $l_1$ -norm, qui joue un rôle important dans les représentations d'images fragmentées,

$$\hat{f} = \arg \min_f \|f \otimes h\|_2^2 + \alpha \|\nabla f\|_1 \quad (19)$$

Où  $\nabla f$  représente les bords de l'image de  $f$  et  $\alpha$  est le coefficient de régularisation.

Puisque (19) est un problème non convexe, il doit être résolu par un algorithme semi-quadratique [12]. Les variables auxiliaires  $\mu$  sont introduites pour transformer le schéma de minimisation en un problème d'optimisation alternatif

$$\min_{f, \mu} \|f \otimes h\|_2^2 + \beta \|\nabla f - \mu\|_2^2 - \alpha \|\mu\|_1 \quad (20)$$

Où  $\beta$  est un poids intermédiaire qui doit être modifié pendant le schéma en alternance. Comme  $\beta \rightarrow \infty$  la solution de (20) converge vers celle de (19). Par conséquent, à chaque itération, la minimisation de (20) pour une constante donnée  $\beta$  peut être divisée en deux sous-problèmes.[13]

## 2. Travaux récents pour deblurring :

### A. Évaluation des techniques de deblurring d'image

La recherche sur le flou de l'image est en cours depuis longtemps. Une grande partie du domaine du traitement d'image est consacrée au flou d'image. Le processus de restauration des images est axé sur la reconstruction des images appropriées à partir de l'image floue. Comme le bruit corrompt également l'image, nous devons effectuer un débruitage d'image. Le débruitage d'image fait également partie de la procédure de suppression de flou.

Il existe trois principaux types de flou dans une image numérique :

- Flou moyen
- Flou gaussien
- Flou de mouvement

#### a. Techniques de deblurring :

##### ➤ Déconvolution non aveugle :

Dans cette technique, nous avons besoin des connaissances préalables sur les paramètres du noyau flou (angle et longueur de la fonction d'étalement du point (PSF)) pour effectuer la déconvolution.

#### 1. Déconvolution à l'aide de filtre Wiener :

Cette approche nécessite une connaissance préalable des paramètres de la fonction de propagation ponctuelle. Une estimation d'un processus aléatoire souhaité peut être produite par un filtre de Wiener à l'aide d'un filtrage linéaire variant dans le temps d'une image bruitée observée.

#### 2. Déconvolution à l'aide de filtre régularisé :

C'est une autre catégorie de déconvolution non aveugle. Il utilise un filtre de régularisation à des fins de suppression. Cette technique est utile lorsque des informations limitées sur le bruit sont présentes.

#### 3. Déconvolution à l'aide de Lucy-Richardson :

L'algorithme LR peut être utilisé efficacement dans le cas où la fonction d'étalement du point PSF (opérateur de flou) est identifiée, mais que des informations modestes ou inexistantes sont disponibles pour le bruit.

##### ➤ Déconvolution aveugle :

La méthode de déconvolution à l'aveugle, comme son nom l'indique, fonctionne à l'aveuglette ; utile si aucune information sur la fonction d'étalement du point (PSF) ou l'opérateur de flou n'est pas présente.

➤ **Algorithme de Van Cittert Itératif :**

C'est un algorithme itératif pour le flou de l'image. Il diminue la différence entre l'image obtenue en soustrayant l'estimation imagée et l'image enregistrée. La formule mathématique de cet algorithme est ci-dessous [14] :

$$f^{n+1} = f^n + (g - Hf^n) \quad (21)$$

Où  $f^{n+1}$  est la nouvelle approximation de la précédente  $f^n$ ,  $g$  est l'image floue capturée,  $n$  est le nombre d'itérations,  $H$  est PSF et, dans la première itération,  $f^n$  est identique à l'image floue  $g$ . L'un des avantages de cet algorithme est qu'il contient des opérations mathématiques simples. Les limitations de celles-ci sont sensibles au bruit dans une image, instables après quelques limites d'itérations et l'image peut sembler fragile.

➤ **Algorithme de Landweber itératif**

C'est une version améliorée de l'algorithme de Van Cittert. C'est aussi un algorithme itératif. En utilisant cet algorithme, un résultat plus stable et plus fiable peut être obtenu lors de l'exécution d'un nombre supplémentaire d'itérations. L'équation de cet algorithme est ci-dessous : [15].

$$f^{n+1} = f^n + \beta H(gHf^n) \quad (22)$$

Où  $f^{n+1}$  est la nouvelle approximation de la précédente  $f^n$ ,  $g$  est l'image floue,  $n$  est le nombre d'itérations,  $H$  est la fonction de flou appelée PSF,  $\beta$  est une constante qui contrôle la quantité de netteté,  $f^n$  à la première itération identique à l'image floue  $g$ . L'inconvénient de cet algorithme est qu'il prend plus de temps

➤ **Algorithme de carte de Poisson itératif :**

C'est un algorithme itératif. C'est le même que l'algorithme de Richardson-Lucy ; La différence entre ces deux algorithmes réside dans le fait que la carte de Poisson utilise une formule sous forme exponentielle. L'équation de cet algorithme est [16] :

$$f^{n+1} = e^{\left[H\left(\frac{g}{Hf^n}\right)-1\right]} \quad (23)$$

Où  $f^{n+1}$  est la nouvelle approximation de la précédente  $f^n$ ,  $g$  est l'image floue capturée,  $n$  est le nombre d'itérations,  $H$  est PSF. Dans la première itération la valeur de ( $f^n$ ) est identique à celle de l'image floue  $g$ . Il existe certaines limitations telles que le calcul complexe en raison de l'utilisation d'une fonction exponentielle et pour cette raison que l'algorithme devient plus lent.

➤ **Filtres d'affûtage Laplaciens :**

C'est l'un des filtres bien connus, utilisés pour rendre une image plus nette. La netteté des images est également une forme de flou d'image. Le filtre Laplacien est une matrice 3x3 qui possède trois types, les matrices -4, -8 et 9. Les types de noyaux Laplaciens sont donnés ci-dessous :

$$\begin{array}{ccc|ccc|ccc} 0 & 1 & 0 & 1 & 1 & 1 & -1 & -1 & -1 \\ 1 & -4 & 0 & 1 & -8 & 1 & -1 & 9 & -1 \\ 0 & 1 & 0 & 1 & 1 & 1 & -1 & -1 & -1 \end{array}$$

Qui seront utilisées dans cette formule :

$$F = I - [I \otimes LK] \quad (24)$$

Où  $F$  est l'image récupérée,  $I$  est l'image dégradée par Noyau flou laplacien et  $\otimes$  est le processus de convolution. Cette équation est utile pour l'affinage des images et prend moins de temps de calcul, mais il est non-itératif.[17]

➤ **Comparaison de différents algorithmes non aveugle deblurring**

PSF	Algorithme	Opération mathématique	PSNR	Temps d'exécution
<b>Gaussien 3x3</b>	Floue	--	34.6665	--
	Richardson Lucy itérative	4	41.3293	0.1424
	Van cittert	4	42.5365	0.0866
	Landweber itérative	5	42.2980	0.1190
	Carte de Poisson itératif	6	41.8169	0.1168
<b>Rayon</b>	Floue	--	28.6338	--
	Richardson Lucy itérative	4	30.1792	0.1207
	Van cittert	4	30.0598	0.0945
	Landweber itérative	5	30.0362	0.1320
	Carte de Poisson itératif	6	27.8539	0.1219

Tableau 3 : comparaison de tous les algorithmes

**b. Deblurring à l'aide d'une approche de réseau neural**

Un ensemble de systèmes informatiques multiprocesseurs contient des éléments de traitement simples et un degré élevé d'interconnexion adaptative entre ces éléments forme un réseau de neurones. En utilisant cette approche, le flou PSF est supposé uniforme. Divisez l'image en différentes parties, puis appliquez l'algorithme de déconvolution d'image sur différentes parties. Avant d'appliquer l'algorithme de déconvolution sur une image floue, entraînez le réseau à l'aide de l'algorithme de propagation arrière pour le nombre de répétitions utilisées dans le processus de flou afin de déterminer la valeur de la vraie PSF [18]

**c. Deblurring avec la fonction de densité de mouvement**

Cette méthode a été proposée uniquement pour une seule image floue afin d'en éliminer les tremblements. Dans ce procédé, la fonction de densité de mouvement, à savoir MDF, représente le mouvement de la caméra. Comme son nom l'indique, il utilise la fonction de densité de mouvement pour enregistrer la fraction de temps passée dans chaque partie discrétisée de l'espace de toutes les poses possibles de la caméra. Cette méthode est utile pour estimer directement une fonction d'étalement du point variant dans l'espace [19]. Le principal inconvénient de cette méthode est qu'elle ne peut pas fonctionner avec toutes les images.

**d. Deblurring Avec la manipulation des valeurs aberrantes**

Selon cette méthode, les pixels de l'image sont divisés en deux catégories : Inliers et Outliers. Les pixels Inliers sont ceux qui seront couverts par la plage dynamique limitée du capteur de la caméra. Les pixels non conformes sont un bruit non gaussien qui provoque des artefacts de sonnerie dans l'image restaurée résultante. Ce type de bruit ne peut pas être traité par d'autres méthodes existantes. Pour résoudre ce problème, une méthode, à savoir la méthode Expectation Maximization, vise à améliorer de manière répétée la classification des valeurs aberrantes et l'image latente [20].

**e. Deblurring Avec ASDS-AR**

Dans cette méthodologie, un plan ASDS (sélection adaptative de domaine éparse) est présenté, qui prend en compte une progression de sous-lexiques minimaux et nomme de manière adaptative chaque voisinage fixe une référence de sous-mot comme étant la zone inadéquate. Avec ASDS, les affectations IR suggérées sont un modèle de représentation pondéré à la norme L1-standard. Les modèles d'AR les mieux adaptés à un patch donné sont sélectionnés de manière adaptative pour régulariser les structures locales de l'image [21].

**f. Deblurring à l'aide d'utilisation de paires d'images floues / bruits**

Dans cette approche, le processus de déconvolution d'image fonctionne à l'aide d'une image bruit. Ce processus est divisé en deux étapes. La première étape utilise les deux images (floue et bruit) pour trouver un noyau de flou exact. Il est souvent très difficile d'obtenir un noyau flou à partir d'une seule image. La deuxième étape consiste à effectuer une déconvolution résiduelle afin de réduire les artefacts apparaissant comme des signaux superflus communs à la déconvolution d'image [22].

Nom de méthode	Type de floue	Performance	Rapport PSNR
Filtre winner	flou gaussien	mauvaise performance	17.05
Lucy Richardson	flou gaussien	Efficace	21.06
Filtre régularisé	flou gaussien	Efficace	20.10
Déconvolution aveugle	Gaussien et Mouvement	Efficace	26.78
Hyperspectral(PCA)	Hyperspectral image floue	Efficace	22.34
Réseau neural	Gaussien et flou	Très efficace	30.11
Densité de mouvement	Mouvement	Efficace	24.31
ADSD-AR	Gaussien	Très efficace	31.20
Paire d'images floues / bruits	Gaussien	Efficace	23.37
Evaluation des valeurs aberrantes	Gaussien	Efficace	21.90

Tableau 4 : Analyse de performance des méthodes de déblurring

### 3. Segmentation du flou de défocalisation basé sur LBP :

#### a. Métrique de floue proposée à base de LBP :

Les motifs binaires locaux (LBP) [23] ont permis de résoudre des problèmes de vision par ordinateur tels que la segmentation de texture [24], la reconnaissance de visage [25], la soustraction d'arrière-plan [26] et la reconnaissance de surfaces texturées en 3D [27]. Le code LBP d'un pixel  $(x_c, y_c)$  est défini comme suit :

$$\text{LBP}_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} S(n_p - n_c) \times 2^p \quad (25)$$

$$\text{Avec } S(x) = \begin{cases} 1 & |x| \geq T_{LBP} \\ 0 & |x| < T_{LBP} \end{cases}$$

Où  $n_c$  est l'intensité du pixel central  $(x_c, y_c)$ ,  $n_p$  correspond aux intensités des P pixels voisins situés sur un cercle de rayon R centré sur  $n_c$ , et  $T_{LBP} > 0$  est un petit seuil positif afin de réaliser la robustesse des régions d'image plate comme dans [24]. La figure 4 montre les emplacements des pixels voisins  $n_p$  pour P = 8 et R = 1. En général, les points  $n_p$  ne tombent pas au centre des pixels de l'image, l'intensité de  $n_p$  est donc obtenue par interpolation bilinéaire.

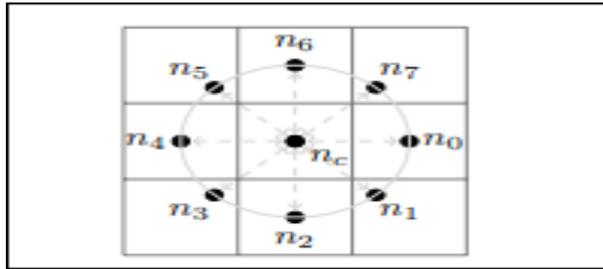


Figure 24 : 8-bit LBP avec  $P=8, R=1$

Une version invariante par rotation de LBP peut être obtenue en effectuant le décalage droit au niveau du bit qui minimise la valeur du code LBP lorsqu'il est interprété comme un nombre binaire [28]. De cette manière, le nombre de motifs uniques est réduit à 36. Ojala et al. ont trouvé que tous les modèles invariants de rotation ne soutiennent pas la rotation de la même manière [28], et ont proposé d'utiliser uniquement des modèles uniformes, qui sont un sous-ensemble des modèles invariants de rotation. Un modèle est uniforme si la séquence circulaire de bits ne contient pas plus de deux transitions de un vers zéro ou de zéro vers un.

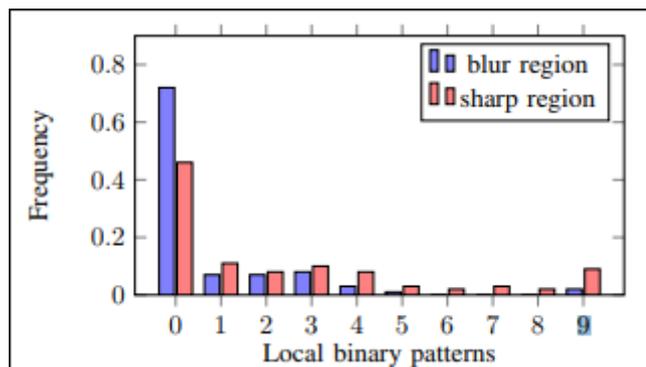


Figure 25 : Distribution du code LBP dans les régions floues et pointues.

Les modèles non uniformes sont alors tous traités comme des casiers 0 à 8 qui sont les comptes des modèles uniformes ; la valeur 9 est le nombre de motifs non uniformes. Les données sont échantillonnées à partir de 100 images partiellement floues.

La figure 25 montre l'histogramme des neuf motifs LBP uniformes apparaissant dans les régions floues et non floues de 100 images sélectionnées aléatoirement à partir d'un jeu de données public de 704 images partiellement floues [29], chacune étant dotée d'un segment segmenté à la main. Image de vérité terrestre indiquant les régions floues et non floues. La case 9 correspond au nombre de motifs non uniformes. La fréquence des motifs 6, 7, 8 et 9 dans les régions floues est nettement inférieure à celle des régions nettes. L'explication intuitive en est que dans les zones plus lisses, la plupart des pixels voisins auront une intensité similaire à  $n_c$ , et les chances qu'un voisin soit déclenché sont plus faibles, ce qui rend plus probable les motifs uniformes numérotés inférieurs avec moins de voisins déclenchés. Des exemples d'histogrammes LBP de taches précises et floues spécifiques sont donnés à la figure 4, qui présente également ce comportement attendu.

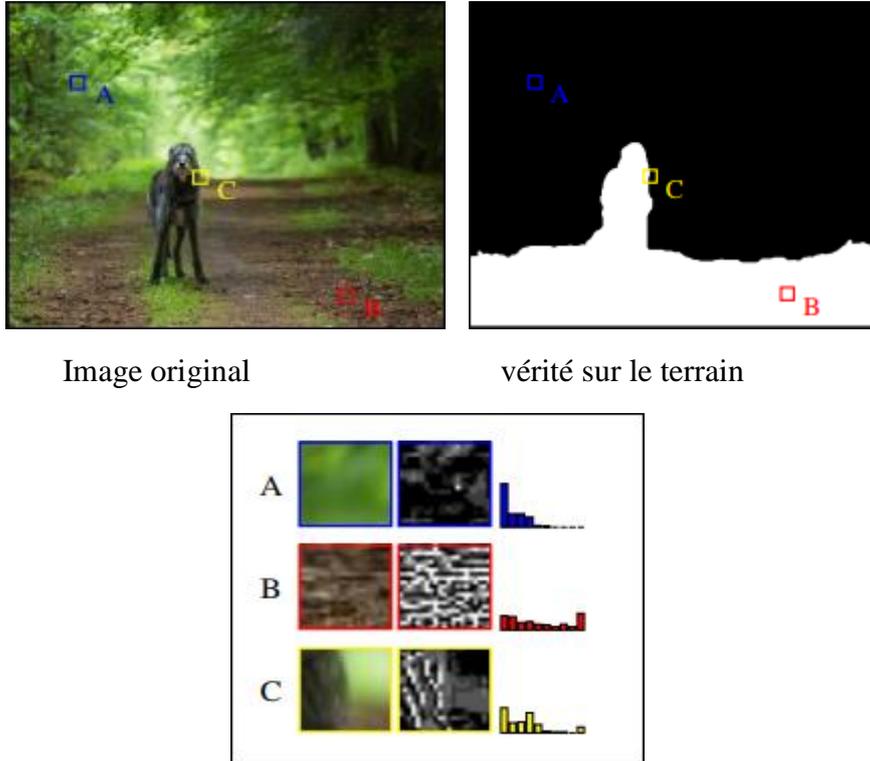


Figure 26 : Histogramme de modèles de lombalgie dans trois zones différentes échantillonnées à partir de flou (A), net (B) et transitif (C) zones respectivement. Dans l'image de vérité au sol, le blanc désigne la région pointue et le noir la région floue.

La métrique de netteté proposée exploite ces observations :

$$m_{LBP} = \frac{1}{N} \sum_{i=6}^9 n(LBP_{8,1}^{riu2} i) \quad (26)$$

Où  $n(LBP_{8,1}^{riu2} i)$  est le nombre de motifs LBP à 8 bits uniformes et invariants en rotation de type  $i$ , et  $N$  le nombre total de pixels dans la région locale sélectionnée, ce qui permet de normaliser la métrique de sorte que  $m_{LBP} \in [0,1]$ . L'un des avantages de la mesure de la netteté dans le domaine LBP est que les fonctions LBP sont robustes pour les changements d'éclairage monotones qui se produisent fréquemment dans les images naturelles.

### b. Nouvel algorithme de segmentation de floue

Cette section présente l'algorithme de segmentation floue / nette régions avec le métrique de netteté basée sur LBP. L'algorithme comporte quatre étapes principales : la génération de cartes de netteté à plusieurs échelles, l'initialisation alpha matting, le calcul de cartes d'alpha et l'inférence de netteté à plusieurs échelles.

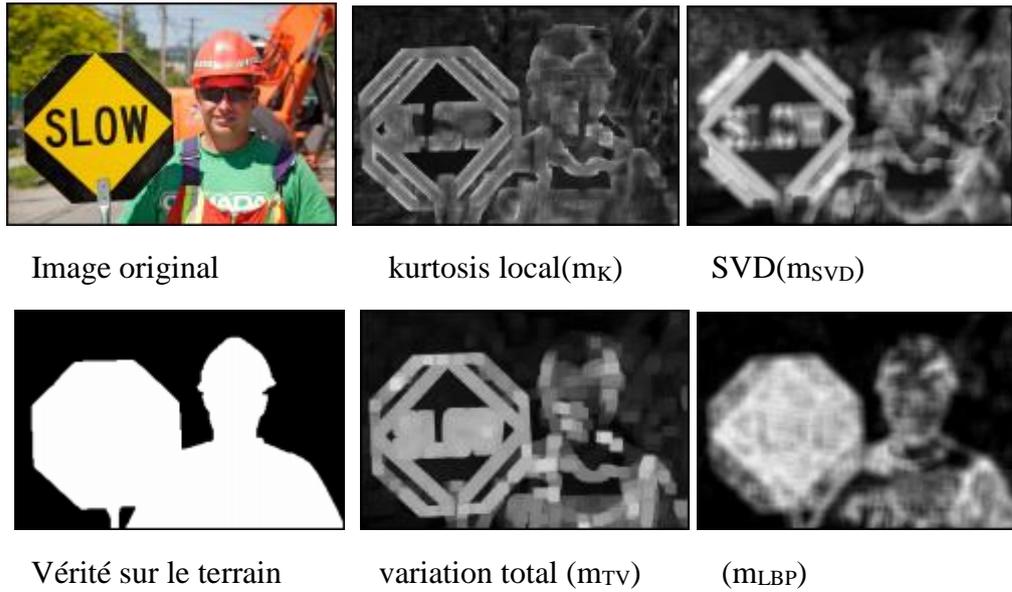


Figure 27 : Réponses métriques pour une image échantillon pour différentes métriques de netteté

### 1. Génération de cartes de netteté à plusieurs échelles :

Dans la première étape, des cartes de netteté à plusieurs échelles sont générées à l'aide de mLBP. La métrique de netteté est calculée pour un patch local autour de chaque pixel de l'image. Les cartes de netteté sont construites à trois échelles, l'échelle se rapportant à la taille du patch local. En utilisant une image intégrale [30], des cartes de netteté peuvent être calculées en temps constant par pixel pour un P et un R. fixes.

### 2. Initialisation Alpha Matting :

L'alpha mat est le processus de décomposition d'une image en avant-plan et en arrière-plan. Le modèle de formation d'image peut être exprimé comme suit :

$$I(x, y) = \alpha_{x, y} F(x, y) + (1 - \alpha_{x, y}) B(x, y) \quad (27)$$

Où le cache alpha,  $\alpha_{x, y}$ , est la valeur d'opacité sur la position du pixel  $(x, y)$ . Cela peut être interprété comme la confiance qu'un pixel est au premier plan. En règle générale, l'alpha-mat nécessite qu'un utilisateur marque de manière interactive les pixels connus de premier plan et d'arrière-plan, en initialisant ces pixels avec  $\alpha = 1$  et  $\alpha = 0$ , respectivement.

En interprétant « premier plan » comme « net » et arrière-plan comme « flou », nous avons initialisé automatiquement le processus de matelassage alpha en appliquant un double seuil aux cartes de netteté calculée à l'étape précédente pour produire une valeur initiale de  $\alpha$  pour chaque pixel:

$$\text{mask}^s(x, y) = \begin{cases} 1 & \text{si } m_{LBP}(x, y) > Tm1. \\ 0 & \text{si } m_{LBP}(x, y) < Tm2. \\ m_{LBP}(x, y) & \text{sinon} \end{cases} \quad (28)$$

Où s indexe l'échelle, c'est-à-dire que les masques  $(x, y)$  sont la carte a initiale de la s-ème échelle.

### 3. Calcul de la carte alpha :

La carte  $\alpha$  a été résolue en minimisant la fonction de coût suivante proposée par Levin [31] :

$$E(\alpha) = \alpha T L \alpha + \lambda (\alpha - \tilde{\alpha}) T (\alpha - \tilde{\alpha}) \quad (29)$$

Où  $\alpha$  est la carte  $\alpha$  vectorisée,  $\tilde{\alpha} = \text{mask}(x, y)$  est l'une des cartes alpha d'initialisation vectorisées de l'étape précédente et  $L$  est la matrice laplacienne matifiante. Le premier terme est le terme de régulation qui garantit la régularité, et le second terme est le terme d'ajustement des données qui encourage la similarité avec  $\tilde{\alpha}$ . Pour plus de détails sur l'équation 18, on se reportera à [31].

### 4. Inférence multi-échelle :

Après avoir déterminé la carte alpha à trois échelles différentes, une modèle graphique multi-échelle a été adoptée pour prendre la décision finale [32]. L'énergie totale sur le modèle graphique est exprimée comme suit :

$$E(h) = \sum_{s=1}^3 \sum_i h_i^s - \check{h}_i^s + \beta (\sum_{s=1}^3 \sum_i \sum_{j \in N_i^s} |h_i^s - h_j^s| + \sum_{s=1}^2 \sum_i |h_i^s - h_i^{s+1}|) \quad (30)$$

Où  $\check{h}_i^s$  est la carte alpha de l'échelle  $s$  à l'emplacement du pixel  $i$  qui a été calculée à l'étape précédente et  $h_i^s$  est la netteté à inférer. Le premier terme à droite est le terme unaire qui est le coût d'attribution de la valeur de netteté  $h_i^s$  au pixel  $i$  à l'échelle  $s$ . Le second est le terme par paire qui applique la régularité à la même échelle et à différentes échelles. Le poids  $\beta$  régule l'importance relative de ces deux termes. Optimisation de l'équation 30 a été réalisée à l'aide de la propagation par croyances en boucle [33].

La sortie de l'algorithme est  $h^3$ , qui est la carte de netteté inférée à la plus grande échelle. Ceci est une image en niveaux de gris, où une intensité plus élevée indique une plus grande netteté.

#### c. Évaluation d'algorithme de segmentation de flou :

Les algorithmes de segmentation du flou ont été testés à l'aide d'un ensemble public de données d'image floue constitué de 704 images partiellement floues et d'images de vérité de terrain segmentées à la main [16]. Chaque image a été segmentée en régions nettes et floues. La métrique de netteté  $m_{LBP}$  a été calculée avec  $T_{LBP} = 0,016$ . Les échelles de la carte de netteté étaient des régions locales carrées de  $11 \times 11$ ,  $15 \times 15$  et  $21 \times 21$  pixels. Les seuils utilisés lors de l'étape alpha-matting étaient  $T_{m1} = 0,3$  et  $T_{m2} = 0,01$ . Un poids  $\beta = 0,5$  a été utilisé dans l'étape d'inférence multi-échelles.

Ils ont comparé son algorithme à six méthodes de comparaison,. Su et al. ont simplement calculé une carte de netteté en utilisant  $m_{SVD}$ , Vu combinait une netteté spectrale et spatiale (appelées  $S1$  et  $S2$  dans le document d'origine) ils utilisant une moyenne géométrique a utilisé tous les  $m_{GHS}$ ,  $m_K$ ,  $m_{LDA}$ ,  $m_{APS}$  avec un classifieur de Bayes naïf et un modèle d'inférence multi-échelle [32].

- **F-mesure :**

Dans une autre expérience, nous avons utilisé un seuil adaptatif dépendant de l'image proposé par [31] pour la segmentation, le seuil étant défini comme suit :

$$T_{seg} = \frac{2}{W \times H} \sum_{x=1}^W \sum_{y=1}^H I(x, y) \quad (31)$$

Où,  $W$ ,  $H$  sont la largeur et la hauteur de la netteté finale carte I. Puis, semblable à [25], la moyenne harmonique pondérée mesure ou F-mesure de précision et le rappel a été calculé en comparaison. La définition est la suivante :

$$F_{\beta} = \frac{(1+\beta^2) \times \text{precision} \times \text{recall}}{\beta^2 \times \text{precision} + \text{recall}} \quad (32)$$

Ici,  $\beta$  était réglé à 0,3

Notez que la carte de segmentation de Zhu a été produite par coupe de graphique au lieu d'un simple seuillage de la profondeur carte. Les paramètres que nous avons utilisés étaient les mêmes que ceux suggérés dans leur article qui sont  $\lambda_0 = 1000$ ,  $\sigma\lambda = 0.04$ ,  $\tau = 2$ . Exemples de cartes de segmentation d'images

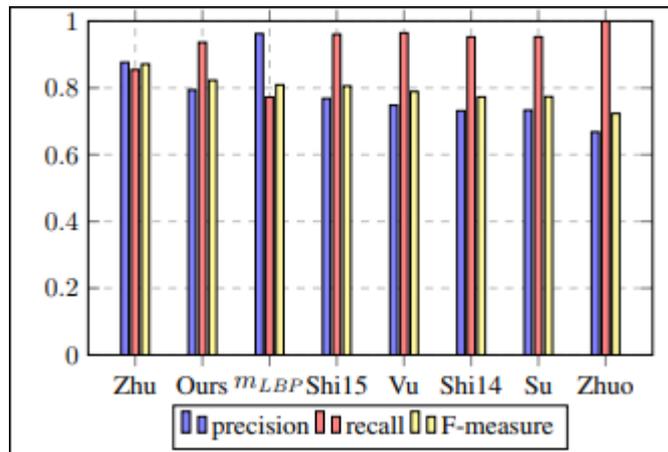


Figure 28 : Mesures de précision, de rappel et F pour les seuils adaptatifs

Le résultat de Zhu est obtenu en utilisant une coupe de graphique au lieu d'un simple seuillage comme suggéré dans leur document. Notez qu'en utilisant un seuil plus petit (c'est-à-dire  $T_{seg} = 0,3$ ), notre méthode peut obtenir des performances comparatives (précision = 0,863, rappel = 0,868, mesure F = 0,864) avec Zhu.

Flou de segmentation	Moy. temps d'exécution
Ours	27.75s
$m_{LBP}$	40 ms
Zhu	12min

Tableau 5 : Comparaison en temps d'exécution de différentes méthodes de segmentation du flou

Une comparaison en temps d'exécution des algorithmes complets de segmentation est présentée dans le tableau 3. Le même réglage a été utilisé pour la mesure de la durée d'exécution, comme indiqué dans le tableau I. Étant donné les performances de  $m_{LBP}$  dans la courbe de précision et de rappel et la mesure F, il présente un avantage de vitesse significatif par rapport aux autres.

- **Discussion :**

Vous avez peut-être remarqué les limites irrégulières de leur carte de segmentation lorsqu'il existe une discontinuité de profondeur distincte entre l'avant-plan et l'arrière-plan. En effet, la netteté est mesurée localement. Il est inévitable d'incorporer des zones de netteté variées dans les régions, en particulier autour des bords où se produit la discontinuité de la profondeur. Par conséquent, la zone nette est agrandie à l'étape d'initialisation du tapis alpha (étape B). Zhu a résolu ce problème en prenant en compte les informations de finesse et de contour de couleur lors de l'étape d'étiquetage de cohérence, mais échouait également dans les cas où la profondeur change progressivement.

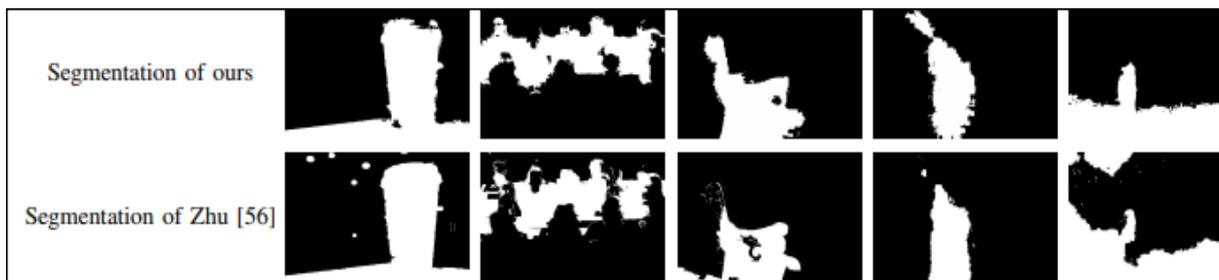


Figure 29 : Comparaison de la carte de segmentation binaire avec Zhu et al

Ils ont proposé une métrique de netteté très simple mais efficace pour la segmentation du flou. Cette métrique est basée sur la distribution de motifs LBP uniformes dans les régions d'image floues et nettes [33], on a utilisé cette approche dans notre travail.

## Conclusion

Dans ce chapitre, nous avons présenté les travaux menés sur la déconvolution et la segmentation de flou des images numériques, et les techniques utilisées pour de segmentation, et les techniques de deblurring. Dans le chapitre suivant on va montrer la description détaillée de notre application dédiée aux déconvolutions automatiques des images floues.

# *Chapitre IV*

## *Conception et*

## *Implémentation*



## Partie I : conception et modélisation

UML est une notation permettant de modéliser un problème de façon standard. Ce langage est né de la fusion de plusieurs méthodes existantes auparavant, et est devenu désormais la référence en termes de modélisation objet, à un tel point que sa connaissance est souvent nécessaire pour obtenir un poste de développeur objet.

### 1. Définition d'UML

UML (Unified Modeling Language) est une méthode de modélisation orientée objet développée en réponse à l'appel à proposition lancé par l'OMG (Object Management Group) dans le but de définir la notation standard pour la modélisation des applications construites à l'aide d'objets. Elle est héritée de plusieurs autres méthodes telles qu'OMT (Object Modelling Technique) et OOSE (Object Oriented Software Engineering) et Booch. Les principaux auteurs de la notation UML sont GradyBooch, Ivar Jacobson et Jim Rumbaugh. UML est une méthode utilisant une représentation graphique. L'usage d'une représentation graphique est un complément excellent à celui de représentations textuelles. En effet, l'une comme l'autre est ambiguë mais leur utilisation simultanée permet de diminuer les ambiguïtés de chacune d'elles. Un dessin permet bien souvent d'exprimer clairement ce qu'un texte exprime difficilement et un bon commentaire permet d'enrichir une figure. UML et les Design Patterns fournissent aux développeurs et aux étudiants les connaissances indispensables pour comprendre et maîtriser l'analyse et la conception orientées objet (A/COO). Son objectif ne se limite pas à la création de diagrammes et à la notation UML ; il vise à leur application réelle dans le contexte de la conception de systèmes logiciels.

#### A. Les modèles d'UML :

La modélisation consiste à créer une représentation simplifiée d'un problème. Grâce au modèle il est possible de représenter simplement un problème, un concept et le simuler.

Les 14 diagrammes UML sont tous programmés dans cet organigramme :

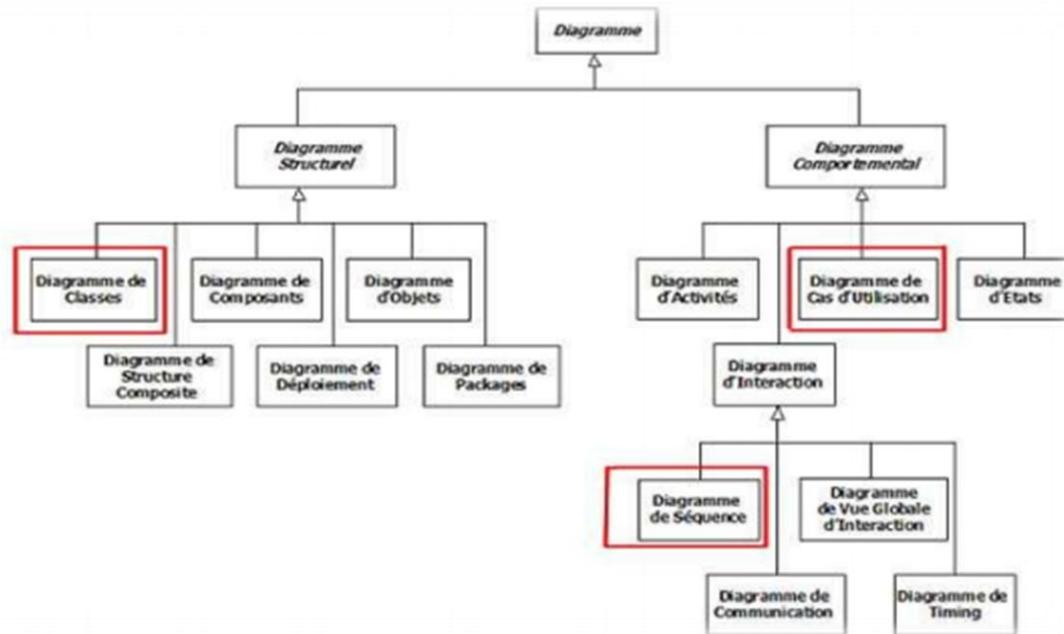


Figure 30 : organigramme des diagrammes de langage UML.

Parmi ces 14 diagrammes, on utilise 3 diagrammes : le diagramme de cas d'utilisation, le diagramme de séquence et le diagramme de classe.

#### a. Le diagramme de cas d'utilisation :

Le diagramme de cas d'utilisation permet de représenter visuellement une séquence d'actions réalisées par un système, représenté par une boîte rectangulaire, produisant un résultat sur un acteur, appelé acteur principal, et ceci indépendamment de son fonctionnement interne. Un acteur, au sens UML, représente le rôle d'une entité externe (utilisateur humain ou non) interagissant avec le système. Il est représenté par un bonhomme en fil de fer (en anglais stickman). On représente généralement à gauche l'acteur principal (idéalement il y en a un seul), et à droite les acteurs secondaires. Il est à noter qu'un utilisateur peut amener à jouer plusieurs rôles vis-à-vis du système et à ce titre être modélisé par plusieurs acteurs. L'objectif poursuivi par les cas d'utilisation est de permettre de décrire, dans des documents lisibles par tous, la finalité des interactions du système et de ses utilisateurs. [34]

#### b. Diagramme de séquence :

Diagramme d'interaction qui représente les objets participant à une interaction particulière et les messages qu'ils échangent, sont organisés en séquences horaires. Axé sur ce que fait un système et non sur la manière dont il le fait, un diagramme de séquence définit la logique d'une instance particulière d'un cas d'utilisation. En général, dans un diagramme de séquence, la Dimension verticale représente les heures (de haut en bas) et la Dimension horizontale représente les différents objets. [35]

#### c. Diagramme de classes :

Les diagrammes de classes jouent un rôle central dans l'analyse et le design orientés objet. Ils présentent un ensemble d'éléments de modèle statiques, leur contenu (structure interne) et leurs relations aux autres éléments. Les principaux éléments représentés dans un diagramme de classes sont les classes, les packages, les associations, les héritages et les dépendances. Les

diagrammes de classes sont créés sur les packages, les classes, les interfaces, les composants et les collaborations. [36]

### B. Entreprise Architect :

L'Architecture d'Entreprise est la logique structurante pour les processus métiers et l'infrastructure informatique, reflétant les exigences d'intégration et de standardisation du modèle opératoire de l'entreprise. [37]

#### 1. Diagramme de cas d'utilisation :

Acteur	Cas d'utilisation
Administrateur	-choix de (utilitaire ou floue focus) <b>utilitaire</b> :-choix de filtre utilisé (Winner, Lucy, Régularisé, Blind) -importer une image vers l'interface MATLAB. -choix de type de PSF (gaussien, moyen, mouvement, focus) -déconvolution <b>Focus</b> :- -importer une image vers l'interface MATLAB. -segmentation -déconvolution (la partie floue) -convolution (la partie nette) -fusionner les 2 images

Tableau 6 : administrateur.

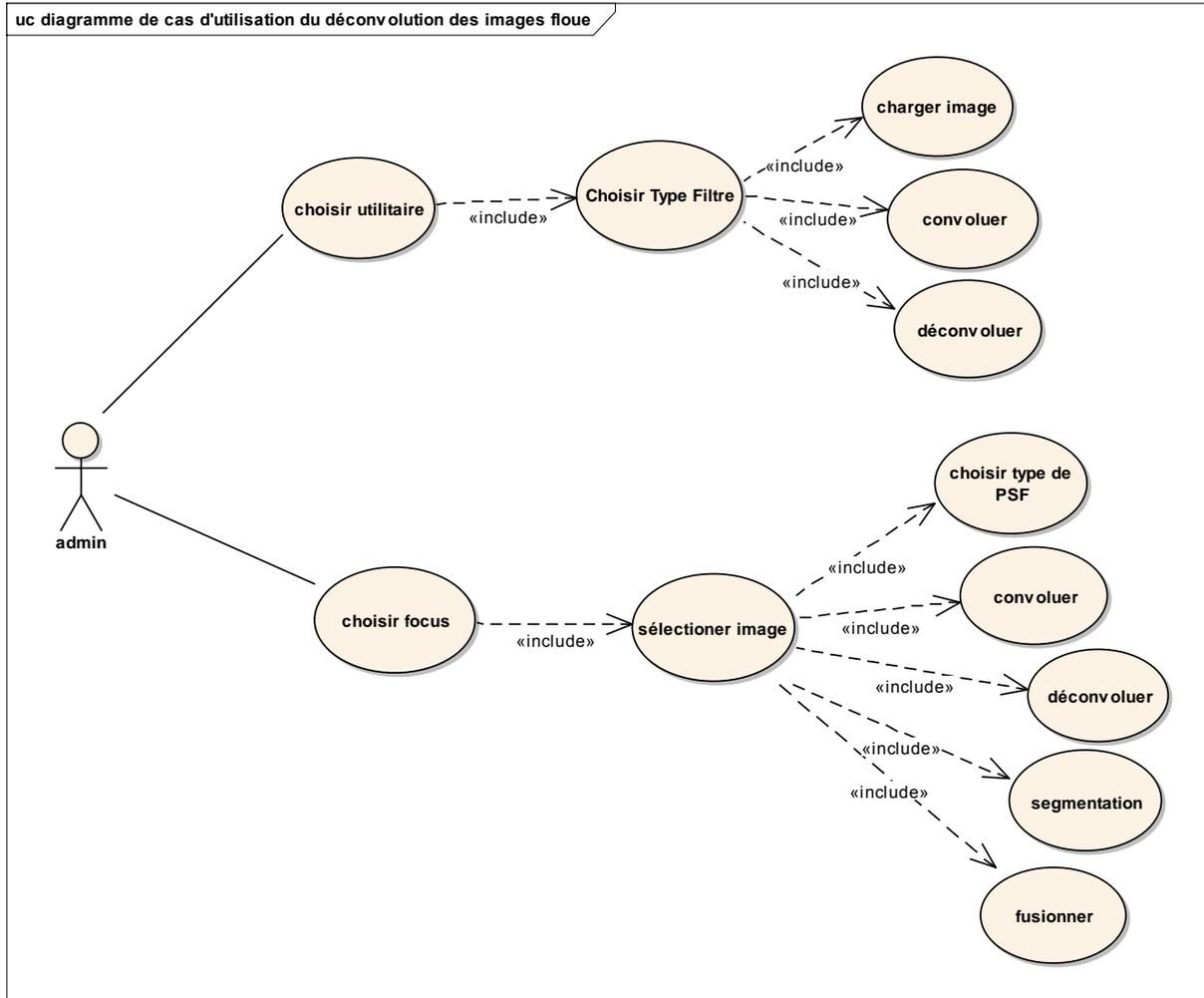


Figure 31 : diagramme de cas d'utilisation.

**2. Diagramme de séquence :**

Description sommaire	
titre	Déconvolution des images floues
But	Admin doit déconvoluer l'image floue
acteur	Admin
Description des enchainements	
précondition	Image floue
enchainement	<ul style="list-style-type: none"> <li>-admin demande de système charger une image</li> <li>- admin choisit le filtre</li> <li>- admin choisit le PSF</li> <li>-admin remplit les valeurs de PSF</li> <li>- admin demande de système la convoluer la partie nette</li> <li>- admin demande de système de déconvoluer la partie floue</li> <li>- admin demande de système de segmenter les 2 résultats</li> <li>- admin demande de système fusionner les 2 partie</li> </ul>
Post condition	Image déconvoluer

*Tableau 7 : déconvolution d'image floue.*

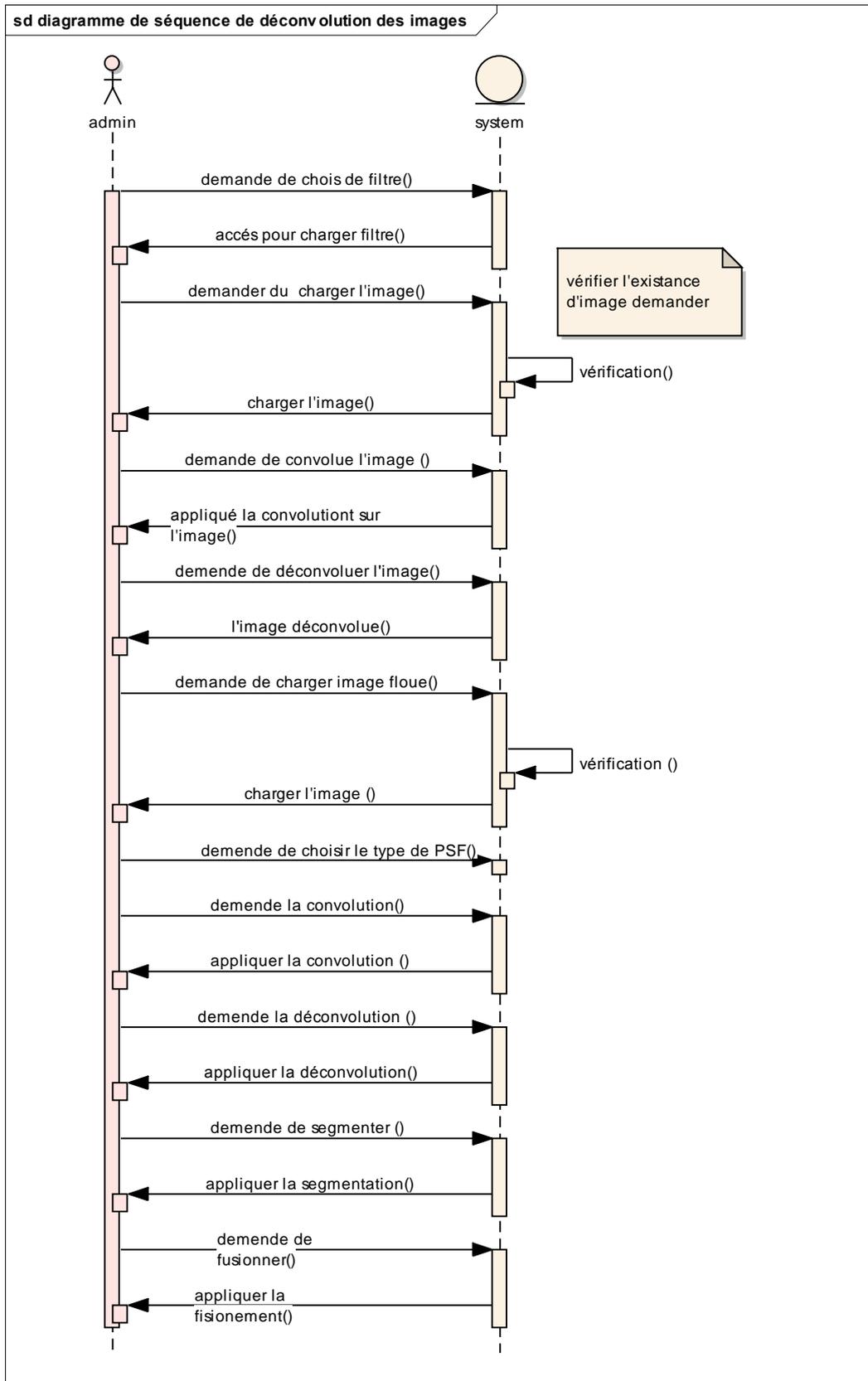


Figure 32 : diagramme de séquence

3. Diagramme de classes :

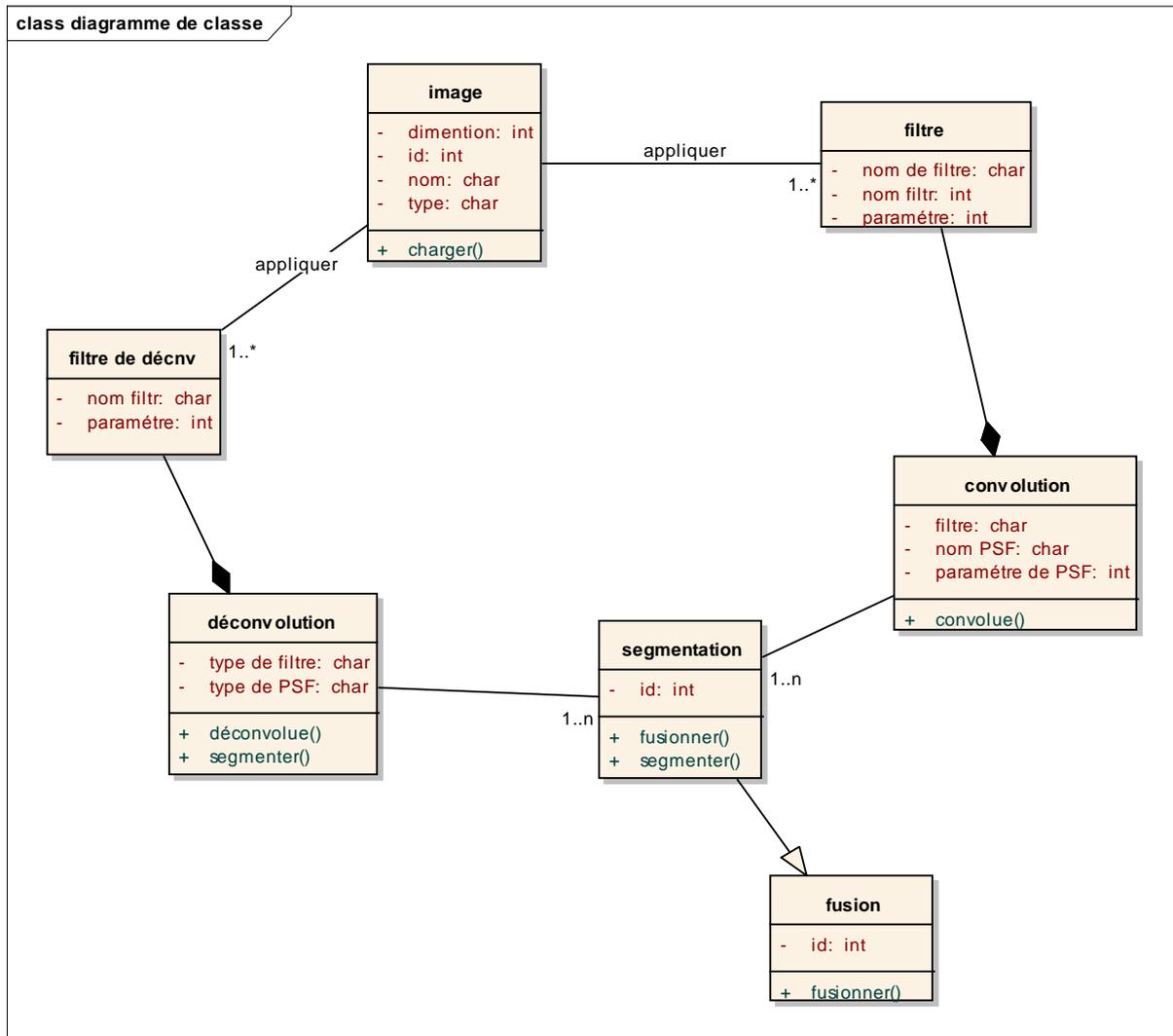


Figure 33: diagramme de classe

Dans ce chapitre, on a utilisé l’UML comme une notation de base pour la modélisation de la conception. On s’est servi des diagrammes de cas d’utilisation, de séquence et de classe. Après cette modélisation, on passe à la phase d’implémentation dans la prochaine partie.

## Partie II : implémentation

Dans cette partie notre objectif est de présenter l'application développée sous l'environnement Matlab

### 1. L'environnement de programmation MATLAB :

Matlab est un langage de programmation de haut niveau pour le calcul numérique développé par Mathworks. Il est particulièrement performant pour le calcul matriciel, car sa structure de données est basée sur les matrices, et il dispose de possibilités d'affichage très riches. Il s'agit d'un langage à interpréter, ce qui permet un développement très rapide. De plus, il existe des boîtes à outils (Toolbox) optionnelles mais très utiles dans certains domaines comme l'optimisation, le traitement d'image et du signal, l'apprentissage (réseaux de neurones. . .), l'automatique (Simulink), etc. Ce logiciel est de plus très utilisé tant dans le monde industriel que dans le monde universitaire. MATLAB possède une riche bibliothèque de fonctions prédéfinies qui simplifie grandement l'élaboration de programmation plus complexes.

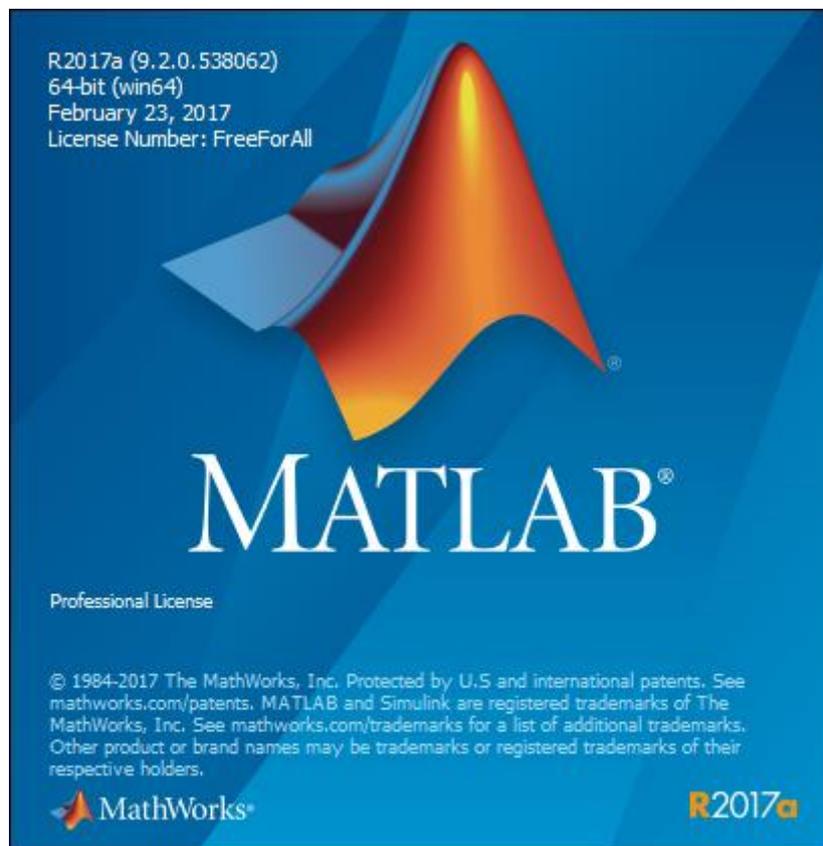


Figure 34 : l'environnement Matlab.

#### a. Intérêts :

- Programmation infiniment rapide pour le calcul et pour affichage.
- Une librairie très riche.
- Possibilité d'inclure une programmation en C/C++.
- Langage interprété : pas de compilation alors pas d'attente pour compiler.
- Code facile à comprendre et très lisible.

- Possibilité d'exécuter du code en dehors du programme.
- Une aide très bien faite.

**b. Inconvénients :**

- Vitesse de calcul moine rapide qu'en C/C++.
- Application auto-exécutable peu pratique.

**2. Analyse des besoins**

Notre tendance est de développer une application, capable de déconvoluer le flou des différentes images à partir des approches, qui permet d'atteindre l'objectif

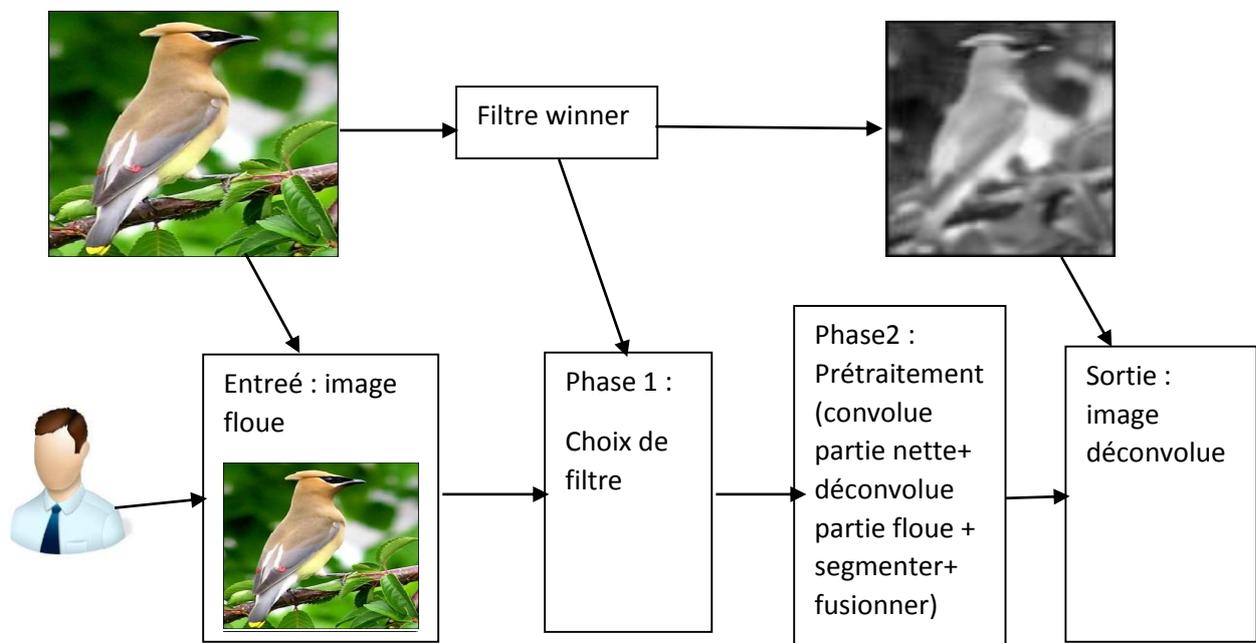
**3. Notre approche**

Figure 35 : présentation de l'approche de l'application.

#### 4. Interface de l'application implémenté



Figure 36 : interface menu principal de l'application.

Le page de menu principale de notre application se compose de deux opérations essentielles :

- 1- Utilitaire
- 2- Floue focus

Chacune de ces options citées précédemment nous conduit vers une nouvelle fenêtre interface pour appliquer un traitement spécifique.

1ere partie utilitaire : le bouton utilitaire ouvre cette fenêtre qui contient les quatre filtres :

- Filtre Wiener.
- Filtre Lucy Richardson.
- Filtre Régularisé.
- Filtre Blind.

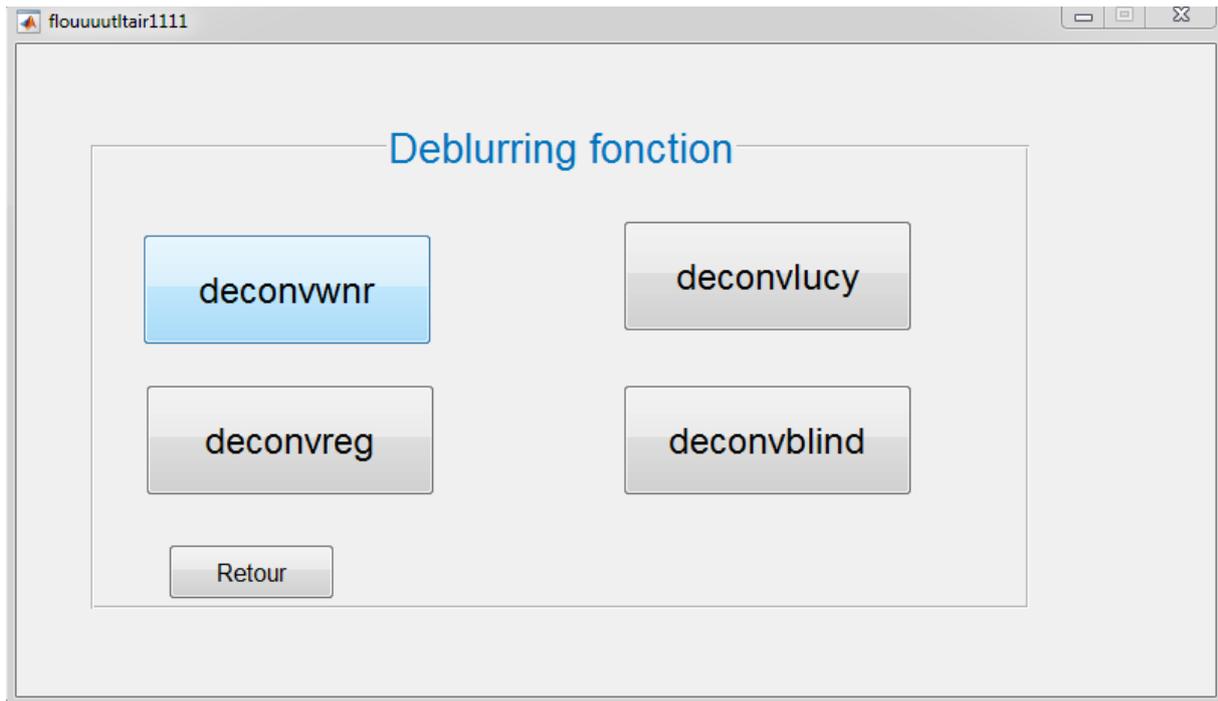


Figure 37 : fenêtre des filtres de déconvolution.

A partir de cette fenêtre nous pouvons réaliser les différents types des filtres comme suit :

- Filtre Wiener.

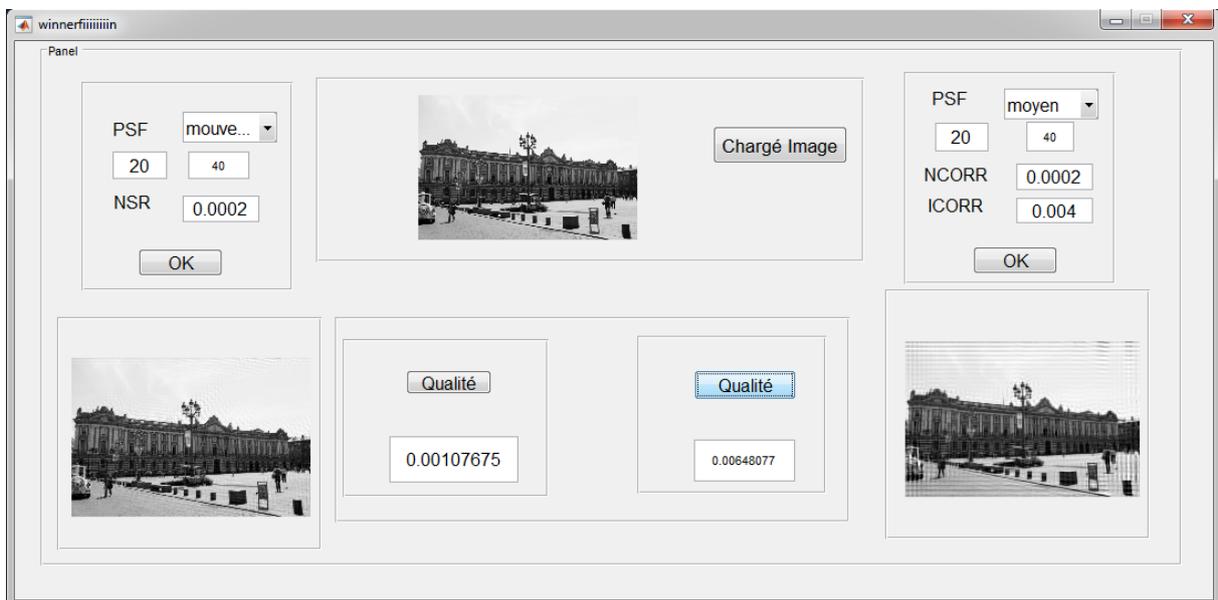


Figure 38 : exemple de déconvolution avec le filtre Wiener.

Le principe de l'implémentation de cette fenêtre c'est :

- Charger une image puis faire le test avec les deux méthodes de filtre (convolution déconvolution) et à la fin calculée la qualité de l'image original avec le résultat.

- Filtre Lucy Richardson.

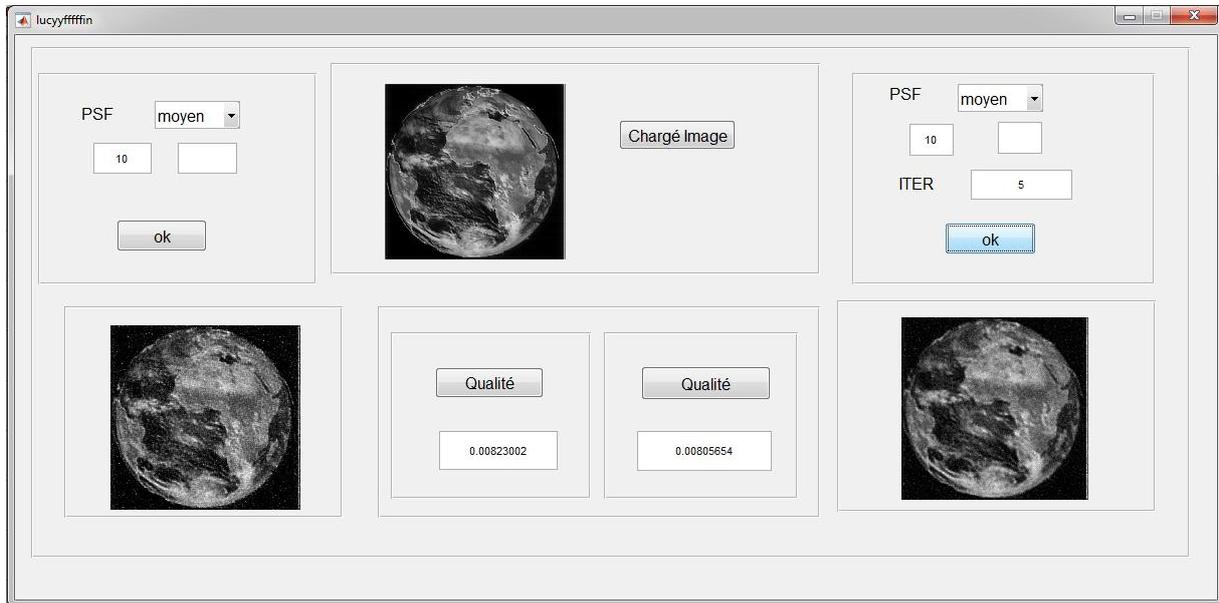


Figure 39 : le résultat de déconvolution avec filtre Lucy Richardson.

- Filtre régularisé.

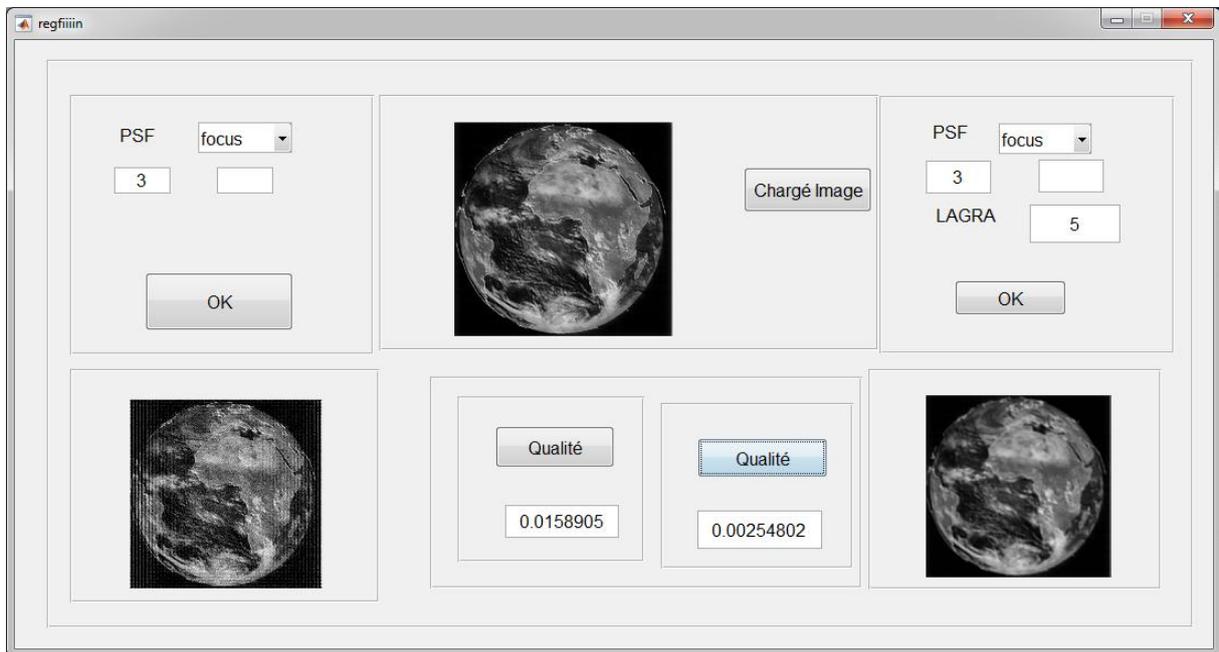


Figure 40 : fenêtre de déconvolution avec filtre régularisé.

- Filtre Blind

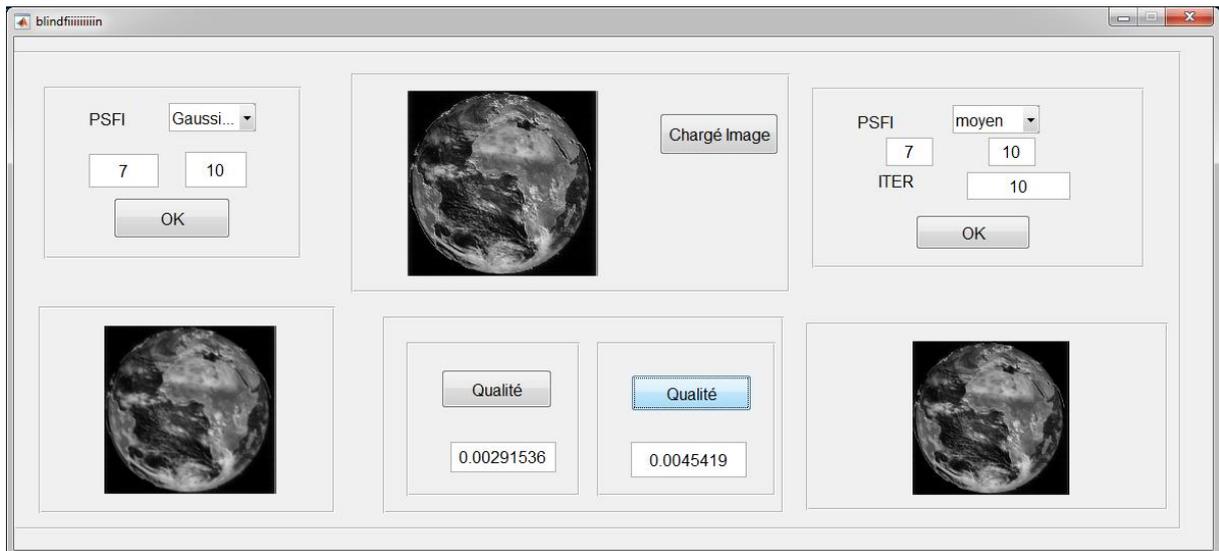


Figure 41 : fenêtre de déconvolution avec filtre Blind.

## 5. Résultat et discussion

2eme partie focus : le bouton focus ouvre cette fenêtre

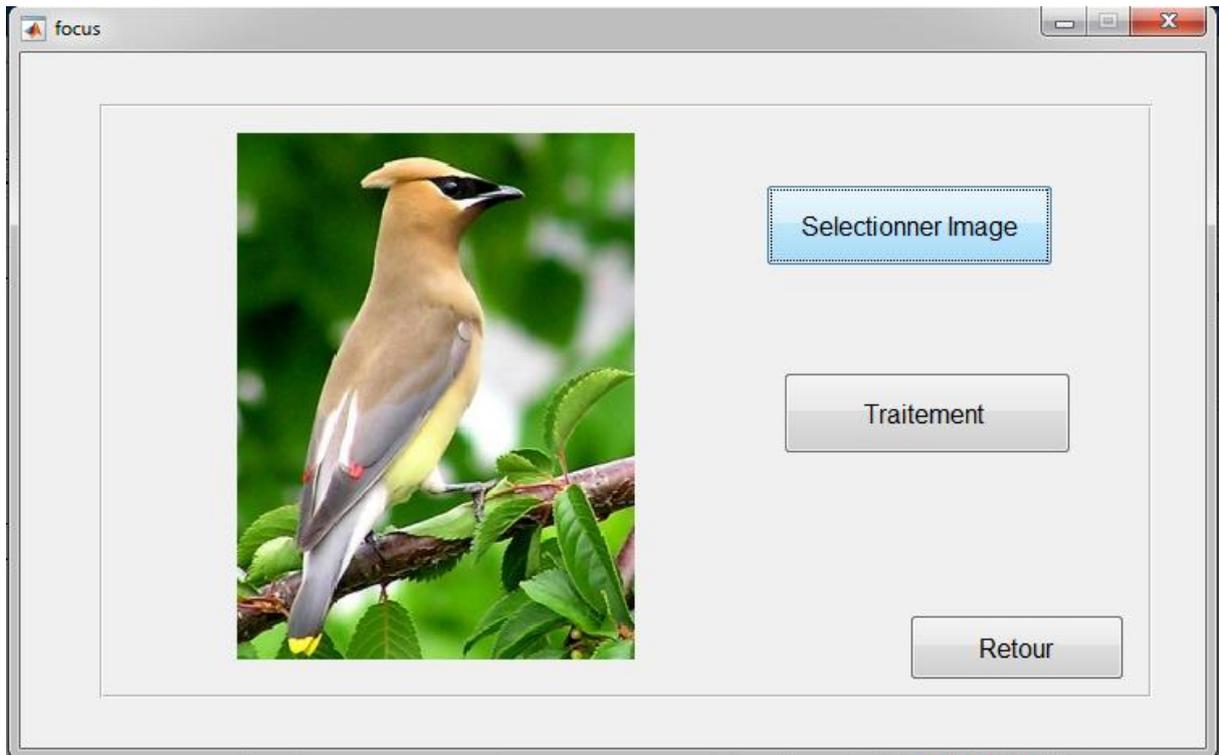


Figure 42 : fenêtre de focus.

Le principe de l'implémentation de cette fenêtre c'est :

- Charger une image
- traitement d'image

Le bouton traitement est donné par la fenêtre suivante :

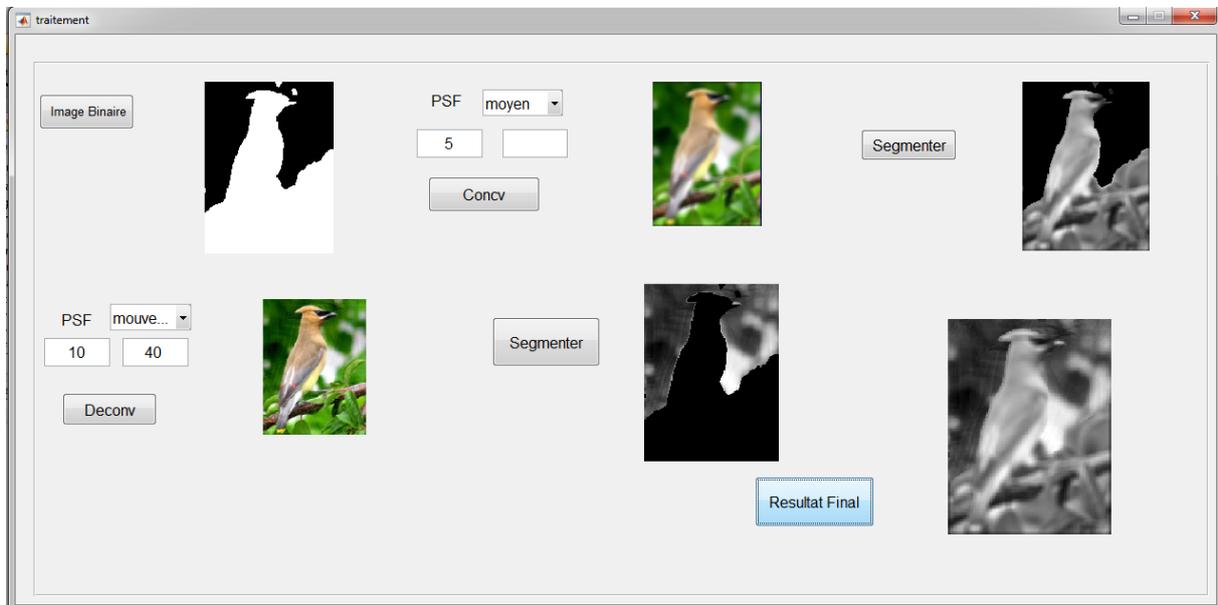


Figure 43 : fenêtre de résultat de traitement

Le principe de l'implémentation de cette fenêtre c'est :

- réduire une image binaire, dans cette phase on a utilisé la méthode de segmentation de floue et nette basé sur LBP.
- convoluer l'image (avec le choix de PSF de convolution)
- segmenter l'image
- déconvoluer l'image (avec le choix de PSF de déconvolution)
- segmenter l'image
- fusionner les 2 segments

Après cette étude on peut dire que la déconvolution n'est pas efficace toujours à 100% en raison de l'absence de l'information préalable sur la fonction PSF qui a causé le floue néanmoins on peut encore dire qu'après plusieurs tests sur différents méthodes de déconvolution et manipulant les paramètres de la PSF en entrée on a constaté que les résultats change et ne donne pas toujours le même résultat.

**Conclusion**

A travers ce chapitre nous avons pu représenter notre travail sous forme d'une application qui sera utilisée par les futurs étudiants de notre département à travers le module TAI (Traitement des Images Numériques) pour avoir la main libre et facile sur le toolbox Debluring du Matlab en plus d'une application concrète sur le floue du type focus.

*Conclusion*

*Général*



### **Conclusion général**

Ce présent travail avait comme objectif de présenter un projet de fin d'étude à savoir la déconvolution automatique des images numériques.

Dans la partie utilitaire on a implémenté plusieurs méthodes d'amélioration des images (convolution, déconvolution) avec les quatre types des filtres (Winner, Lucy, Blind, Régularise) ainsi les différents types de fonction de PSF (moyen, gaussien, mouvement, focus).

Dans la partie floue de type focus on a implémenté toutes les méthodes d'utilitaire ainsi qu'une méthode de segmentation, ensuite une phase de fusion de l'image pour enfin obtenir le résultat inversé théoriquement attendue.

Ce projet de fin d'étude représente réellement une plateforme très modeste pour le traitement d'images numériques dans le domaine du deblurring tout en espérant que cette application sera enrichie par d'autres approches de traitement comme par exemple introduction de la logique floue afin de mieux estimer le type de floue et en conséquence choisir soigneusement le filtre approprié pour ce dit traitement ou bien nous pensons encore à une phase de post traitement qui permettra peut-être à estimer la qualité du deblurring afin de mener d'autre traitements possibles d'amélioration comme l'utilisation d'une deuxième application du même filtre ou d'un autre pour aboutir à une meilleurs qualité. Enfin nous espérons que ce modeste travail formera une plateforme préliminaire pour de futurs travaux scientifiques. Et que cette application facilitera la tâche d'enseignement de ce genre de domaine que ce soit du côté enseignant ou bien étudiants.

## Bibliographie

### Chapitre I

[1] : **K. Tanaka, Y. Nakamura, and K. Matsui.** « Embedding Secret Information into a Dithered »Multilevel Image ». In 1990 IEEE Military Communications Conférence, pages 216–220, 1990.

[2] : **Mémoire de fin d'études** Pour l'obtention du diplôme d'Ingénieur d'État en Informatique Systèmes Informatiques, ALLANE Ahmed Critères d'évaluation de la segmentation d'images- Etude comparative -en 2014. (Ibrahim, A. A. Mémoire de fin d'études pour l'obtention du diplôme d'Ingénieur des Techniques Agricoles.)

[4] : **J. P. Gastellu,** Acquisition et traitement d'image numérique, Université de Paul Sabatier, **France**, 2008.

### Chapitre II

[5] : Estimation rapide du paramètre de régularisation en déconvolution d'images. André Jalobeanu, Laure Blanc-Féraud, Josiane Zerubia Projet Ariana (projet commun CNRS/INRIA/UNSA) - INRIA Sophia Antipolis

[6] : J. Idier. Problèmes inverses en restauration de signaux et d'images. Mémoire d'habilitation à diriger les recherches, 2000.

[7] **Richardson, William Hadley** (1972). "Méthode itérative bayésienne de restauration d'image". *JOSA* . 62 (1): 55–59. doi : 10.1364 / JOSA.62.000055 .

[8] : **Lucy, LB** (1974). "Une technique itérative pour la rectification des distributions observées". *Journal astronomique*. 79 (6): 745–754

[9] : **Poisson DA ; Brinicombe AM; Pike ER;walker JG**(1995) « déconvolution aveugle moyen de l'algorithme de Richardson-Lucy (PDF),journal de la optical society of America

[10]: **Lam, Edmund Y.;** Goodman, Joseph W. (2000). "Approche statistique itérative de la déconvolution en aveugle". *Journal de la Optical Society of America A*. **17** (7): 1177–1184. Bibcode : 2000JOSAA...17.1177L . Doi : 10.1364 / JOSAA.17.001177 .

[11] : **Caron, James N .;** Namazi, Nader M .; Rollins, Chris J. (2002). "Restauration aveugle de données en aveugle en utilisant une fonction de filtre extraite". *Optique appliquée*. **41** (32): 6884–9. Bibcode : 2002ApOpt..41.6884C . Doi : 10.1364 / AO.41.006884 . PMID 12440543 .

### Chapitre III

[12] **Zhao, Minghua, et al,** "Restoration of single image based on kernel estimation with L1-regularization method," Ninth International Conference on Digital Image Processing, International Society for Optics and Photonics, vol. 10420, 2017.

[13]: **Efficient Self-Adaptive Image Deblurring Based on Model Parameter Optimization**" 2018 3rd IEEE International Conference on Image, Vision and Computing".

- [14] **A. Bennis and S. M. Riad**, “Filtering Capabilities and Convergence of the Van-Cittert Deconvolution Technique”, IEEE Transactions on Instrumentation and Measurement, vol. 41, no. 2, 1992
- [15] **L. Lang and Y. Xu**, “Adaptive Landweber method to deblur images”, IEEE Signal Processing Letters, vol. 10, no. 5, 2003
- [16] **Zohair Al-Ameen, Ghazali Sulong and Md. Gapar Md. Johar**,” A Comprehensive Study on Fast image Deblurring Techniques”, International Journal of Advanced Science and Technology Vol. 44, July, 2012
- [17] **A Survey on different Image Deblurring Techniques**,” International Journal of Computer Applications (0975 – 8887) Volume 116 – No. 13, April 2015.
- [18] **Neetin kumar, Dr. Manish shrivastva**,” Image deblurring using a neural network approaches”, IJEIT, vol 2, September 2012
- [19] **Ankit Gupta, Michel Cohen, Brian Curless**, “ Single image deblurring using motion density functions”, chapter- “computer vision Eccv 2010”.
- [20] **Aarpna Ashok, deepa**, “handling noise and outliers in single image Deblurring using L0 Sparsity”, vol 4, issue 7, july 2015.
- [19] **Jian-Feng Cai, Hui Ji, Chaoqiang Liu and Zuowei Shen**, “Blind motion deblurring from a single image using sparse approximation”, 978-1-4244-3991-1/09/\$25.00 ©2009 IEEE
- [20] **Lu Yuan, Jian Sun**,”Image deblurring using blurry/noisy image pairs” ACM.
- [21] **T. Ojala, M. Pietikainen, and D. Harwood**. A comparative study of “ texture measures with classification based on featured distributions. Pattern recognition, 29(1):51–59, 1996.
- [22] **T. Ojala and M. Pietikainen**. Unsupervised texture segmentation using “ feature distributions. Pattern Recognition, 32(3):477–486, 1999.
- [23] **T. Ahonen, A. Hadid, and M. Pietikainen**. Face description with local binary patterns: Application to face recognition. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 28(12):2037–2041, 2006.
- [24] **M. Heikkila and M. Pietikainen**. A texture-based method for modeling the background and detecting moving objects. IEEE transactions on pattern analysis and machine intelligence, 28(4):657–662, 2006.
- [25] **M. Pietikainen, T. Nurmela, T. Mäenpää, and M. Turtinen**. View-based “ recognition of real-world textures. Pattern Recognition, 37(2):313–323, 2004.
- [26] **T. Ojala, M. Pietikainen, and T. Maenpää**. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 24(7):971– 987, 2002.
- [27] **J. Shi, L. Xu, and J. Jia**. Blur detection dataset. <http://www.cse.cuhk.edu.hk/~leo/jia/projects/dblurdetect/dataset.html>, 2014.

- [28] **P. Viola and M. Jones.** Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, volume 1, pages I–511. IEEE, 2001.
- [29] **A. Levin, D. Lischinski, and Y. Weiss.** A closed-form solution to natural image matting. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 30(2):228–242, 2008.
- [30] **J. Shi, L. Xu, and J. Jia.** Discriminative blur detection features. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pages 2965–2972. IEEE, 2014.
- [31] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In Computer vision and pattern recognition, 2009. cvpr 2009. iee conference on, pages 1597–1604. IEEE, 2009.
- [32] Xin Yi, Mark Eramian,” LBP-based Segmentation of Defocus Blur “.
- [33] **S. Dai and Y. Wu.** Removing partial blur in a single image. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, pages 2544–2551. IEEE, 2009.

### ***Webographie***

- [3] : <http://www.kaddour.com/>
- [34] : [www.commentcamarche.net](http://www.commentcamarche.net) > ...>Développeurs /DBA> programmation >UML.
- [35] : [www.olecorre.com/4006/Diagramme-de-sequence](http://www.olecorre.com/4006/Diagramme-de-sequence).
- [36] :[support.objecteering.com/objectreeng6.1/help/fr/objectreeng.../class\\_diagrams.h tm](http://support.objecteering.com/objectreeng6.1/help/fr/objectreeng.../class_diagrams.htm).
- [37] : [www.cigref.fr/cigref\\_publications/.../cercle\\_Architecture\\_Enteprise\\_2008.pdf](http://www.cigref.fr/cigref_publications/.../cercle_Architecture_Enteprise_2008.pdf).

