

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE

*Université de Mohamed El-Bachir El-Ibrahimi - Bordj Bou Arreridj*

*Faculté des Sciences et de la technologie*

*Département d'Electronique*

# *Mémoire*

*Présenté pour obtenir*

**LE DIPLOME DE MASTER**

**FILIERE : ELECTRONIQUE**

**Spécialité : Electronique des Systèmes Embarqués (ESE)**

Par

- **M<sup>elle</sup> Souha ZOUAOUI**
- **M<sup>elle</sup> Feriel SOUICI**

*Intitulé*

**Développement d'un simulateur de restauration variationnelle d'images  
par filtrage et déconvolution**

*Soutenu le : 26 juin 2024*

*Devant le Jury composé de :*

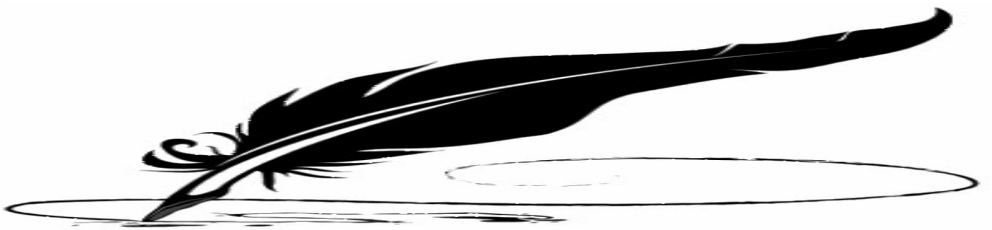
<i>Nom &amp; Prénom</i>	<i>Grade</i>	<i>Qualité</i>	<i>Etablissement</i>
<i>Dr. Mohammed Lamine TALBI</i>	<i>MCA</i>	<i>Président</i>	<i>Univ-BBA</i>
<i>Dr. Nacira DIFFELLAH</i>	<i>MCA</i>	<i>Encadreur</i>	<i>Univ-BBA</i>
<i>Dr. Rabah HAMDINI</i>	<i>MAB</i>	<i>Co-encadreur</i>	<i>Univ-BBA</i>
<i>M<sup>me</sup>. Sabrina MEGUELATI</i>	<i>MAA</i>	<i>Examineur</i>	<i>Univ-BBA</i>

*Année Universitaire 2023/2024*

Développement d'un simulateur de  
restauration variationnelle d'images par  
filtrage et déconvolution

Souha ZOUAOUI

Feriel SOUICI



26 juin 2024

# Table des matières

<b>Page de garde</b>	<b>1</b>
<b>Remerciements</b>	<b>III</b>
<b>Dédicace</b>	<b>V</b>
<b>Résumés</b>	<b>V</b>
<b>Liste des figures</b>	<b>VII</b>
<b>Liste des tableaux</b>	<b>IX</b>
<b>Liste des algorithmes</b>	<b>X</b>
<b>Liste des abbréviations</b>	<b>XI</b>
<b>Introduction</b>	<b>1</b>
<b>I Partie théorique</b>	<b>3</b>
<b>1 Modèles de dégradation d'images</b>	<b>5</b>
1.1 Introduction	6
1.2 Principales distributions de probabilités	6
1.2.1 Paramètres des distributions	6
1.2.2 Distribution de Gauss	7
1.2.3 Distribution uniforme	7
1.2.4 Distribution Gamma	8
1.2.5 Distribution de Rayleigh	8
1.2.6 Distribution de Poisson	9
1.2.7 Distribution de poivre et sel	10
1.2.8 Comparaison des distributions	10
1.3 Dégradation d'images	11
1.3.1 Notion de bruit	11
1.3.2 Causes du bruit dans le traitement des images	12
1.3.3 Types de bruit	12
1.4 Modèles de bruit	13

1.4.1	Bruit Gaussien . . . . .	13
1.4.2	Bruit poivre et sel . . . . .	13
1.4.3	Bruit du chatoiement (speckle) . . . . .	14
1.4.4	Flou . . . . .	14
1.4.5	Exemples de perturbation d'images . . . . .	14
1.5	Mesures de bruit . . . . .	18
1.5.1	Erreur Quadratique Moyenne . . . . .	18
1.5.2	Rapport signal sur bruit . . . . .	18
1.5.3	Rapport crête signal sur bruit . . . . .	19
1.5.4	Indice de similarité structurelle . . . . .	19
1.6	Conclusion . . . . .	19
<b>2</b>	<b>Restauration par filtrage et déconvolution</b>	<b>21</b>
2.1	Introduction . . . . .	21
2.2	Techniques classiques de restauration d'images . . . . .	22
2.2.1	Filtrage d'images . . . . .	22
2.2.2	Filtre gaussien . . . . .	22
2.2.3	Filtre de Wiener . . . . .	22
2.2.4	Filtre moyennneur . . . . .	22
2.2.5	Filtre médian . . . . .	23
2.2.6	Exemple de filtrage classique . . . . .	23
2.3	Restauration d'images par les méthodes Variationnelles . . . . .	24
2.3.1	Problème de minimisation $\ell_2$ . . . . .	24
2.3.2	Problème de minimisation $\ell_1$ . . . . .	26
2.4	Mesure de la qualité de la restauration . . . . .	28
2.5	Conclusion . . . . .	29
<b>II</b>	<b>Partie simulation</b>	<b>30</b>
<b>3</b>	<b>Simulateur sous appdesigner</b>	<b>32</b>
3.1	Introduction . . . . .	32
3.2	Notion générale sur l' App Designer . . . . .	32
3.3	Structure générale de l'application . . . . .	33
3.4	Résultats de simulation . . . . .	35
3.4.1	Images de test . . . . .	35
3.4.2	Tab 1 : Le <b>Design View</b> appliqué au débruitage d'image . . . . .	37
3.4.3	Tab 2 : Le <b>Design View</b> appliqué auau défloutage d'image . . . . .	38
3.4.4	Expérience 1 : Test de l'image IM003.tif . . . . .	39
3.4.5	Expérience 2 : Test de l'image IM052.tif . . . . .	42
3.4.6	Expérience 3 : Test de l'image cameraman.tif . . . . .	45
3.4.7	Expérience 4 : Test de l'image lena.png . . . . .	48
3.4.8	Expérience 5 : Test de l'image IM096.tif . . . . .	51
3.4.9	Expérience 6 : Test de l'image IM008.tif . . . . .	54
3.4.10	Interprétation et discussion des résultats . . . . .	57
3.5	Conclusion . . . . .	58

<i>TABLE DES MATIÈRES</i>	III
<b>Conclusion</b>	<b>59</b>
<b>Bibliographie</b>	<b>61</b>

# Remerciements

Nous tenons tout d'abord à exprimer notre profonde gratitude à nos encadrants de mémoire, **Dr.Nacira DIFFELLAH** et **Dr. Rabah HAMDINI** , pour leurs encadrement, leurs patience et leurs confiance tout au long de ce travail de recherche. Leurs précieux conseils, leurs expertises , et leurs soutien inébranlable ont été d'une aide inestimable et ont grandement contribué à l'aboutissement de notre projet.

Merci également aux membres du jury de notre soutenance de mémoire **Dr. Mohamed Lamine TALBI** et **Mme Sabrina MEGUELATI** pour avoir accepté de faire partie du jury.

Un merci spécial à mes camarades de classe et amis, [MAOUCHE,..], pour leurs encouragements, leurs échanges intellectuels stimulants et pour tous les moments partagés.

Nous ne saurions oublier nos familles, **ZOUAOUI** et **SOUICI**, pour leurs soutien indéfectible et pour avoir toujours cru en nous. Leurs encouragements ont été notre refuge et notre motivation durant tout notre parcours académique.

Enfin, Nous tenons à remercier toutes les personnes qui, de près ou de loin, ont contribué à la réalisation de ce mémoire. Que ce soit à travers des discussions enrichissantes, des conseils ou simplement par leur présence, leur contribution a été précieuse.

Ce mémoire est le fruit d'un travail collectif autant que personnel, et nous sommes profondément reconnaissantes envers tous ceux qui nous ont accompagnées.

# Dédicace

À Mes chers parents  
À Toute ma famille  
À Tous ceux qui me sont chers  
À Tous mes amis  
Je dédie le fruit de mes 22 ans  
d'études.

Souha ZOUAOUI



À Mes chers parents  
À Toute ma famille  
À Tous ceux qui me sont chers  
À Tous mes amis  
Je dédie le fruit de mes 22 ans  
d'études.

Feriel SOUCI

# Résumés

## ملخص

في هذا المشروع نقوم بتطوير محاكي رسومي لتصفية وفك تحليل الصور باستخدام تقنيات تباينية. تسمح التطبيق بتحميل، إضافة ضوضاء، ومعالجة مجموعة من الصور بالألوان أو الصور الرمادية. سنقوم بتنفيذ قياسات كمية لتقييم جودة الصور المعالجة تلقائيًا.

كلمات مفتاحية: معالجة الصور، محاكي رسومي، إزالة الضوضاء، فك التحليل، قياس الجودة، أساليب تباينية .

## Résumé

DANS ce projet nous nous intéressons à la création et développement d'un simulateur graphique pour l'utilisation de la procédure de filtrage et deconvolution des images par différentes techniques variationnelles. Les différentes zones de l'application permettent de charger, bruitez et appliquer le traitement sur une collection d'images en niveaux de gris ou couleurs. Des mesures quantitatives seront étudiées et implémentées pour prévoir automatiquement la qualité perceptive de l'image traitée.

*Mots clés : traitement d'images, simulateur graphique, débruitage, deconvolution, mesure de qualité, méthodes variationnelles...*

## Abstract

In this project, we are developing a graphical simulator for filtering and deconvolution of images using variational techniques. The application allows loading, adding noise to, and processing a collection of grayscale or color images. We will implement quantitative measures to automatically assess the perceptual quality of the processed images.

*Keywords : image processing, graphical simulator, denoising, deconvolution, quality measurement, variational methods...*

# Liste des figures

1.1	Distribution de Gauss . . . . .	7
1.2	Distribution uniforme . . . . .	8
1.3	Distribution de Gamma . . . . .	8
1.4	Distribution de Rayleigh . . . . .	9
1.5	Distribution de Poisson . . . . .	9
1.6	Distribution poivre et sel . . . . .	10
1.7	Bruitage de l'image IM052.tif . . . . .	15
1.8	Bruitage de l'image IM003.tif . . . . .	15
1.9	Influence de la variance sur le bruitage de l'image ' <i>IM052.tif</i> ' . . . . .	16
1.10	Influence de la variance sur le bruitage de l'image ' <i>IM003.tif</i> ' . . . . .	17
2.1	Restauration de l'image IM003.tif . . . . .	23
2.2	Restauration de l'image IM052.tif . . . . .	23
3.1	Etapes de création du simulateur . . . . .	33
3.2	L'interface <b>Design View</b> du simulateur . . . . .	34
3.3	Images de test . . . . .	36
3.4	Tab 1 : <b>Design View</b> du débruitage de l'image cameraman.tif . . . . .	37
3.5	Tab 2 : <b>Design View</b> du défloutage de l'image cameraman.tif . . . . .	38
3.6	Débruitage de l'image ' <i>IM003.tif</i> ' pour différentes valeurs de la variance . . . . .	40
3.7	Défloutage de l'image ' <i>IM0003.tif</i> ' pour différentes valeurs de la densité . . . . .	41
3.8	Débruitage de l'image ' <i>IM052.tif</i> ' pour différentes valeurs de la densité . . . . .	43
3.9	Défloutage de l'image ' <i>IM052.tif</i> ' pour différentes valeurs de la variance . . . . .	44
3.10	Débruitage de l'image ' <i>cameraman.tif</i> ' pour différentes valeurs de la densité . . . . .	46
3.11	Défloutage de l'image ' <i>cameraman.tif</i> ' pour différentes valeurs de la variance . . . . .	47
3.12	Débruitage de l'image ' <i>lena.pgg</i> ' pour différentes valeurs de la densité . . . . .	49
3.13	Défloutage de l'image ' <i>lena.png</i> ' pour différentes valeurs de la variance . . . . .	50
3.14	Débruitage de l'image ' <i>IM096.tif</i> ' pour différentes valeurs de la densité . . . . .	52

3.15 Défloutage de l'image ' <i>IM096.tif</i> ' pour différentes valeurs de la variance . . . . .	53
3.16 Débruitage de l'image ' <i>IM008.tif</i> ' pour différentes valeurs de la densité . . . . .	55
3.17 Défloutage de l'image ' <i>IM008.tif</i> ' pour différentes valeurs de la variance . . . . .	56

# Liste des tableaux

1.1	Caractéristiques des différentes distributions . . . . .	11
3.1	Métriques de l'image 'IM003.tif' . . . . .	39
3.2	Métriques de l'image 'IM052.tif' . . . . .	42
3.3	Métriques de l'image 'cameraman.tif' . . . . .	45
3.4	Métriques de l'image 'lena.png' . . . . .	48
3.5	Métriques de l'image 'IM096.tif' . . . . .	51
3.6	Métriques de l'image 'IM008.tif' . . . . .	54

# Liste des Algorithmes

1	Problème de minimisation $\ell_2$ . . . . .	26
2	Problème de minimisation $\ell_1$ . . . . .	28

# Liste des abréviations

<i>MSE</i>	Mean Square Error
<i>SNR</i>	Signal to Noise Ratio
<i>PSNR</i>	Peak Signal to Noise Ratio
<i>SSIM</i>	Structural Similarity Index
<i>SAP</i>	Bruit sel et poivre
<i>PDF</i>	Fonction de densité de probabilité
<i>EQE</i>	Erreur Quadratique Moyenne
<i>VT</i>	Variation Totale
<i>u</i>	Image originale
<i>f</i>	Image bruitée ou observée
$\hat{u}$	Image restaurée ou estimée
$\mu$	Espérance ou moyenne du bruit
$\sigma$	Variance du bruit
$\sigma^2$	Ecart type du bruit
$E(Z)$	Espérance de la variable aléatoire $Z$
$V(Z)$	Variance de la variable aléatoire $Z$
$p(z)$	le PDF d'une loi gaussienne
$f(z)$	Le PDF d'une loi uniforme
$p(k)$	Le PDF de la loi de Poisson
$\lambda$	Paramètre de la loi de Poisson
$K$	Masque
$\eta$	Bruit ajouté à l'image
$L(.)$	Lagrangien augmenté
$\rho_r$	Paramètre de régularisation
$y, z, r$	Multiplicateurs de Lagrange
*	Opérateur de convolution



# *Introduction*

*'Science sans conscience n'est que ruine de l'âme'*

# Introduction générale

Les images numériques peuvent être dégradées par du bruit lors du processus d'acquisition des données, des interférences de phénomènes naturels, de la transmission et de la compression. Le processus de débruitage des images consiste à trouver l'image vraie et inconnue  $u$  à partir d'une image bruitée  $f$ .

Dans la littérature, différentes méthodes ont été proposées pour supprimer le bruit et restaurer l'image réelle telles que le filtrage médian itératif, le filtre médian pondéré, le filtre médian adaptatif, la transformée en ondelettes, la diffusion anisotrope et le filtre de Variation Totale ( $VT$ ). Dans ce mémoire, nous nous intéressons au dernier type de filtrage, minimiser  $VT$  engendre la solution à être constante par morceaux, ce qui est bien adapté pour débruiter l'image tout en préservant les contours. Nous tentons de récupérer une image qui a été dégradée en utilisant une connaissance a priori du phénomène de dégradation. Nous traitons le problème de débruitage et deconvolution des images via deux formulations variationnelles, nous effectuons une minimisation de la variation totale avec deux types d'attache aux données norme- $\ell_1$  et norme- $\ell_2$ .

Trois chapitres composent ce mémoire :

Dans un premier chapitre nommé '**Modèles de dégradation d'images**'<sup>1</sup>, les principales distributions de probabilités (Gauss, Gamma, Rayleigh, poivre et sel), les modèles et les types bruits (additif, multiplicatif, convolutif) et les mesures de bruit (l'erreur quadratique moyenne, le rapport signal sur bruit, le rapport crête signal sur bruit et l'indice de similarité structurelle) sont présentés.

Le deuxième chapitre intitulé '**Restauration par filtrage et déconvolution**'<sup>2</sup> est consacré aux méthodes de restauration d'images. Nous explorerons certaines techniques classiques (le filtre gaussien, le filtre moyenneur, le filtre de Wiener et le filtre médian) et deux approches de minimisation variationnelles dites norme- $\ell_1$  et norme- $\ell_2$ .

Quant au troisième chapitre qui s'intitule '**Simulateur sous appdesigner Matlab**'<sup>3</sup>, nous allons passer comme son nom l'indique au développement d'un simulateur de restauration variationnelle d'images par filtrage et déconvolution. Nous allons nous intéresser aux images numériques en niveaux de gris et couleur<sup>1</sup>.

Pour terminer une conclusion générale 3.5 et références bibliographiques 3.5 sont fournis. La performance d'une méthode de restauration peut être calculée à l'aide d'un des indicateurs suivants : le  $MSE$ , le  $SNR$ , le  $PSNR$ ) et le  $SSIM$ .

---

1. <https://lcondat.github.io/imagebase.html>

**Première partie**

**Partie théorique**



# Chapitre 1

*Une heure de lecture est le souverain remède contre les dégoûts de la vie.*

# Chapitre 1

## Modèles de dégradation d'images

### Sommaire

---

<b>1.1 Introduction</b>	<b>6</b>
<b>1.2 Principales distributions de probabilités</b>	<b>6</b>
1.2.1 Paramètres des distributions	6
1.2.2 Distribution de Gauss	7
1.2.3 Distribution uniforme	7
1.2.4 Distribution Gamma	8
1.2.5 Distribution de Rayleigh	8
1.2.6 Distribution de Poisson	9
1.2.7 Distribution de poivre et sel	10
1.2.8 Comparaison des distributions	10
<b>1.3 Dégradation d'images</b>	<b>11</b>
1.3.1 Notion de bruit	11
1.3.2 Causes du bruit dans le traitement des images	12
1.3.3 Types de bruit	12
<b>1.4 Modèles de bruit</b>	<b>13</b>
1.4.1 Bruit Gaussien	13
1.4.2 Bruit poivre et sel	13
1.4.3 Bruit du chatoiement (speckle)	14
1.4.4 Flou	14
1.4.5 Exemples de perturbation d'images	14
<b>1.5 Mesures de bruit</b>	<b>18</b>
1.5.1 Erreur Quadratique Moyenne	18
1.5.2 Rapport signal sur bruit	18
1.5.3 Rapport crête signal sur bruit	19
1.5.4 Indice de similarité structurelle	19
<b>1.6 Conclusion</b>	<b>19</b>

---

## 1.1 Introduction

Ce chapitre couvrira de manière approfondie les distributions de probabilités, essentielles en mathématiques et en statistiques. Nous explorons les lois de probabilité, en soulignant des concepts clés comme l'espérance, la variance et l'écart-type, qui décrivent la répartition des valeurs d'une variable aléatoire. Diverses distributions seront examinées en détail, notamment la gaussienne, l'uniforme, la distribution de Poisson, de Rayleigh et gamma, avec une explication claire de leurs propriétés. Nous discuterons après les différents types de bruits, tels que le bruit additif, convolutif et multiplicatif, ainsi que leurs sources. Enfin, nous aborderons les mesures de qualité comme le *MSE*, le *SNR*, le *PSNR* et le *SSIM*, en soulignant leurs importances respectives.

## 1.2 Principales distributions de probabilités

Cette section présente quatre distributions continues : la distribution gaussienne, la distribution uniforme, la distribution gamma et la distribution de Rayleigh. Ensuite, elle aborde la distribution discrète de Poisson. Chaque distribution est caractérisée par sa fonction de densité de probabilité [1], [2].

### 1.2.1 Paramètres des distributions

Deux paramètres fondamentaux permettant de décrire une distribution donnée sont : la moyenne et la variance ou l'écart-type :

1. **Espérance ou moyenne** : L'espérance d'une variable aléatoire est la valeur attendue de cette variable. Elle représente la moyenne des valeurs possibles pondérées par leurs probabilités. Mathématiquement, l'espérance d'une variable aléatoire discrète  $Z$  est définie comme :

$$E(Z) = \sum_i z_i \cdot P(Z = z_i) \quad (1.1)$$

2. **Variance** : La variance mesure la dispersion des valeurs autour de l'espérance. Elle quantifie la variabilité des résultats possibles. Mathématiquement, la variance d'une variable aléatoire  $Z$  est donnée par l'équation :

$$V(Z) = E((Z - E(Z))^2) \quad (1.2)$$

3. **L'écart-type** L'écart-type est la mesure de dispersion la plus couramment utilisée. Il est défini comme la racine carrée de la variance :

$$\sigma = \sqrt{V(Z)} \quad (1.3)$$

## 1.2.2 Distribution de Gauss

Également connue sous le nom de loi normale, cette loi est souvent utilisée comme modèle pour les distributions d'erreurs de mesure autour d'une valeur vraie. On dit qu'une variable aléatoire  $Z$  suit une loi Gaussienne si elle admet une densité de probabilités spécifique de la forme :

$$P_{(\mu, \sigma^2)}(z) = \frac{1}{\sqrt{\sigma^2 2\pi}} e^{-\frac{(z-\mu)^2}{2\sigma^2}}, \quad \forall z \in \mathbb{R} \quad (1.4)$$

Le paramètre  $\mu$  est appelé la moyenne ou espérance,  $\sigma$  est l'écart-type et  $\sigma^2$  est la variance.

La figure 1.1 présente l'allure de la densité de probabilité d'une gaussienne :

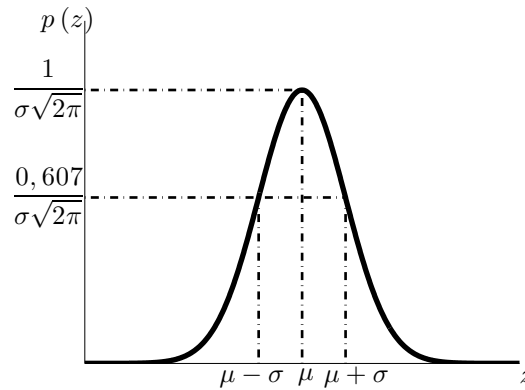


FIGURE 1.1 – Distribution de Gauss

De la figure 1.1, on remarque que :

- La distribution de Gauss a une forme en cloche symétrique. La symétrie indique que la moyenne arithmétique et la médiane de la distribution sont égales.
- La moyenne est le point central de la distribution.
- L'écart-type mesure la dispersion des données autour de la moyenne.

## 1.2.3 Distribution uniforme

Une variable aléatoire suit une loi uniforme si toutes les valeurs possibles ont la même probabilités d'occurrence. La fonction de densité de probabilité d'une loi uniforme sur l'intervalle  $[a, b]$ , est donnée par la formule (1.5) :

$$f(z) = \begin{cases} \frac{1}{b-a} & \text{pour } a \leq z \leq b. \\ 0 & \text{ailleurs} \end{cases} \quad (1.5)$$

La figure 1.2 illustre la représentation graphique de la distribution uniforme :

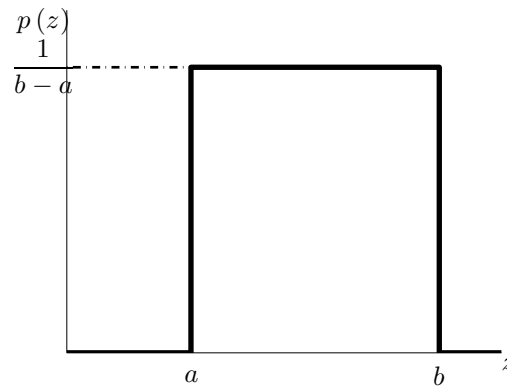


FIGURE 1.2 – Distribution uniforme

### 1.2.4 Distribution Gamma

La fonction de densité de probabilité de la loi gamma est donnée par la formule suivante :

$$p(z) = \frac{1}{a^b (b-1)!} z^{b-1} e^{-\frac{z}{a}} \quad (1.6)$$

Les paramètres  $a$  et  $b$  sont des entiers positifs tels que  $a > b$ , '!' indique la factorielle.

La distribution de la loi Gamma a la forme d'une exponentielle comme le montre la figure 1.3 :

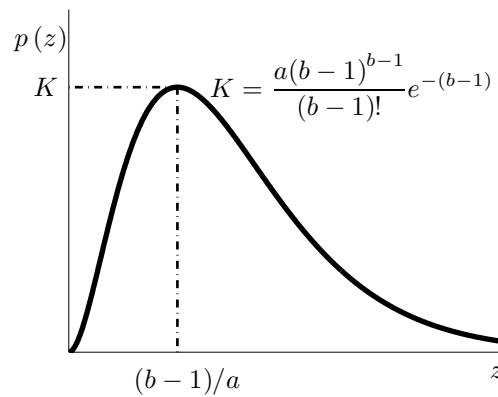


FIGURE 1.3 – Distribution de Gamma

### 1.2.5 Distribution de Rayleigh

La fonction de densité de probabilité de la loi de Rayleigh est donnée par la formule suivante :

$$p(z) = \begin{cases} \frac{2}{b} (z-a) e^{-(z-a)^2/b} & z \geq a \\ 0 & z < a \end{cases} \quad (1.7)$$

La figure 1.4 montre l'allure de la distribution de Rayleigh (1.7) :

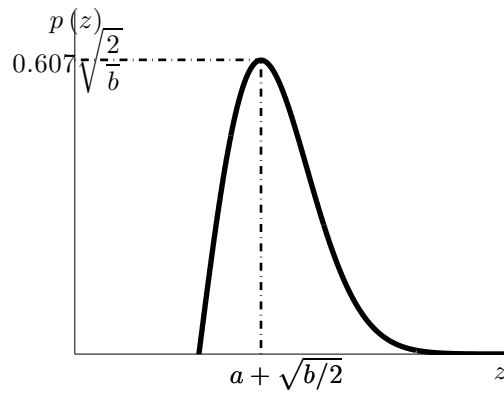


FIGURE 1.4 – Distribution de Rayleigh

### 1.2.6 Distribution de Poisson

La fonction de probabilité de la loi de Poisson est donnée par l'équation 1.8 :

$$p(k) = p(z = k) = \frac{\lambda^k}{k!} e^{-\lambda}, k = 0, 1, 2, \dots \quad (1.8)$$

- Le paramètre  $\lambda$  de la distribution de Poisson représente le taux moyen d'occurrence des événements dans un intervalle donné. la distribution de Poisson est centrée autour de cette valeur moyenne.
- Plus le paramètre  $\lambda$  est grand, moins la distribution est dispersée, et vice versa.

La représentation graphique de la loi de Poisson est illustrée par la figure 1.5 :

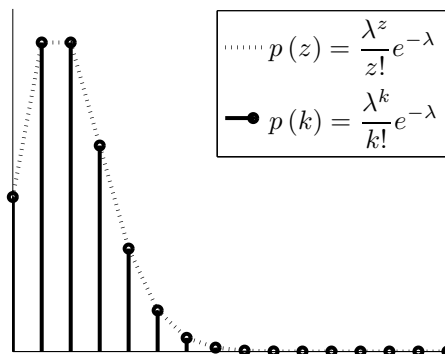


FIGURE 1.5 – Distribution de Poisson

### 1.2.7 Distribution de poivre et sel

La fonction de probabilité de la loi poivre et sel est donnée par l'équation 1.9 :

$$P(z) = \begin{cases} P_S & \text{Si } z = 2^k - 1 \\ P_p & \text{Si } z = 0 \\ 1 - (P_s + P_p) & \text{Si } z = V \end{cases} \quad (1.9)$$

Où  $V$  est une valeur entière dans la plage  $0 < V < 2^k - 1$ .

La représentation graphique de la loi de poivre et sel est montrée par la figure 1.6 :

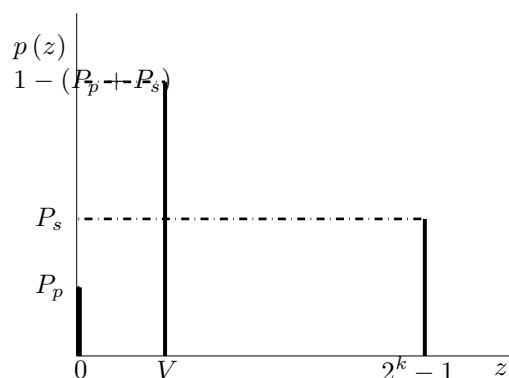


FIGURE 1.6 – Distribution poivre et sel

### 1.2.8 Comparaison des distributions

Le tableau 1.1 récapitule les densités de probabilité des différentes lois ainsi que leurs valeurs moyennes et leurs écarts-types :

TABLEAU 1.1 – Caractéristiques des différentes distributions

Loi	PDF $f(z)$	$\mu$	$\sigma^2$
Normale $Z \sim N(\mu, \sigma^2)$	$p(z) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(z-\mu)^2}{2\sigma^2}}$	$\mu$	$\sigma^2$
Uniforme $Z \sim U(a, b)$	$U(a, b) = \begin{cases} \frac{1}{b-a} & \text{pour } a \leq z \leq b. \\ 0 & \text{ailleurs} \end{cases}$	$\frac{a+b}{2}$	$\frac{(b-a)^2}{12}$
Gamma $Z \sim G(a, b)$	$p(z) = \frac{1}{a^b(b-1)!} z^{b-1} e^{-\frac{z}{a}}$	$\frac{b}{a}$	$\frac{b}{a^2}$
Rayleigh $Z \sim R(\sigma^2)$	$p(z) = \begin{cases} \frac{2}{b} (z-a) e^{-(z-a)^2/b} & z \geq a \\ 0 & z < a \end{cases}$	$a + \sqrt{\pi b/4}$	$\frac{b(4-\pi)}{4}$
Poisson $K \sim P(\lambda)$	$p(z = k) = \frac{\lambda^k}{k!} e^{-\lambda}$	$\lambda$	$\lambda$
Poivre et sel $Z \sim P(z)$	$P(z) = \begin{cases} P_S & \text{si } z = 2^K - 1 \\ P_P & \text{si } z = 0 \\ 1 - (P_S + P_P) & \text{si } z = V \end{cases}$	$\mu_{PS}$	$\sigma_{PS}$

avec

$$\mu_{PS} = (0)P_P + K(1 - P_S - P_P) + (2^K - 1)P_S$$

$$\sigma_{PS} = (0 - \mu)^2 P_P + (K - \mu)^2 (1 - P_S - P_P) + (2^K - 1)^2 P_S$$

$Z$  est la variable aléatoire.

Nous avons inclus 0 comme une valeur explicite dans les deux équations pour indiquer que la valeur du bruit de poivre est supposée être zéro[1].

## 1.3 Dégradation d'images

### 1.3.1 Notion de bruit

En général, le bruit peut être défini comme tout signal indésirable qui peut être synthétisé avec l'image originale. C'est un phénomène presque inévitable dans le traitement des images, qu'il soit introduit lors de l'acquisition initiale ou pendant la transmission. Plusieurs facteurs peuvent contribuer à son apparition. La détermination de la quantité de bruit dans une image dépend généralement du nombre de pixels altérés[3],[4],[5].

### 1.3.2 Causes du bruit dans le traitement des images

Les principales sources de bruit dans les images numériques comprennent[3],[4],[5] :

- Conditions environnementales affectant le capteur d'imagerie lors de l'acquisition, notamment la lumière ambiante et la température.
- Origine de l'image : si elle est numérisée à partir d'une photographie sur film, les particules de film, la détérioration du film ou même le scanner peuvent introduire du bruit.
- Mécanisme d'acquisition des données pour les images prises directement en format numérique.
- Niveaux de lumière insuffisants ou variations de température affectant le capteur.
- Transmission électronique des données d'image, susceptible d'introduire des interférences et du bruit.
- Interférences sur le chemin de transmission, pouvant altérer l'image pendant son transfert.
- La présence de particules de poussière sur l'écran du scanner, une source commune de bruit dans les images scannées.

### 1.3.3 Types de bruit

Les quatre principaux types de bruit que nous voyons habituellement dans les images sont les suivants<sup>1</sup>[3],[5],[4] :

- Bruit additif : Causé par des variations aléatoires de luminosité ou d'informations de couleur sur l'image. Il s'agit du type de bruit le plus courant observé dans les images.
- Bruit multiplicatif : Cela se produit lorsque la valeur du bruit est multipliée par la valeur du pixel d'origine, ce qui entraîne souvent une mauvaise qualité d'image autour des parties les plus lumineuses de l'image. Il s'agit du type de bruit le plus difficile à supprimer en raison des variations importantes de la valeur des pixels.
- Bruit impulsif : Causé par des changements soudains dans la valeur des pixels qui sont visibles sous forme de pixels aléatoires en noir et blanc, perçus comme de fortes perturbations dans l'image. On l'appelle également « bruit de sel et de poivre ». Cela résulte de défauts de caméra, d'erreurs de transmission ou de rayons cosmiques.
- Bruit convolutif : Ce type de bruit peut être causé par plusieurs facteurs, notamment les imperfections optiques dans les systèmes de capture d'image, les vibrations ou les mouvements de la caméra pendant l'acquisition de l'image, les conditions atmosphériques telles que la turbulence de l'air, ou même des défauts dans le processus de numérisation ou de traitement de l'image.

---

1. <https://www.unite.ai/fr/what-is-noise-in-image-processing-a-primer/>

## 1.4 Modèles de bruit

En traitement d'images, les modèles de bruit sont des représentations mathématiques des divers types de perturbations susceptibles d'altérer les images. Nous citons certains exemples de modèles de bruit fréquemment utilisés : le bruit impulsif (bruit sel et poivre), le bruit d'amplificateur (bruit gaussien), le bruit de quantification (bruit uniforme) et le bruit multiplicatif (bruit de speckle)[4],[3] :

### 1.4.1 Bruit Gaussien

Le bruit gaussien est réparti de manière homogène sur le signal. Cela signifie que chaque pixel de l'image bruitée est la somme de la valeur réelle du pixel et d'une valeur de bruit gaussien aléatoire. Le bruit est indépendant de l'intensité de la valeur du pixel à chaque point. Les principales sources de bruit gaussien dans les images numériques apparaissent lors de l'acquisition, par exemple le bruit du capteur causé par une mauvaise illumination, une température élevée ou une transmission .

La formule (1.10) donne l'équation d'une image dégradée par un une image bruit additif Gaussien :

$$f = Ku + \eta \quad (1.10)$$

Où  $f$  est l'image dégradée (observée),  $K$  est un masque et  $\eta$  sont des variables aléatoires indépendantes et identiquement distribuées suivant une loi Gaussienne centrée de variance  $\sigma^2$  selon l'équation (1.4).

### 1.4.2 Bruit poivre et sel

Le bruit poivre et sel est un type de bruit impulsif. Il est causé par des pics d'intensité. Ce type de bruit se produit en raison d'erreurs lors de la transmission des données. Il apparaît dans l'image en raison de changements brusques et soudains du signal de l'image. Les pixels bruités peuvent prendre uniquement les valeurs maximales et minimales de la plage dynamique. Dans une image de 8 bits, la valeur typique pour le bruit poivre est de 0 et pour le bruit sel elle est de 255. Ce bruit est généralement causé par des dysfonctionnements des éléments de pixel dans les capteurs de l'appareil photo, des emplacements de mémoire défectueux ou des erreurs de synchronisation dans le processus de numérisation.

Une image corrompue par un bruit poivre et sel selon la distribution(1.9) est donnée par la relation (1.11) :

$$f = u + \eta \quad (1.11)$$

$u$ , est l'image originale,  $f$ , est l'image bruitée,  $\eta$ , est le bruit poivre et sel. Le bruit sel et poivre (SAP) existe généralement dans les images naturelles, et les pixels contaminés par le bruit SAP prennent la valeur maximale ou minimale (1.12) :

$$\eta = \begin{pmatrix} 0 & \text{Bruit poivre} \\ 255 & \text{Bruit sel} \end{pmatrix} \quad (1.12)$$

### 1.4.3 Bruit du chatolement (speckle)

Le bruit de speckle, également connu sous le nom de bruit de granulation, est un type de bruit multiplicatif. Il se produit dans la plupart des systèmes d'imagerie cohérente tels que les lasers, l'acoustique et les images radar à ouverture synthétique (SAR). Ce bruit est causé par des interférences aléatoires entre les signaux cohérents.

Une image corrompue par un bruit multiplicatif peut être modélisée comme suit :

$$f = u \cdot \eta \quad (1.13)$$

Le bruit de speckle suit une distribution gamma, ce qui lui confère ses caractéristiques particulières

En général, le bruit multiplicatif suit les distributions gamma ou Rayleigh selon les lois (1.6) ou (1.7).

### 1.4.4 Flou

Le bruit de type flou, également connu sous le nom de flou gaussien, est un type de bruit qui affecte les images en les rendant moins nettes. Le flou peut provenir de différentes sources, telles que la turbulence atmosphérique, la diffraction, la mise au point optique, les tremblements de l'appareil photo, et bien d'autres. Il peut donner à l'image une apparence floue ou estompée.

Une dégradation floue de l'image est modélisée par l'expression suivante :

$$f = u * \eta \quad (1.14)$$

### 1.4.5 Exemples de perturbation d'images

Les figures 1.7 et 1.8 montrent des exemples de perturbations d'une image (bruit et flou) :

- Les images originales en niveau de gris et couleur sont montrées respectivement par les figures 1.8a et 1.7a.
- Le bruit de Gauss affecte tous les pixels de l'image comme le montre les figures 1.7b et 1.8b.
- Les figures 1.7d et 1.8d montrent que le bruit impulsionnel n'affecte que certains pixels de l'image.
- La présence des granules est causée par le bruit multiplicatif 1.7c, 1.8c
- Les figures 1.7e et 1.8e montrent la dégradation des images par le flou.

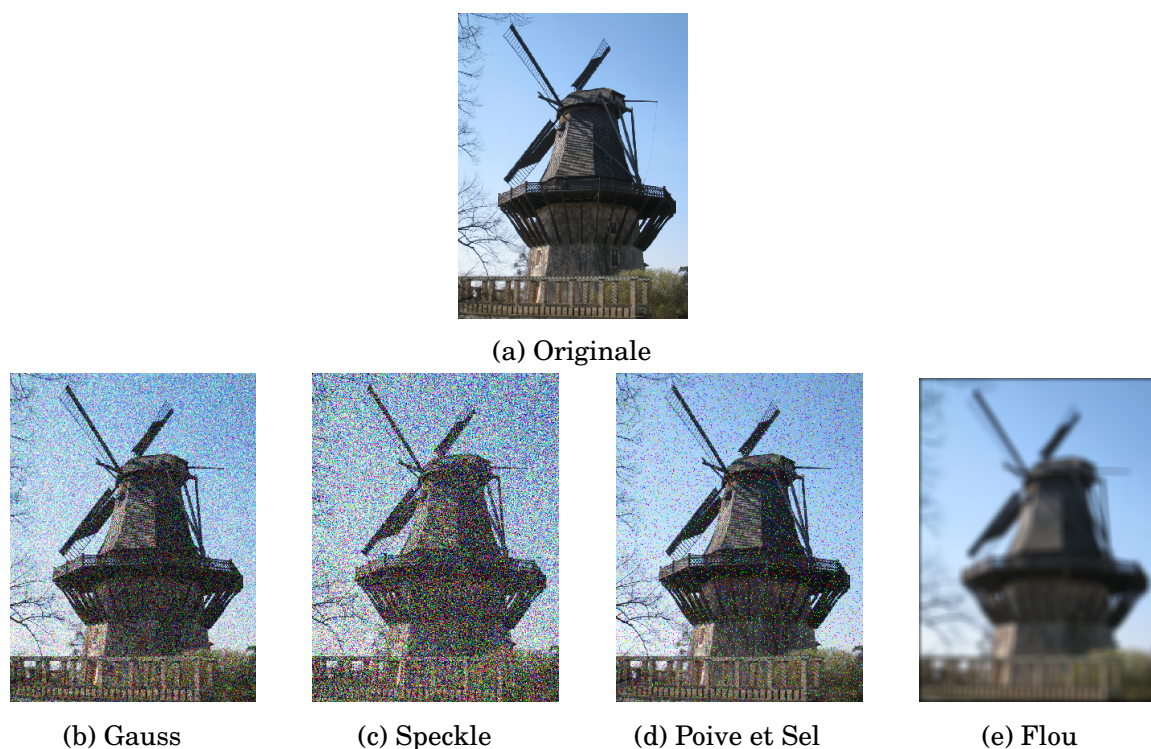


FIGURE 1.7 – Bruitage de l'image IM052.tif

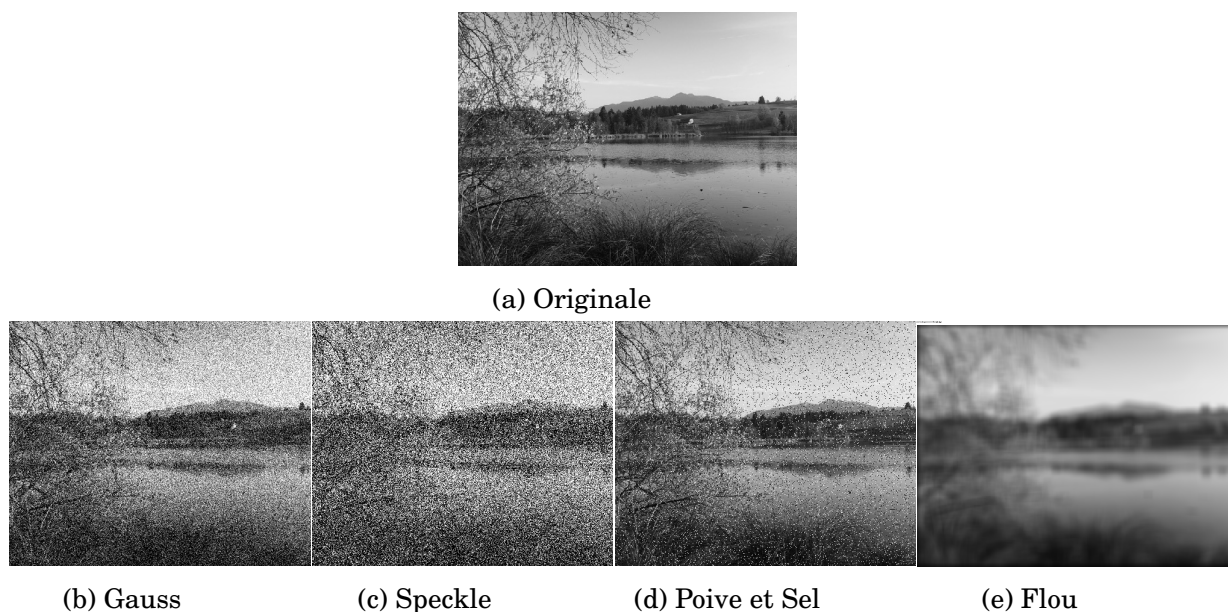


FIGURE 1.8 – Bruitage de l'image IM003.tif

Les figures 1.9 et 1.10 montrent l'influence de la variance et la densité sur les images 'IM052.tif' et 'IM003.tif'.

- Plus la variance ou la densité sont élevées, plus l'image est dégradée.
- Une faible densité de bruit entraîne des images plus nettes comme le montre les figures de l'image couleur 1.9b , 1.9e, 1.9h, 1.9k ou l'image en niveaux de



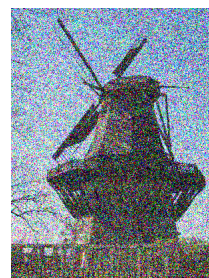
(a) Originale



(b) Gauss  $\sigma = 0.02$



(c) Gauss  $\sigma = 0.06$



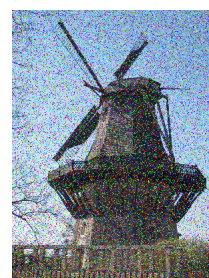
(d) Gauss  $\sigma = 0.2$



(e) Poivre et sel  $d = 0.02$



(f) Poivre et sel  $d = 0.06$



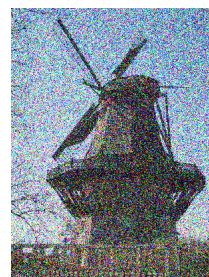
(g) Poivre et sel  $d = 0.2$



(h) Speckle  $\sigma = 0.02$



(i) Speckle  $\sigma = 0.06$



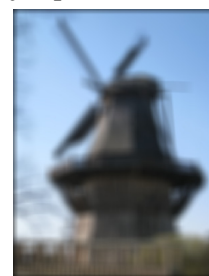
(j) Speckle  $\sigma = 0.2$



(k) Flou  $\sigma = 10$



(l) Flou  $\sigma = 20$



(m) Flou  $\sigma = 30$

FIGURE 1.9 – Influence de la variance sur le bruitage de l'image 'IM052.tif'

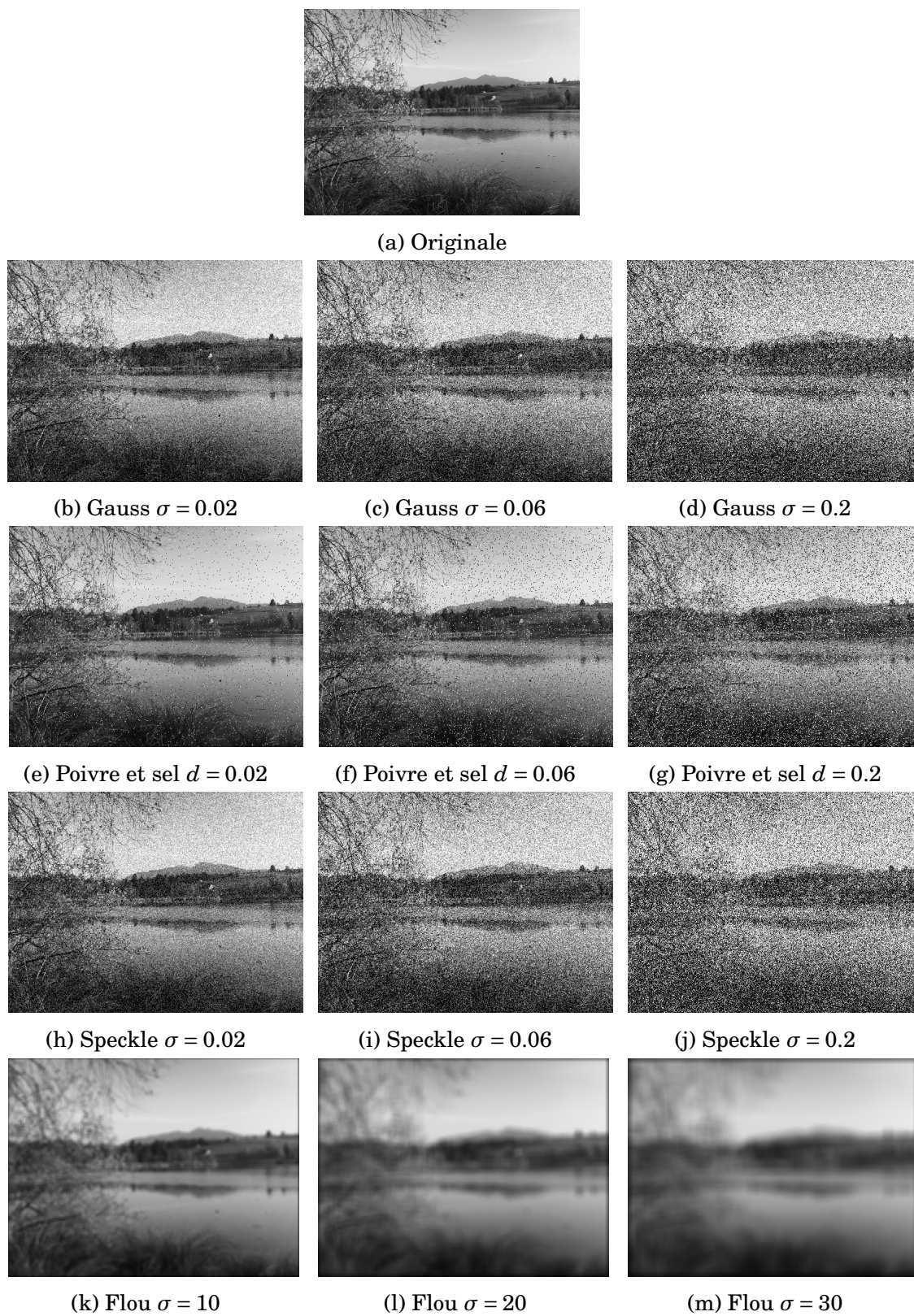


FIGURE 1.10 – Influence de la variance sur le bruitage de l'image 'IM003.tif'

gris 1.10b, 1.10e, 1.10h et 1.10k.

- La densité (ou variance) du bruit peut avoir un impact significatif sur la qualité des images. Un bruit excessif peut entraîner une dégradation de la netteté et de la clarté de l'image, rendant les détails difficiles à distinguer (voir les figures 1.9d, 1.9g, 1.9j, 1.9m pour l'image couleur et les figures 1.10d, 1.10g, 1.10j, 1.10m pour l'image en niveaux de gris.)

## 1.5 Mesures de bruit

Lorsqu'une image subit une dégradation, il est important de mesurer le niveau de détérioration qu'elle a subi. Il existe plusieurs métriques permettant de fournir des valeurs numériques qui représentent le degré de gêne provoqué par la dégradation de l'image. Pour évaluer le bruit et quantifier le niveau de bruit, nous utilisons quatre mesures : l'erreur quadratique moyenne, le rapport signal sur bruit, le rapport crête signal sur bruit et l'indice de similarité structurelle [6],[3].

### 1.5.1 Erreur Quadratique Moyenne

Pour calculer l'Erreur Quadratique Moyenne  $EQM$  (En anglais Mean Square Error  $MSE$ ), nous mesurons l'erreur quadratique moyenne entre les pixels de l'image originale et ceux de l'image dégradée .

**Définition 1** ( $MSE$ ) *L'erreur quadratique moyenne de la dégradation  $f$  de l'image  $u$  est définie par :*

$$MSE(u, f) = \frac{1}{MN} \sum_i^M \sum_j^N (u(i, j) - f(u, j))^2 \quad (1.15)$$

*La différence  $u(i, j) - f(i, j)$  est considérée comme un bruit.*

*$u(i, j)$  désigne l'image originale et  $f(i, j)$  son image bruitée,  $M$  et  $N$  sont les dimensions de l'image  $u(i, j)$ .*

### 1.5.2 Rapport signal sur bruit

Le Rapport Signal sur Bruit ( $RSB$ ) (En anglais Signal Noise to Ratio également connu sous l'acronyme  $SNR$ ), est une mesure très répandue. Il représente le rapport entre l'énergie du signal et celle du bruit.

**Définition 2** ( $SNR$ ) *Le rapport signal à bruit est défini par :*

$$SNR(u, f) = \frac{\sum_i^M \sum_j^N u(i, j)^2}{\sum_i^M \sum_j^N (u(i, j) - f(i, j))^2} \quad (1.16)$$

*Le  $SNR$  est exprimé en décibels  $dB$ .*

### 1.5.3 Rapport crête signal sur bruit

Le Rapport Crête Signal sur Bruit (En anglais Peak Signal to Noise Ratio *PSNR*) est basé sur le calcul direct de l'écart pixel par pixel de chaque image. Le *PSNR* est exprimé en décibels (*dB*) :

**Définition 3 (*PSNR*)** *Le rapport crête signal sur bruit est défini par :*

$$PSNR(u, f) = 10 \log \left( \frac{L_d^2}{MSE} \right) \quad (1.17)$$

*Le *PSNR* est exprimé en décibels *dB*.  $L_d^2$  est la plage dynamique des valeurs de pixels.*

Si l'image d'entrée a un type de données entier non signé sur 8 bits, alors  $L_d = 255$ .

### 1.5.4 Indice de similarité structurelle

L'indice de similarité structurelle ( En anglais Structural SIMilarités *SSIM*) est une mesure qui évalue la similarité structurelle entre deux images. Le *SSIM* prend en compte à la fois la similarité des luminances, des contrastes et des structures de l'image.

**Définition 4 (*SSIM*)** *L'indice de similarité structurelle entre l'image originale et bruitée est défini par :*

$$SSIM(u, f) = \frac{(2\mu_u\mu_f + C_1)(2\sigma_{uf} + C_2)}{(\mu_u^2 + \mu_f^2 + C_1)(\sigma_u^2 + \sigma_f^2 + C_2)} \quad (1.18)$$

*où  $\mu_u$  et  $\mu_f$  sont les moyennes des intensités de pixels dans les images  $u$  et  $f$ ,  $\sigma_u$  et  $\sigma_f$  sont les écarts-types des intensités de pixels dans les images  $u$  et  $f$ ,  $\sigma_{uf}$  est la covariance entre les intensités de pixels dans les images  $u$  et  $f$  et  $C_1, C_2$  sont des constantes pour stabiliser la division avec un dénominateur faible.*

## 1.6 Conclusion

Dans ce chapitre, nous avons introduit les distributions de probabilités, les sources de bruits, les différents types de bruit qui peuvent exister et les mesures de qualité entre l'image originale et dégradée.



*'Pour faire de grandes choses, il ne faut pas être un si grand génie ; il ne faut pas être au-dessus des hommes, il faut être avec eux.'*

# Chapitre 2

## Restauration par filtrage et déconvolution

### Sommaire

---

<b>2.1</b>	<b>Introduction</b>	<b>21</b>
<b>2.2</b>	<b>Techniques classiques de restauration d'images</b>	<b>22</b>
2.2.1	Filtrage d'images	22
2.2.2	Filtre gaussien	22
2.2.3	Filtre de Wiener	22
2.2.4	Filtre moyeneur	22
2.2.5	Filtre médian	23
2.2.6	Exemple de filtrage classique	23
<b>2.3</b>	<b>Restauration d'images par les méthodes Variationnelles</b>	<b>24</b>
2.3.1	Problème de minimisation $\ell_2$	24
2.3.2	Problème de minimisation $\ell_1$	26
<b>2.4</b>	<b>Mesure de la qualité de la restauration</b>	<b>28</b>
<b>2.5</b>	<b>Conclusion</b>	<b>29</b>

---

### 2.1 Introduction

La restauration d'image, première étape cruciale dans les systèmes de vision par ordinateur, vise à récupérer l'image originale à partir de l'image dégradée, en tenant compte des caractéristiques des dégradations et la nature du bruit.

Dans ce chapitre, nous nous intéresserons aux approches de restauration d'images par filtrage ou déconvolution, qui permettent de supprimer les artefacts et le bruit pour rapprocher l'image restaurée à l'image d'origine tout en conservant la même résolution spatiale.

## 2.2 Techniques classiques de restauration d'images

### 2.2.1 Filtrage d'images

Le filtrage vise à réduire, supprimer ou améliorer certains éléments, permettant de mettre en évidence les caractéristiques importantes de l'image tout en éliminant les détails inutiles ou le bruit. Cela rend l'image plus nette et plus claire.

Il existe différents types de filtres, certains étant linéaires, exprimés sous forme de convolution, et d'autres non linéaires,[3],[5],[4].

### 2.2.2 Filtre gaussien

Un filtre gaussien[3],[5] est un filtre linéaire spécial avec des caractéristiques mathématiques spécifiques qui convolue une image avec une fonction gaussienne. Cela permet d'atténuer la haute fréquence et de lisser l'image.

### 2.2.3 Filtre de Wiener

Le filtre Wiener [3],[5] minimise l'erreur quadratique moyenne  $e^2 = E[(u - \hat{u})^2]$ . Nous pouvons utiliser le filtre de Wiener si nous possédons des connaissances préliminaires sur le bruit et l'image d'origine. La forme du filtre est obtenue à partir de l'équation suivante :

$$\hat{U} = \frac{|K|^2}{K \cdot |H|^2 + \frac{S_\eta}{S_u}} F \quad (2.1)$$

Le filtre de Wiener est donné par l'équation (2.2) :

$$W = \frac{|H|^2}{H \cdot |H|^2 + \frac{S_\eta}{S_u}} \quad (2.2)$$

$S_u = |U|^2$  et  $S_\eta = E[|N|^2]$  sont les densités spectrales de puissance de l'image et de bruit respectivement.  $R = \frac{S_\eta}{S_u}$  est le rapport bruit sur signal qui est l'inverse du rapport signal sur bruit.  $|H|^2 = H^*H$ ,  $H^*$  est le conjugué de  $H$  et  $\hat{U}$ ,  $F$  et  $H$  sont les transformées de Fourier de  $\hat{u}$ ,  $f$  et  $h$  respectivement.

Le filtre de Wiener est très utile pour conserver les détails dans les zones avec des bords ou d'autres caractéristiques de contraste élevé.

### 2.2.4 Filtre moyenneur

Le filtre moyenneur [3],[5] remplace la valeur centrale de la fenêtre par la moyenne des valeurs des pixels voisins, y compris lui-même, cela permet de lisser l'image en remplaçant les pixels qui ne correspondent pas à leur environnement.

Le filtre moyenneur est un filtre linéaire passe bas qui permet de supprimer les fréquences élevées, qui sont liées au bruit, il présente un désavantage, il supprimera également les fréquences élevées correspondant aux détails de l'image. L'image sera moins bruitée mais plus floue.

### 2.2.5 Filtre médian

Le filtre médian [3],[5] est un filtre non-linéaire. Comme son nom l'indique, il sélectionne le pixel de la fenêtre d'analyse ayant la valeur médiane.

Le filtre médian a la propriété de laisser les transitions monotones entre régions inchangées.

### 2.2.6 Exemple de filtrage classique

Les figures 2.1 et 2.2 représentent la restauration des images '*IM003.tif*' et '*IM052.tif*' corrompues par un bruit Poivre et sel, par diverses techniques classiques de restauration d'images.

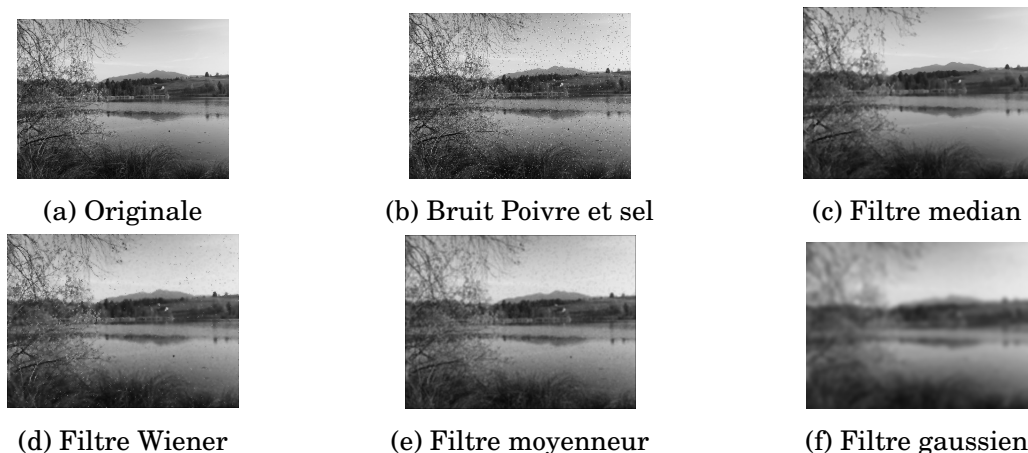


FIGURE 2.1 – Restauration de l'image IM003.tif



FIGURE 2.2 – Restauration de l'image IM052.tif

- Les figures 2.1a et 2.2a et montrent les images d’origine en couleurs et en niveaux de niveaux gris respectivement.
- Le bruit poivre et sel est un type de bruit aléatoire qui se traduit l’apparition de pixels noirs et blancs. Les iimage bruitées par ce type de bruit sont illustrées par les figures 2.1b et 2.2b.
- Le filtre médian est utilisé pour réduire le bruit en remplaçant chaque pixel par la médiane des pixels voisins, cela peut améliorer la netteté de l’image. Les images obtenues après l’application du filtre médian sont montrées par les figures 2.1c et 2.2c.
- Les iimages représentées par les figures 2.1d et 2.2d et indiquent que Le filtre de Wiener est utilisé pour réduire le bruit tout en préservant les détails de l’image.
- Le filtre moyeneur calcule la moyenne des pixels voisins pour chaque pixel, ce qui peut adoucir l’image comme le montre clairement les figures 2.1e et 2.2e et.
- Le filtre gaussien floute l’image en utilisant une distribution gaussienne pour pondérer les pixels voisins, ceci se traduit par les figures 2.1f et 2.2f.

Chaque filtre a un effet différent sur l’apparence de l’image. Par exemple, le filtre médian réduit le bruit, tandis que le filtre gaussien adoucit l’image.

## 2.3 Restauration d’images par les méthodes Variationnelles

La méthode de variation totale a été initialement proposée par Rudin, Osher et Fatemi en vision par ordinateur [7]. Elle sert de critère de régularisation pour résoudre des problèmes inverses, et est particulièrement efficace pour régulariser les images sans altérer les contours des objets.

Les méthodes variationnelles  $\ell_1$  et  $\ell_2$  se basent sur la minimisation d’une fonction objectif qui combine une mesure de fidélité aux données et une régularisation pour encourager la préservation des contours et des détails importants de l’image tout en réduisant le bruit[7],[8],[9].

### 2.3.1 Problème de minimisation $\ell_2$

La méthode  $\ell_2$ , également connue sous le nom de norme- $\ell_2$  ou régularisation de Tikhonov est décrite par le modèle suivant :

$$\min_u \left\{ \frac{\lambda}{2} \|Ku - f\|_2^2 + \|u\|_1 \right\} \quad (2.3)$$

$u \in \mathfrak{R}^N$  est l’image réelle et  $f \in \mathfrak{R}^N$  est l’image contaminée par un bruit,  $\lambda > 0$  est le paramètre de régularisation,  $\|\cdot\|_2$  et  $\|\cdot\|_1$  désignent respectivement la norme 2 et 1. Le premier terme de l’équation (2.3) lié au bruit,  $J_1(u, f) = \|Ku - f\|_2^2$  est appelé le terme d’ajustement des données (la fidélité), le second terme  $J_2(u) = \|u\|_1$  est appelé le terme de régularisation (ou de pénalité), qui réalisent en fait la réduction

du bruit. Notons que les deux termes sont convexes.

Pour trouver la solution de ce problème nous procédons comme suit :

- D'abord, nous transformons le problème (2.3) sous forme d'un problème de minimisation de la fonction objectif en respectant la contrainte :

$$\begin{aligned} \min_{u,g} \frac{\lambda}{2} |Ku - f|^2 + |g|_1 \\ \text{Sous contraintes} \\ g = Du \end{aligned} \quad (2.4)$$

- Nous transformons par la suite le problème 2.4 en un problème de minimisation sans contrainte en utilisant le Lagrangien augmenté :

$$L(u, g, y) = \frac{\lambda}{2} |Ku - f|^2 + |g|_1 - y^T (g - Du) + \frac{\rho_r}{2} |g - Du|^2 \quad (2.5)$$

$L(u, g, y)$  est le lagrangien augmenté,  $\rho_r$  est un paramètre de régularisation et  $y$  est le multiplicateur de Lagrange associé à la contrainte  $g = Du$ .

Il est à noter que le point de selle du Lagrangien augmenté 2.5 est aussi la solution du problème initial 2.3.

- Après, nous passons à la recherche de  $u$ ,  $g$  et  $y$  de  $L(u, g, y)$  :
  - Résolution du sous-problème  $u$

$$u = \underset{u}{\operatorname{argmin}} L(u, g, y) \quad (2.6)$$

$$u = \underset{u}{\operatorname{argmin}} \left\{ \frac{\lambda}{2} |Ku - f|^2 - y^T (g - Du) + \frac{\rho_r}{2} |g - Du|^2 \right\} \quad (2.7)$$

- Résolution du sous-problème  $g$

$$g = \underset{g}{\operatorname{argmin}} L(u, g, y) \quad (2.8)$$

$$g = \underset{g}{\operatorname{argmin}} \left\{ |g|_1 - y^T (g - Du) + \frac{\rho_r}{2} |g - Du|^2 \right\} \quad (2.9)$$

- Résolution du sous-problème  $y$

$$y = \underset{y}{\operatorname{argmin}} L(u, g, y) \quad (2.10)$$

$$y = \underset{y}{\operatorname{argmin}} \left\{ -y^T (g - Du) \right\} \quad (2.11)$$

L'algorithme 1 permet de trouver l'image estimée en appliquant la méthode variationnelle  $\ell_2$  :

---

**Algorithme 1** Problème de minimisation  $\ell_2$

---

- 1: Entrée : image bruitée  $f$  et la matrice  $K$
- 2: Paramètre d'entrée :  $\lambda$ ,
- 3: Fixer le paramètre  $\rho_r = 2$
- 4: Initialiser  $u_0 = f$ ,  $g_0 = Du_0$ ,  $y = 0$  et  $k = 0$
- 5: **while**  $\frac{\|u_{k+1} - u_k\|}{\|u_k\|} \geq tol$  **do**
- 6:     Résoudre le sous-problème  $u$  (2.7)

$$u_{k+1} = \operatorname{argmin}_u \left\{ \frac{\lambda}{2} |Ku - f|^2 - y_k^T (g_k - Du) + \frac{\rho_r}{2} |g_k - Du|^2 \right\} \quad (2.12)$$

- 7:     Résoudre le sous-problème  $g$  (2.9)

$$g_{k+1} = \operatorname{argmin}_g \left\{ |g|_1 - y_k^T (g - Du_{k+1}) + \frac{\rho_r}{2} |g - Du_{k+1}|^2 \right\} \quad (2.13)$$

- 8:     Mettre à jour le multiplicateur de Lagrange  $y$  (2.11) en appliquant la descente du gradient :

$$y_{k+1} = y_k - \rho_r (g_{k+1} - Du_{k+1}) \quad (2.14)$$

- 9: **end while**
- 

### 2.3.2 Problème de minimisation $\ell_1$

Le problème de régularisation  $\ell_1$ , connu aussi sous le nom de norme- $\ell_1$  est donné par l'équation (2.15) :

$$\min_u \{ \lambda \|Ku - f\|_1 + \|u\|_1 \} \quad (2.15)$$

Pour trouver la solution de ce problème nous suivons le plan suivant :

- Transformons d'abord le problème (2.15) sous forme d'un problème de minimisation de la fonction objectif en respectant les deux contraintes :

$$\begin{aligned} & \min_{u,r,g} \{ \lambda |r|_1 + |g|_1 \} \\ & \text{Sous contraintes} \\ & r = Ku - f \\ & g = Du \end{aligned} \quad (2.16)$$

- Nous transformons par la suite le problème sous contraintes 2.16 en un problème de minimisation sans contrainte en utilisant le Lagrangien augmenté :

$$\begin{aligned} L(u, r, g, y, z) = & \lambda |r|_1 + |g|_1 \\ & - z^T (r - Ku + f) + \frac{\rho_0}{2} \|r - Ku + f\|^2 \\ & - y^T (g - Du) + \frac{\rho_r}{2} |g - Du|^2 \end{aligned} \quad (2.17)$$

$L(u, r, g, y, z)$  est le lagrangien augmenté,  $\rho_r$  est un paramètre de régularisation,  $y$  et  $z$  sont les multiplicateurs de Lagrange associés aux contraintes  $g = Du$  et  $r = Ku - f$  respectivement.

Le point de selle du Lagrangien augmenté 2.17 est aussi la solution du problème initial 2.15.

— Recherche de  $u, g, r, y$  et  $z$  de  $L(u, r, g, y, z)$  :

— Résolution du sous-problème  $u$

$$u = \underset{u}{\operatorname{argmin}} L(u, r, g, y, z) \quad (2.18)$$

$$u = \underset{u}{\operatorname{argmin}} \left\{ z^T Ku + \frac{\rho_0}{2} \|r - Ku + f\|^2 + y^T Du + \frac{\rho_r}{2} |g - Du|^2 \right\} \quad (2.19)$$

— Résolution du sous-problème  $g$

$$g = \underset{g}{\operatorname{argmin}} L(u, r, g, y, z) \quad (2.20)$$

$$g = \underset{g}{\operatorname{argmin}} \left\{ |g|_1 - y^T g + \frac{\rho_r}{2} |g - Du|^2 \right\} \quad (2.21)$$

Le sous-problème  $g$  de  $\ell_1$  (2.21) est le même que celui  $\ell_2$  (2.9). Nous avons donc la même solution.

— Résolution du sous-problème  $r$

$$r = \underset{r}{\operatorname{argmin}} L(u, r, g, y, z) \quad (2.22)$$

$$r = \underset{r}{\operatorname{argmin}} \left\{ \lambda |r|_1 - z^T r + \frac{\rho_0}{2} \|r - Ku + f\|^2 \right\} \quad (2.23)$$

— Résolution du sous-problème  $y$

$$y = \underset{y}{\operatorname{argmin}} L(u, r, g, y, z) \quad (2.24)$$

$$y = \underset{y}{\operatorname{argmin}} \left\{ y^T (g - Du) \right\} \quad (2.25)$$

— Résolution du sous-problème  $z$

$$z = \underset{z}{\operatorname{argmin}} L(u, r, g, y, z) \quad (2.26)$$

$$z = \underset{z}{\operatorname{argmin}} \left\{ z^T (r - Ku + f) \right\} \quad (2.27)$$

L'algorithme 2 permet de trouver l'image estimée en appliquant la méthode variationnelle  $\ell_1$  :

---

**Algorithme 2** Problème de minimisation  $\ell_1$

---

- 1: Entrée : image bruitée  $f$  et la matrice  $K$
- 2: Paramètre d'entrée :  $\lambda$
- 3: Fixer les paramètres  $\rho_r = 2$  et  $\rho_o = 100$
- 4: Initialiser  $u_0 = f$ ,  $g_0 = Du_0$ ,  $y_0 = 0$ ,  $r_0 = Ku_0 - f$ ,  $z_0 = 0$  et  $k = 0$
- 5: **while**  $\frac{\|u_{k+1} - u_k\|}{\|u_k\|} \geq tol$  **do**
- 6:     Résoudre le sous-problème  $u$  (2.19)

$$u_{k+1} = \underset{u}{\operatorname{arg\,min}} \left\{ z_k^T Ku + \frac{\rho_o}{2} \|r_k - Ku + f_k\|^2 + y_k^T Du + \frac{\rho_r}{2} |g_k - Du|^2 \right\} \quad (2.28)$$

- 7:     Résoudre le sous-problème  $g$  (2.21)

$$g_{k+1} = \underset{g}{\operatorname{arg\,min}} \left\{ |g|_1 - y_k^T g + \frac{\rho_r}{2} |g - Du_{k+1}|^2 \right\} \quad (2.29)$$

- 8:     Résoudre le sous-problème  $r$  (2.23)

$$r_{k+1} = \underset{r}{\operatorname{arg\,min}} \left\{ \lambda |r|_1 - z^T r + \frac{\rho_o}{2} \|r - Ku_{k+1} + f_{k+1}\|^2 \right\} \quad (2.30)$$

- 9:     Mettre à jour le multiplicateur de Lagrange  $y$  (2.25)

$$y_{k+1} = y_k - \rho_r (g - Du) \quad (2.31)$$

- 10:     Mettre à jour le multiplicateur de Lagrange  $z$  (2.27)

$$z_{k+1} = z_k - \rho_o (r_{k+1} - Ku_{k+1} + f) \quad (2.32)$$

- 11: **end while**
- 

## 2.4 Mesure de la qualité de la restauration

Les mesures de la qualité [6],[2] de la restauration sont basées sur la mesure de différence entre une image de référence de taille  $M \times N$  et une image estimée notées respectivement  $u(i, j)$  et  $\hat{u}(i, j)$ . Le  $SNR$  est utilisé pour estimer la qualité d'une image  $\hat{u}(i, j)$  par rapport à une image de référence  $u(i, j)$ . Le  $PSNR$  donne une approximation de la fidélité de l'image reconstruite par rapport à l'originale [2]. Ces métriques sont définies respectivement par les formules (2.33), (2.34), (2.35) et le SSIM (2.36) :

$$SNR(u, \hat{u}) = \frac{\sum_i \sum_j \hat{u}(i, j)^2}{\sum_i \sum_j (u(i, j) - \hat{u}(i, j))^2} \quad (2.33)$$

$$PSNR(u, \hat{u}) = 10 \log \left( \frac{L_d^2}{MSE} \right) \quad (2.34)$$

avec

$$MSE(u, \hat{u}) = \frac{1}{M \times N} \sum_i^M \sum_j^N (u(i, j) - \hat{u}(u, j))^2 \quad (2.35)$$

$$SSIM(u, \hat{u}) = \frac{(2\mu_u \mu_{\hat{u}} + C_1)(2\sigma_{u\hat{u}} + C_2)}{(\mu_u^2 + \mu_{\hat{u}}^2 + C_1)(\sigma_u^2 + \sigma_{\hat{u}}^2 + C_2)} \quad (2.36)$$

Une faible valeur du  $MSE$  implique une grande valeur du  $PSNR$ . Plus le  $SNR$  et le  $PSNR$  sont élevés, l'image est considérée de meilleure qualité. L'indice  $SSIM$  [6] résultant est une valeur comprise entre  $-1$  et  $1$ , la valeur  $1$  n'est accessible que dans le cas d'une meilleure qualité d'image ou de deux images identiques.

## 2.5 Conclusion

Nous avons présenté dans ce chapitre les techniques de réduction du bruit qui contribuent à améliorer la qualité de l'image telles que : le filtrage moyenneur, le filtrage médian, le filtrage gaussien, le filtrage de Wiener et le filtrage variationnel par la norme- $\ell_1$  et la norme- $\ell_1$ . Nous avons aussi décrit les différentes métriques utilisées pour évaluer la qualité de l'image filtrée : le  $MSE$ , le  $PNR$  et  $PSNR$  et le  $SSIM$ .

**Deuxième partie**  
**Partie simulation**



*'Il faut étudier beaucoup pour en savoir un peu.'*

# Chapitre 3

## Simulateur sous appdesigner

### Sommaire

---

<b>3.1</b>	<b>Introduction . . . . .</b>	<b>32</b>
<b>3.2</b>	<b>Notion générale sur l' App Designer . . . . .</b>	<b>32</b>
<b>3.3</b>	<b>Structure générale de l'application . . . . .</b>	<b>33</b>
<b>3.4</b>	<b>Résultats de simulation . . . . .</b>	<b>35</b>
3.4.1	Images de test . . . . .	35
3.4.2	Tab 1 : Le <b>Design View</b> appliqué au débruitage d'image . .	37
3.4.3	Tab 2 : Le <b>Design View</b> appliqué au défloutage d'image .	38
3.4.4	Expérience 1 : Test de l'image IM003.tif . . . . .	39
3.4.5	Expérience 2 : Test de l'image IM052.tif . . . . .	42
3.4.6	Expérience 3 : Test de l'image cameraman.tif . . . . .	45
3.4.7	Expérience 4 : Test de l'image lena.png . . . . .	48
3.4.8	Expérience 5 : Test de l'image IM096.tif . . . . .	51
3.4.9	Expérience 6 : Test de l'image IM008.tif . . . . .	54
3.4.10	Interprétation et discussion des résultats . . . . .	57
<b>3.5</b>	<b>Conclusion . . . . .</b>	<b>58</b>

---

### 3.1 Introduction

Ce chapitre vise la création d'un simulateur de filtrage et deconvolution d'images sous Matlab App Designer. Nous allons essayer de développer une application interactive pour l'analyse et la manipulation d'images à l'aide de deux méthodes de restauration variationnelles  $\ell_1$  et  $\ell_2$ .

### 3.2 Notion générale sur l' App Designer

L'App Designer de Matlab est un outil de développement qui permet de créer des applications autonomes en utilisant du code Matlab. Il offre la possibilité de créer des applications de qualité professionnelle en utilisant différents composants

et fonctions de rappel. L'App Designer comprend deux vues principales : la vue de conception *Design View* où l'on peut ajouter différents composants à l'application, et la vue de code *Code View* où l'on modifie les fonctions de rappel. Lorsqu'un objet est ajouté à la mise en page de l'application, le code correspondant est automatiquement ajouté dans la vue de code. L'interaction avec les différents objets se fait à travers des fonctions de rappel qui sont déclenchées lorsque l'utilisateur interagit avec ces objets. Chaque composant de l'App Designer possède des fonctions de rappel spécifiques [10].

### 3.3 Structure générale de l'application

La figure 3.1 montre les étapes de création du simulateur :

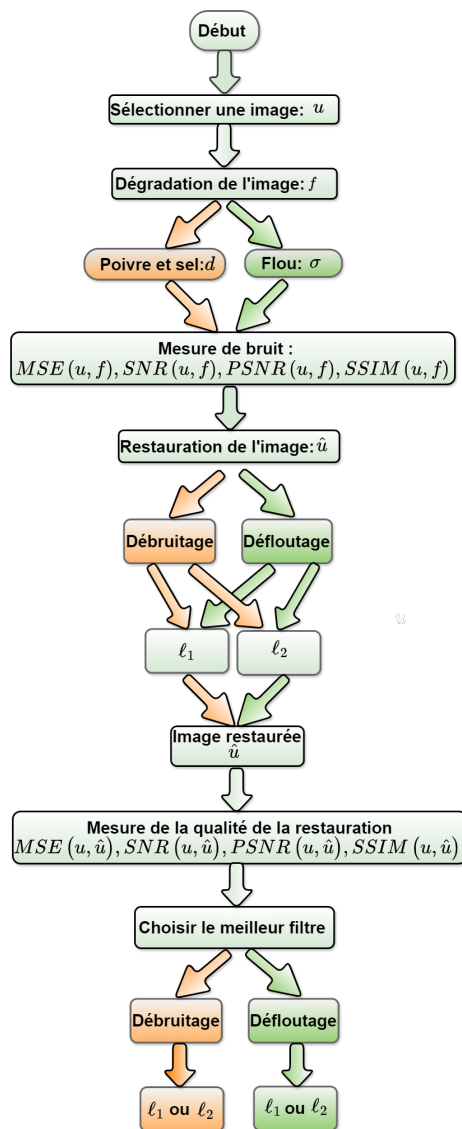


FIGURE 3.1 – Étapes de création du simulateur

L'interface **Design View** du simulateur est illustrée par la figure 3.2 :

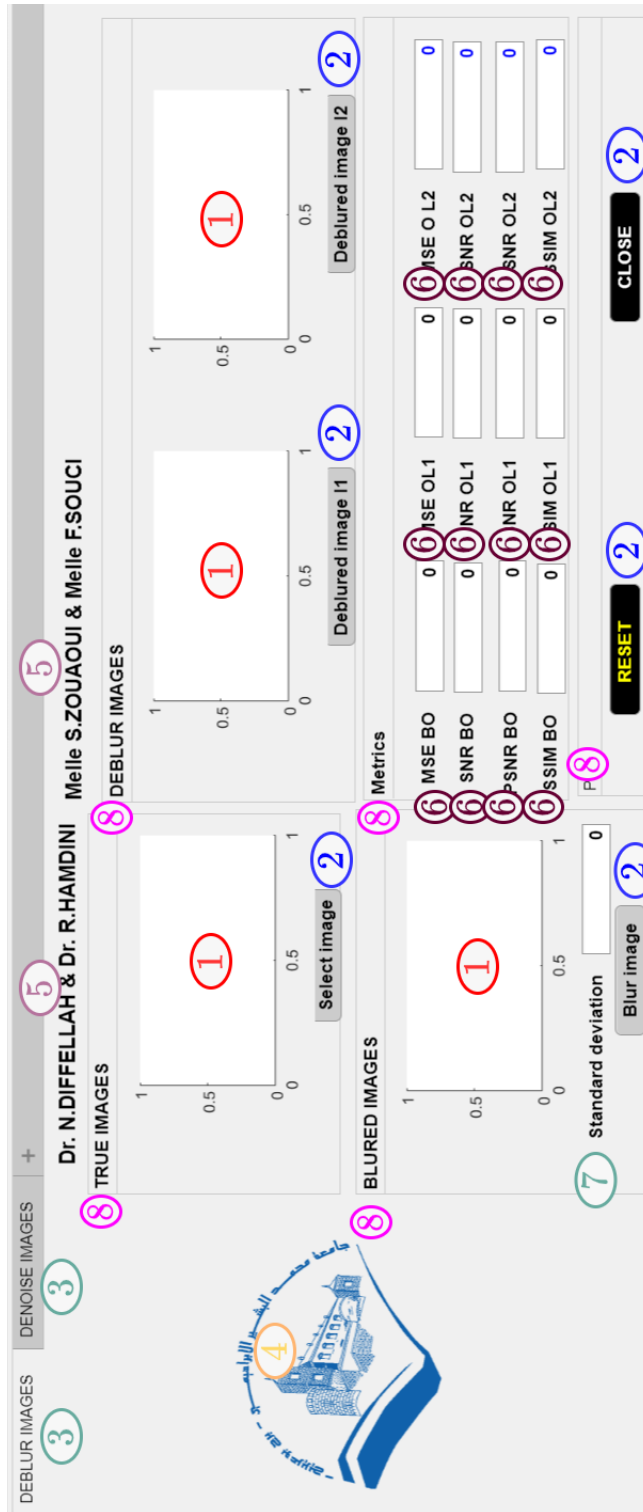


FIGURE 3.2 – L'interface **Design View** du simulateur

- ① Axes : utilisé pour afficher les graphiques.
- ② Bouton : Utilisé pour déclencher des actions lorsqu'elles sont cliquées.
- ③ Tab : utilisé pour organiser l'interface en plusieurs onglets. c'est à dire créer plusieurs onglets dans l'App Designer et organiser les composants dans ces onglets pour différentes sections de l'application).
- ④ Image : pour afficher des images. Pour notre cas, nous avons afficher le logo de notre université Mohamed El Bachie El Ibrahim.
- ⑤ Label : pour afficher du texte statique.
- ⑥ Edit Field Numeric : utilisé pour afficher des valeurs numériques.
- ⑦ Edit Field Numeric : Pour entrer des valeurs numériques.
- ⑧ Panel : Les panneaux sont utilisés pour organiser les composants de l'interface en groupes logiques.

## 3.4 Résultats de simulation

Nous avons implémenté les deux méthodes variationnelles basées sur les normes  $\ell_1$  et  $\ell_2$ . L'utilisateur peut sélectionner une image dégradée et appliquer l'une des méthodes de restauration pour améliorer la qualité de l'image. Les mesures de qualité telles que Le *MSE*, le *SNR*, le *PSNR* et la fonction de similarité structurale *SSIM* sont également calculées pour évaluer l'efficacité de chaque méthode de restauration.

### 3.4.1 Images de test

Pour faire le test, on considère six images de dimensions différentes, la figures 3.3 montre les images choisies :

- Trois images en niveaux de gris : 3.3a, 3.3c et 3.3e de dimensions  $720 \times 540$ ,  $256 \times 256$  et  $540 \times 720$  respectivement.
- Trois images en couleurs 3.3b, 3.3d et 3.3f de dimensions  $720 \times 540$ ,  $512 \times 512$  et  $540 \times 720$  respectivement.

Les images *IM003.tif* 3.3a, *IM008.tif* 3.3e, *IM096.tif* 3.3b et *IM052.tif* 3.3f sont tirées de la base de donnée :

<https://lcondat.github.io/imagebase.html>



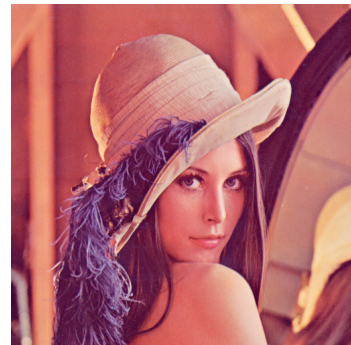
(a) '*IM003.tif*'  
720 × 540



(b) '*IM0096.tif*'  
720 × 540



(c) '*cameraman.tif*'  
256 × 256



(d) '*Lena.png*'  
512 × 512



(e) '*IM0008.tif*'  
540 × 720



(f) '*IM052.tif*'  
540 × 720

FIGURE 3.3 – Images de test

### 3.4.2 Tab 1 : Le Design View appliqué au débruitage d'image

La figure 3.4 montre le **Design View** appliqué au débruitage d'image pour le cas de l'image 'cameraman.tif' :

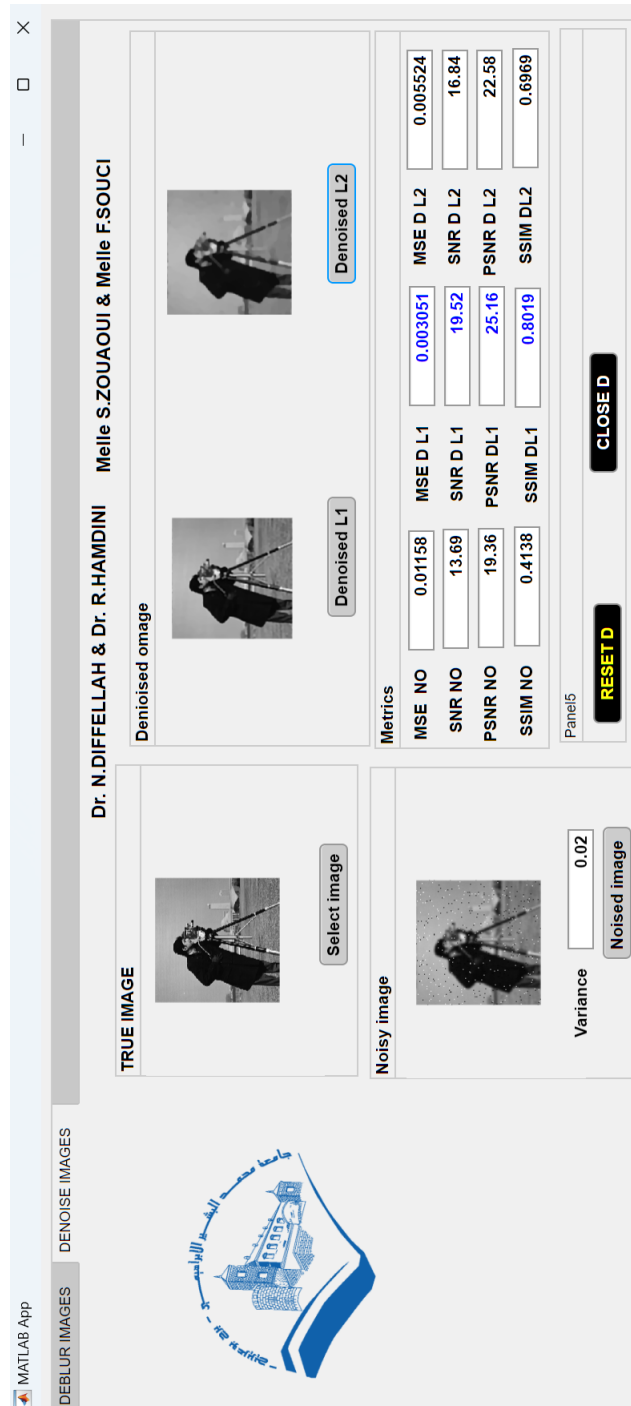


FIGURE 3.4 – Tab 1 : **Design View** du débruitage de l'image cameraman.tif

### 3.4.3 Tab 2 : Le Design View appliqué auau défloutage d'image

La figure 3.5 montre le **Design View** appliqué au défloutage d'image pour le cas de l'image 'cameraman.tif' :

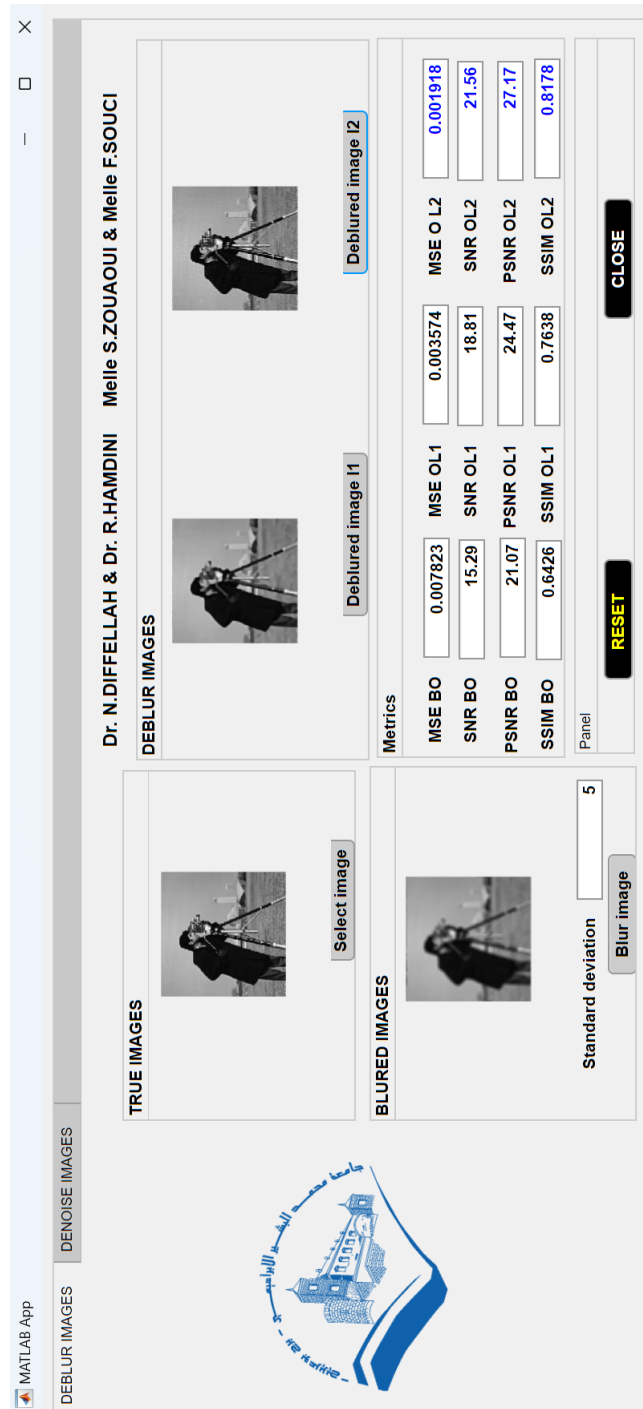


FIGURE 3.5 – Tab 2 : **Design View** du défloutage de l'image cameraman.tif

## 3.4.4 Expérience 1 : Test de l'image IM003.tif

TABLEAU 3.1 – Métriques de l'image 'IM003.tif'

Métrique	$d$	Image bruitée	$\ell_1$	$\ell_2$	$\sigma$	Image floutée	$\ell_1$	$\ell_2$
MSE	0.02	0.01299	0.00558	0.00764	0.2	0.00001	0.00001	0.00001
	0.06	0.02531	0.00576	0.00805	1	0.00402	0.00287	0.00171
	0.08	0.03120	0.00590	0.00837	2	0.00671	0.00569	0.00489
	0.2	0.06802	0.00566	0.01153	3	0.00762	0.00642	0.00489
	0.4	0.12951	0.00816	0.02155	5	0.00811	0.00638	0.00443
	0.6	0.18975	0.03973	0.03721	10	0.00833	0.00627	0.00419
	0.8	0.25195	0.2233	0.05851	20	0.00839	0.00623	0.00413
SNR	0.02	13.52	17.17	15.72	0.2	41.73	42.38	41.72
	0.06	10.74	17.04	15.45	1	18.58	20.10	22.38
	0.08	9.90	16.93	15.26	2	16.33	17.08	17.76
	0.2	6.87	17.11	13.74	3	15.78	16.55	17.75
	0.4	4.58	15.55	10.89	5	15.49	16.57	18.19
	0.6	3.38	9.07	8.45	10	15.37	16.65	18.43
	0.8	2.57	3.32	6.50	20	15.34	16.67	18.50
PSNR	0.02	18.86	22.53	21.17	0.2	47	47.66	47
	0.06	15.97	22.39	20.94	1	23.96	25.42	27.68
	0.08	15.06	22.28	20.77	2	21.73	22.44	23.11
	0.2	11.67	22.47	19.38	3	21.19	21.92	23.11
	0.4	8.87	20.88	16.67	5	20.91	21.95	23.53
	0.6	7.21	14.01	14.30	10	20.79	22.03	23.77
	0.8	5.98	6.51	12.33	20	20.76	22.05	23.86
SSIM	0.02	0.33	0.58	0.47	0.2	0.99	0.99	0.99
	0.06	0.17	0.55	0.46	1	0.71	0.79	0.85
	0.08	0.14	0.53	0.45	2	0.52	0.59	0.65
	0.2	0.08	0.57	0.40	3	0.48	0.54	0.64
	0.4	0.05	0.40	0.31	5	0.46	0.54	0.67
	0.6	0.04	0.18	0.25	10	0.46	0.55	0.68
	0.8	0.03	0.07	0.22	20	0.46	0.55	0.68



(a) Originale



(b)  $\sigma = 0.02$   
MSE=0.01299,SNR=13.52dB  
PSNR=18.86dB,SSIM=0.33

(c)  $\ell_1$   
MSE=0.00558,SNR=17.17dB,  
PSNR=22.53dB,SSIM=0.58

(d)  $\ell_2$   
MSE=0.00764,SNR=15.72dB  
PSNR=21.17dB,SSIM=0.47



(e)  $\sigma = 0.08$   
MSE=0.0312,SNR=9.90dB  
PSNR=15.06dB,SSIM=0.14

(f)  $\ell_1$   
MSE=0.00590,SNR=16.93dB  
PSNR=22.28dB,SSIM=0.53

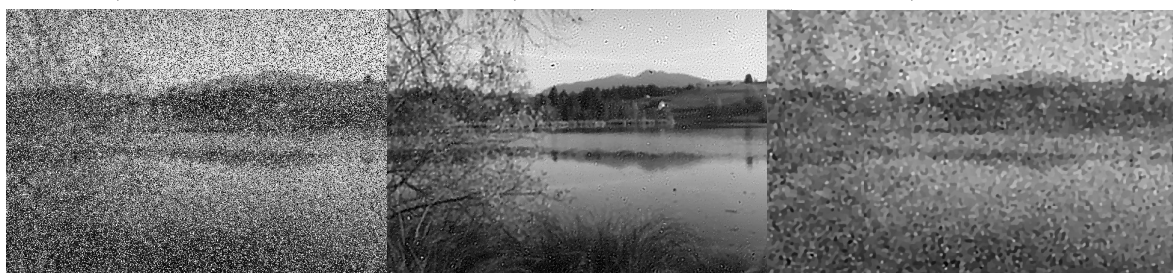
(g)  $\ell_2$   
MSE=0.00837,SNR=15.26dB  
PSNR=20.77dB,SSIM=0.45



(h)  $\sigma = 0.2$   
MSE=0.06802,SNR=6.86dB  
PSNR=11.67dB,SSIM=0.08

(i)  $\ell_1$   
MSE=0.00566,SNR=17.11dB  
PSNR=22.47dB,SSIM=0.57

(j)  $\ell_2$   
MSE=0.01153,SNR=13.74dB  
PSNR=19.38dB,SSIM=0.40



(k)  $\sigma = 0.4$   
MSE=0.1295,SNR=4.58dB  
PSNR=8.87dB,SSIM=0.05

(l)  $\ell_1$   
MSE=0.00816,SNR=15.55dB  
PSNR=20.88dB,SSIM=0.40

(m)  $\ell_2$   
MSE=0.02155,SNR=10.89dB  
PSNR=16.67dB,SSIM=0.32

FIGURE 3.6 – Débruitage de l'image 'IM003.tif' pour différentes valeurs de la variance



(a) Originale



(b)  $d = 1$   
MSE=0.00230, SNR=22.19dB  
PSNR=26.38 dB, SSIM=0.89

(c)  $\ell_1$   
MSE=0.00133, SNR=24.58dB,  
PSNR=28.73 dB, SSIM=0.92

(d)  $\ell_2$   
MSE=0.000818, SNR=26.73dB  
PSNR=30.87 dB, SSIM=0.94



(e)  $d = 3$   
MSE=0.0076, SNR=15.78dB  
PSNR=21.19dB, SSIM=0.49

(f)  $\ell_1$   
MSE=0.006421, SNR=16.55dB  
PSNR=21.92dB, SSIM=0.54

(g)  $\ell_2$   
MSE=0.004892, SNR=17.75dB  
PSNR=23.11dB, SSIM=0.64



(h)  $d = 5$   
MSE=0.008114, SNR=15.49dB  
PSNR=20.91dB, SSIM=0.46

(i)  $\ell_1$   
MSE=0.006387, SNR=16.57dB  
PSNR=21.95dB, SSIM=0.54

(j)  $\ell_2$   
MSE=0.004435, SNR=18.19dB  
PSNR=23.53dB, SSIM=0.67



(k)  $d = 20$   
MSE=0.008395, SNR=15.34dB  
PSNR=20.76dB, SSIM=0.46

(l)  $\ell_1$   
MSE=0.006235, SNR=16.67dB  
PSNR=22.05dB, SSIM=0.55

(m)  $\ell_2$   
MSE=0.004134, SNR=18.5dB  
PSNR=23.86dB, SSIM=0.68

FIGURE 3.7 – Défloutage de l'image 'IM0003.tif' pour différentes valeurs de la densité

## 3.4.5 Expérience 2 : Test de l'image IM052.tif

TABLEAU 3.2 – Métriques de l'image 'IM052.tif'

Métrique	$d$	Image bruitée	$\ell_1$	$\ell_2$	$\sigma$	Image floutée	$\ell_1$	$\ell_2$
MSE	0.02	0.01148	0.00315	0.00482	0.2	0.00001	0.00001	0.00001
	0.06	0.02498	0.00302	0.00552	1	0.00230	0.00133	0.00082
	0.08	0.03201	0.00307	0.00602	2	0.00463	0.00312	0.00242
	0.2	0.07284	0.00312	0.01089	3	0.00568	0.00370	0.002602
	0.4	0.1414	0.00734	0.02568	5	0.00634	0.00359	0.00231
	0.6	0.2095	0.04376	0.04859	10	0.00664	0.00351	0.00216
	0.8	0.2767	0.2351	0.07842	20	0.00672	0.00348	0.00213
SNR	0.02	15.2	20.84	18.91	0.2	42.9083	43.809	42.919
	0.06	11.88	21.03	18.22	1	22.1876	24.585	26.7335
	0.08	10.84	20.96	17.8	2	19.1148	20.880	21.990
	0.2	7.419	20.88	14.96	3	18.219	20.135	21.683
	0.4	4.793	17.21	10.85	5	17.73	20.26	22.21
	0.6	3.325	9.765	7.755	10	17.5287	20.372	22.498
	0.8	2.335	3.643	5.466	20	17.478	20.403	22.559
PSNR	0.02	19.4	25.01	23.17	0.2	47.04	47.942	47.052
	0.06	16.02	25.2	22.58	1	26.3764	28.735	30.872
	0.08	14.95	25.13	22.2	2	23.3377	25.0514	26.1504
	0.2	11.38	25.05	19.63	3	22.455	24.3142	25.8471
	0.4	8.495	21.34	15.9	5	21.9759	24.4406	26.3715
	0.6	6.789	13.59	13.13	10	21.77	24.55	26.655
	0.8	5.579	6.288	11.06	20	21.724	24.5817	26.7161
SSIM	0.02	0.5813	0.8316	0.771	0.2	0.9971	0.99744	0.997
	0.06	0.3955	0.8489	0.7477	1	0.89569	0.92804	0.94493
	0.08	0.3413	0.8447	0.734	2	0.80385	0.8480	0.8760
	0.2	0.1861	0.8446	0.6368	3	0.7704	0.82112	0.86434
	0.4	0.09545	0.6445	0.4725	5	0.75268	0.82325	0.87672
	0.6	0.05712	0.3468	0.3366	10	0.7455	0.82688	0.880394
	0.8	0.03942	0.1219	0.2497	20	0.74374	0.82798	0.88174



(a) Originale



(b)  $\sigma = 0.02$   
MSE=0.01148,SNR=15.2dB  
PSNR=19.4dB,SSIM=0.58



(c)  $\ell_1$   
MSE=0.003157,SNR=20.84dB  
PSNR=25.01dB,SSIM=0.83



(d)  $\ell_2$   
MSE=0.004822,SNR=18.91dB  
PSNR=23.17dB,SSIM=0.77



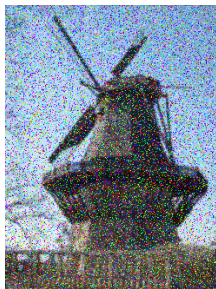
(e)  $\sigma = 0.08$   
MSE=0.03201,SNR=10.84dB  
PSNR=14.95dB,SSIM=0.34



(f)  $\ell_1$   
MSE=0.003071,SNR=20.96dB  
PSNR=25.13dB,SSIM=0.84



(g)  $\ell_2$   
MSE=0.006026,SNR=17.8dB  
PSNR=22.2dB,SSIM=0.73



(h)  $\sigma = 0.2$   
MSE=0.07284,SNR=7.41dB  
PSNR=11.38dB,SSIM=0.18



(i)  $\ell_1$   
MSE=0.003127,SNR=20.88dB  
PSNR=25.05dB,SSIM=0.84



(j)  $\ell_2$   
MSE=0.01089,SNR=14.96dB  
PSNR=19.63dB,SSIM=0.63



(k)  $\sigma = 0.4$   
MSE=0.1414,SNR=4.79dB  
PSNR=8.49dB,SSIM=0.09



(l)  $\ell_1$   
MSE=0.007342,SNR=17.21dB  
PSNR=21.34dB,SSIM=0.64



(m)  $\ell_2$   
MSE=0.02568,SNR=10.85dB  
PSNR=15.9dB,SSIM=0.47

FIGURE 3.8 – Débruitage de l'image 'IM052.tif' pour différentes valeurs de la densité



(a) Originale



(b)  $d = 1$   
MSE=0.00230,SNR=22.19dB  
PSNR=26.38dB,SSIM=0.89



(c)  $\ell_1$   
MSE=0.00133,SNR=24.59dB  
PSNR=28.74dB,SSIM=0.92



(d)  $\ell_2$   
MSE=0.000818,SNR=26.73dB  
PSNR=30.87dB,SSIM=0.94



(e)  $d = 3$   
MSE=0.005681,SNR=18.22dB  
PSNR=22.46dB,SSIM=0.77



(f)  $\ell_1$   
MSE=0.003702,SNR=20.13dB  
PSNR=24.31dB,SSIM=0.82



(g)  $\ell_2$   
MSE=0.0026019,SNR=21.68dB  
PSNR=25.84703dB,SSIM=0.86



(h)  $d = 5$   
MSE=0.00634,SNR=17.73dB  
PSNR=21.97dB,SSIM=0.75



(i)  $\ell_1$   
MSE=0.00359SNR=20.26dB  
PSNR=24.44dB,SSIM=0.82



(j)  $\ell_2$   
MSE=0.0023059,SNR=22.21dB  
PSNR=26.37dB,SSIM=0.87



(k)  $d = 20$   
MSE=0.006723,SNR=17.47dB  
PSNR=21.72dB,SSIM=0.74



(l)  $\ell_1$   
MSE=0.00348,SNR=20.40dB  
PSNR=24.58dB,SSIM=0.82



(m)  $\ell_2$   
MSE=0.00213,SNR=22.55dB  
PSNR=26.71dB,SSIM=0.88

FIGURE 3.9 – Défloutage de l'image 'IM052.tif' pour différentes valeurs de la variance

3.4.6 Expérience 3 : Test de l'image cameraman.tif

TABLEAU 3.3 – Métriques de l'image 'cameraman.tif'

Métrique	$d$	Image bruitée	$\ell_1$	$\ell_2$	$\sigma$	Image floutée	$\ell_1$	$\ell_2$
MSE	0.02	0.01177	0.00306	0.00551	0.2	0.000020	0.00002	0.00002
	0.06	0.0239	0.00309	0.00615	1	0.00254	0.00114	0.00072
	0.08	0.02964	0.00309	0.00636	2	0.00552	0.00327	0.00249
	0.2	0.06608	0.00325	0.01046	3	0.006919	0.003946	0.002227
	0.4	0.1274	0.004333	0.01987	5	0.007826	0.003573	0.001909
	0.6	0.1876	0.03658	0.03473	10	0.00825	0.00341	0.00177
	0.8	0.2476	0.2157	0.05638	20	0.00835	0.00336	0.00176
SNR	0.02	13.62	19.5	16.86	0.2	41.41	41.86	41.45
	0.06	10.69	19.46	16.33	1	20.27	23.84	25.86
	0.08	9.83	19.46	16.17	2	16.84	19.21	20.41
	0.2	6.75	19.24	13.91	3	15.84	18.38	20.9
	0.4	4.49	18	11.02	5	15.29	18.81	21.58
	0.6	3.37	9.155	8.66	10	15.06	19.02	21.85
	0.8	2.60	3.33	6.67	20	15	19.07	21.92
PSNR	0.02	19.29	25.13	22.59	0.2	46.99	47.44	47.03
	0.06	16.22	25.1	22.11	1	25.95	29.44	31.45
	0.08	15.28	25.09	21.96	2	22.58	24.85	26.03
	0.2	11.8	24.87	19.8	3	21.6	24.04	26.52
	0.4	8.949	23.63	17.02	5	21.06	24.47	27.19
	0.6	7.268	14.37	14.59	10	20.84	24.67	27.47
	0.8	6.06	6.66	12.49	20	20.78	24.73	27.53
SSIM	0.02	0.41	0.80	0.69	0.2	0.98	0.99	0.98
	0.06	0.20	0.8	0.66	1	0.83	0.90	0.88
	0.08	0.16	0.79	0.66	2	0.70	0.78	0.8198
	0.2	0.07	0.79	0.53	3	0.66	0.75	0.81
	0.4	0.04	0.68	0.40	5	0.64	0.76	0.81
	0.6	0.036	0.21	0.29	10	0.63	0.77	0.81
	0.8	0.02	0.077	0.23	20	0.63	0.77	0.81



FIGURE 3.10 – Débruitage de l'image 'cameraman.tif' pour différentes valeurs de la densité



(a) Originale



(b)  $d = 1$   
MSE=0.002541,SNR=20.27dB  
PSNR=25.95dB,SSIM=0.84



(c)  $\ell_1$   
MSE=0.001137,SNR=23.84dB  
PSNR=29.44dB,SSIM=0.90



(d)  $\ell_2$   
MSE=0.0007161,SNR=25.86dB  
PSNR=31.45dB,SSIM=0.88



(e)  $\sigma = 3$   
MSE=0.006919,SNR=15.84dB  
PSNR=21.6dB,SSIM=0.66



(f)  $\ell_1$   
MSE=0.003946,SNR=18.38dB  
PSNR=24.04dB,SSIM=0.76



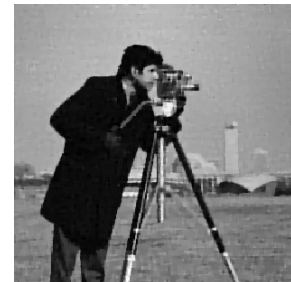
(g)  $\ell_2$   
MSE=0.002227,SNR=20.9dB  
PSNR=26.52dB,SSIM=0.82



(h)  $\sigma = 5$   
MSE=0.007826,SNR=15.29dB  
PSNR=21.06dB,SSIM=0.64



(i)  $\ell_1$   
MSE=0.003573,SNR=18.81dB  
PSNR=24.47dB,SSIM=0.76



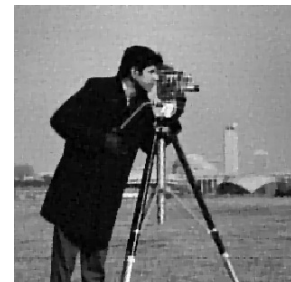
(j)  $\ell_2$   
MSE=0.001909,SNR=21.58dB  
PSNR=27.19dB,SSIM=0.82



(k)  $d = 20$   
MSE=0.008357,SNR=15dB  
PSNR=20.78dB,SSIM=0.63



(l)  $\ell_1$   
MSE=0.003367,SNR=19.07dB  
PSNR=24.73dB,SSIM=0.77



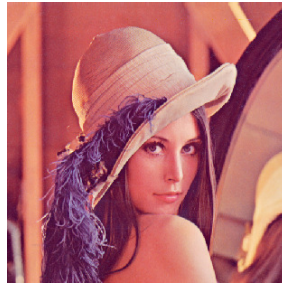
(m)  $\ell_2$   
MSE=0.001764,SNR=21.92dB  
PSNR=27.53dB,SSIM=0.81

FIGURE 3.11 – Défloutage de l'image *cameraman.tif'* pour différentes valeurs de la variance

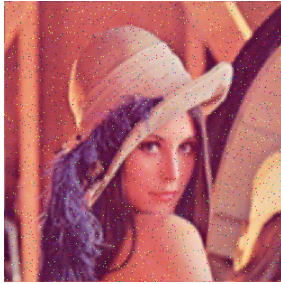
## 3.4.7 Expérience 4 : Test de l'image 'lena.png'

TABLEAU 3.4 – Métriques de l'image 'lena.png'

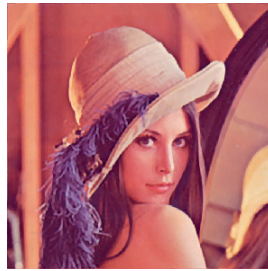
Métrique	$d$	Image bruitée	$\ell_1$	$\ell_2$	$\sigma$	Image floutée	$\ell_1$	$\ell_2$
MSE	0.02	0.00756	0.00086	0.00183	0.2	0.000019	0.00002	0.000011
	0.06	0.0194	0.00099	0.0023	1	0.000639	0.00037	0.00042
	0.08	0.025	0.00096	0.0026	2	0.00156	0.00088	0.00073
	0.2	0.0616	0.0010	0.0056	3	0.0020	0.00115	0.00089
	0.4	0.1218	0.0029	0.0150	5	0.00237	0.00116	0.00088
	0.6	0.181	0.033	0.029	10	0.00252	0.001131	0.0008
	0.8	0.24158	0.21271	0.0482	20	0.00256	0.00112	0.00087
SNR	0.02	16.08	25.49	22.16	0.2	41.85	41.441	41.86
	0.06	12.08	24.86	21.09	1	26.77	29.13	28.56
	0.08	10.90	24.99	20.53	2	22.89	25.35	26.20
	0.2	7.43	24.79	17.00	3	21.69	24.23	25.34
	0.4	4.95	20.24	12.61	5	21.04	24.166	25.39
	0.6	3.63	10.04	9.63	10	20.77	24.30	25.41
	0.8	2.79	3.581	7.40	20	20.70	24.32	25.44
PSNR	0.02	21.21	30.64	27.37	0.2	46.99	46.57	46.99
	0.06	17.11	30.01	26.35	1	31.94	34.27	33.69
	0.08	15.86	30.14	25.83	2	28.07	30.50	31.35
	0.2	12.10	29.94	22.44	3	26.89	29.38	30.49
	0.4	9.14	25.36	18.23	5	26.24	29.32	30.53
	0.6	7.40	14.81	15.34	10	25.98	29.46	30.56
	0.8	6.16	6.72	13.16	20	25.91	29.48	30.59
SSIM	0.02	0.84	0.97	0.95	0.2	0.99	0.99	0.99
	0.06	0.67	0.97	0.94	1	0.98	0.98	0.98
	0.08	0.60	0.97	0.94	2	0.96	0.97	0.97
	0.2	0.37	0.97	0.88	3	0.95	0.96	0.97
	0.4	0.19	0.92	0.71	5	0.95	0.96	0.97
	0.6	0.10	0.61	0.51	10	0.94	0.96	0.97
	0.8	0.06	0.22	0.34	20	0.94	0.97	0.97



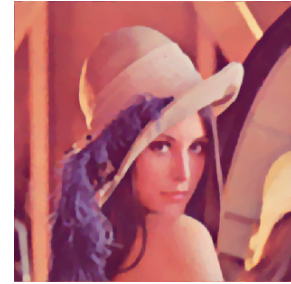
(a) Originale



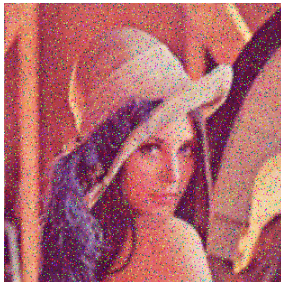
(b)  $\sigma = 0.02$   
MSE=0.00756,SNR=16.08dB  
PSNR=21.21dB,SSIM=0.84



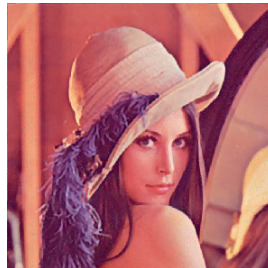
(c)  $\ell_1$   
MSE=0.00086,SNR=25.49dB  
PSNR=30.64dB,SSIM=0.97



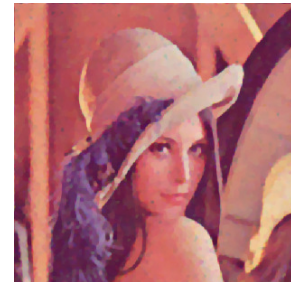
(d)  $\ell_2$   
MSE=0.00183,SNR=22.16dB  
PSNR=27.37dB,SSIM=0.95



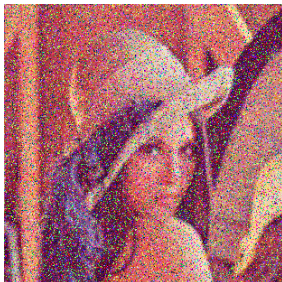
(e)  $\sigma = 0.08$   
MSE=0.025,SNR=10.90dB  
PSNR=15.86dB,SSIM=0.60



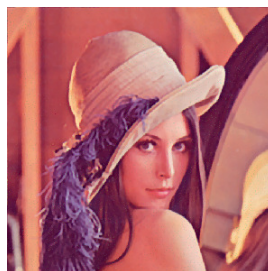
(f)  $\ell_1$   
MSE=0.00096,SNR=24.99dB  
PSNR=30.14dB,SSIM=0.97



(g)  $\ell_2$   
MSE=0.0026,SNR=20.53dB  
PSNR=25.83dB,SSIM=0.94



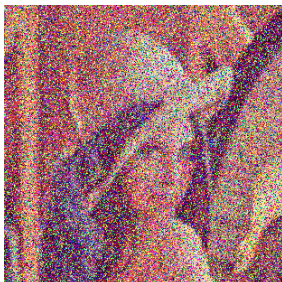
(h)  $\sigma = 0.2$   
MSE=0.0616,SNR=7.43dB  
PSNR=12.10dB,SSIM=0.37



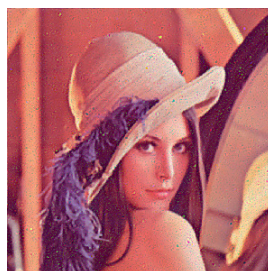
(i)  $\ell_1$   
MSE=0.0010,SNR=24.79dB  
PSNR=29.94dB,SSIM=0.97



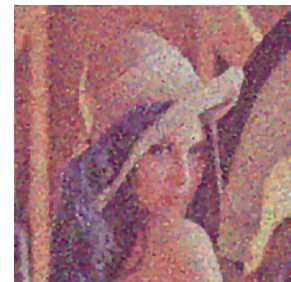
(j)  $\ell_2$   
MSE=0.0056,SNR=17.00dB  
PSNR=22.44dB,SSIM=0.88



(k)  $\sigma = 0.4$   
MSE=0.1218,SNR=4.95dB  
PSNR=9.14dB,SSIM=0.19

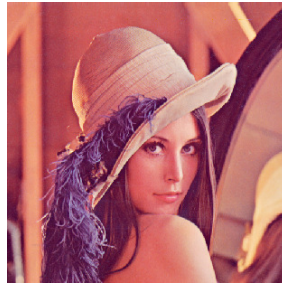


(l)  $\ell_1$   
MSE=0.0029,SNR=20.24dB  
PSNR=25.36dB,SSIM=0.92

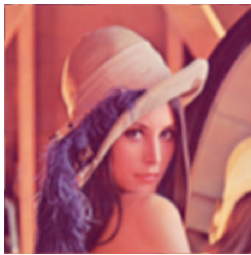


(m)  $\ell_2$   
MSE=0.0150,SNR=12.61dB  
PSNR=18.23dB,SSIM=0.71

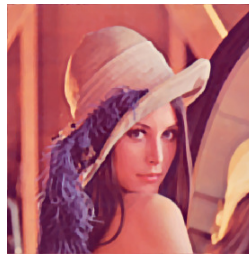
FIGURE 3.12 – Débruitage de l'image 'lena.pgg' pour différentes valeurs de la densité



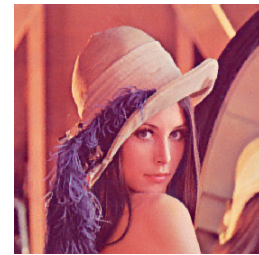
(a) Originale



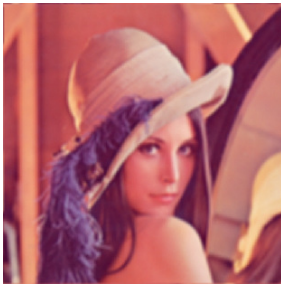
(b)  $d = 1$   
MSE=0.000639,SNR=26.79dB  
PSNR=31.94dB,SSIM=0.98



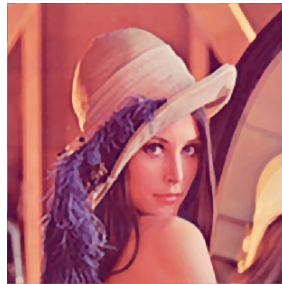
(c)  $\ell_1$   
MSE=0.000373,SNR=29.14dB  
PSNR=34.27dB,SSIM=0.99



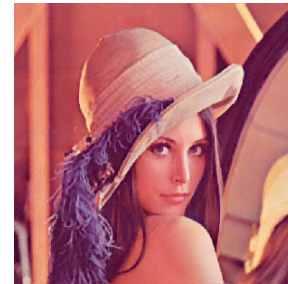
(d)  $\ell_2$   
MSE=0.000426,SNR=28.56dB  
PSNR=33.69dB,SSIM=0.98



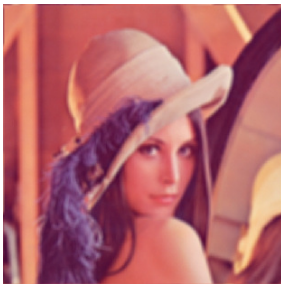
(e)  $\sigma = 3$   
MSE=0.0020,SNR=21.69dB  
PSNR=26.89dB,SSIM=0.95



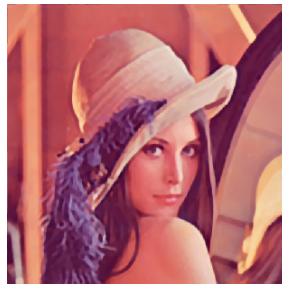
(f)  $\ell_1$   
MSE=0.00115,SNR=24.23dB  
PSNR=29.38dB,SSIM=0.96



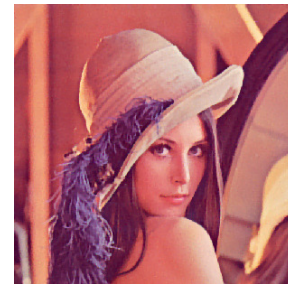
(g)  $\ell_2$   
MSE=0.00089,SNR=25.34dB  
PSNR=30.49dB,SSIM=0.97



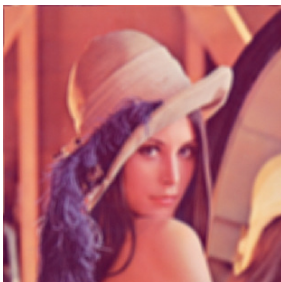
(h)  $d = 5$   
MSE=0.00237,SNR=21.04dB  
PSNR=26.24dB,SSIM=0.95



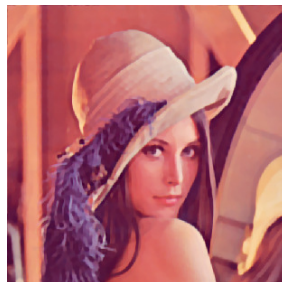
(i)  $\ell_1$   
MSE=0.00116,SNR=24.16dB  
PSNR=29.32dB,SSIM=0.96



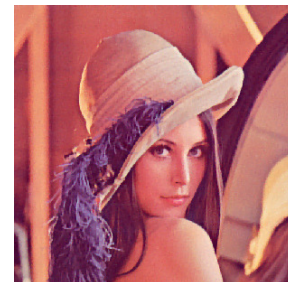
(j)  $\ell_2$   
MSE=0.00088,SNR=25.39dB  
PSNR=30.53dB,SSIM=0.97



(k)  $d = 20$   
MSE=0.00256,SNR=20.70dB  
PSNR=25.91dB,SSIM=0.94



(l)  $\ell_1$   
MSE=0.00112,SNR=24.32dB  
PSNR=29.48dB,SSIM=0.97



(m)  $\ell_2$   
MSE=0.00087,SNR=25.44dB  
PSNR=30.59dB,SSIM=0.97

FIGURE 3.13 – Défloutage de l'image 'lena.png' pour différentes valeurs de la variance

## 3.4.8 Expérience 5 : Test de l'image IM096.tif

TABLEAU 3.5 – Métriques de l'image 'IM096.tif'

Métrique	$d$	Image bruitée	$\ell_1$	$\ell_2$	$\sigma$	Image floutée	$\ell_1$	$\ell_2$
MSE	0.02	0.00992	0.00195	0.00366	0.2	0.00001	0.000017	0.00001
	0.06	0.02278	0.00214	0.00430	1	0.00152	0.00067	0.00051
	0.08	0.02912	0.00219	0.00474	2	0.00349	0.00202	0.00150
	0.2	0.06732	0.00205	0.00886	3	0.00448	0.00252	0.00147
	0.4	0.13120	0.00530	0.02114	5	0.00514	0.00236	0.00131
	0.6	0.19613	0.03895	0.03943	10	0.00546	0.00219	0.00125
	0.8	0.25957	0.22287	0.06403	20	0.00554	0.00216	0.00124
SNR	0.02	14.82	21.88	19.07	0.2	41.83	42.84	41.83
	0.06	11.32	21.49	18.30	1	22.92	26.50	27.67
	0.08	10.31	21.39	17.85	2	19.29	21.73	23.02
	0.2	7.00	21.67	14.98	3	18.19	20.77	23.11
	0.4	4.58	17.58	11.02	5	17.58	21.05	23.61
	0.6	3.28	9.32	8.21	10	17.32	21.36	23.83
	0.8	2.46	3.36	6.13	20	17.26	21.44	23.87
PSNR	0.02	20.03	27.07	24.35	0.2	47.00	48.01	47.00
	0.06	16.42	26.68	23.65	1	28.15	31.68	32.84
	0.08	15.35	26.59	23.23	2	24.56	26.93	28.21
	0.2	11.71	26.87	20.52	3	23.48	25.98	28.30
	0.4	8.82	22.75	16.74	5	22.88	26.26	28.80
	0.6	7.07	14.09	14.04	10	22.62	26.57	29.02
	0.8	5.85	6.51	11.93	20	22.56	26.65	29.05
SSIM	0.02	0.73	0.92	0.86	0.2	0.99	0.99	0.99
	0.06	0.56	0.90	0.84	1	0.94	0.97	0.97
	0.08	0.51	0.90	0.83	2	0.88	0.92	0.92
	0.2	0.31	0.91	0.76	3	0.85	0.90	0.93
	0.4	0.16	0.80	0.59	5	0.84	0.90	0.94
	0.6	0.09	0.52	0.42	10	0.83	0.91	0.94
	0.8	0.05	0.19	0.29	20	0.83	0.91	0.95

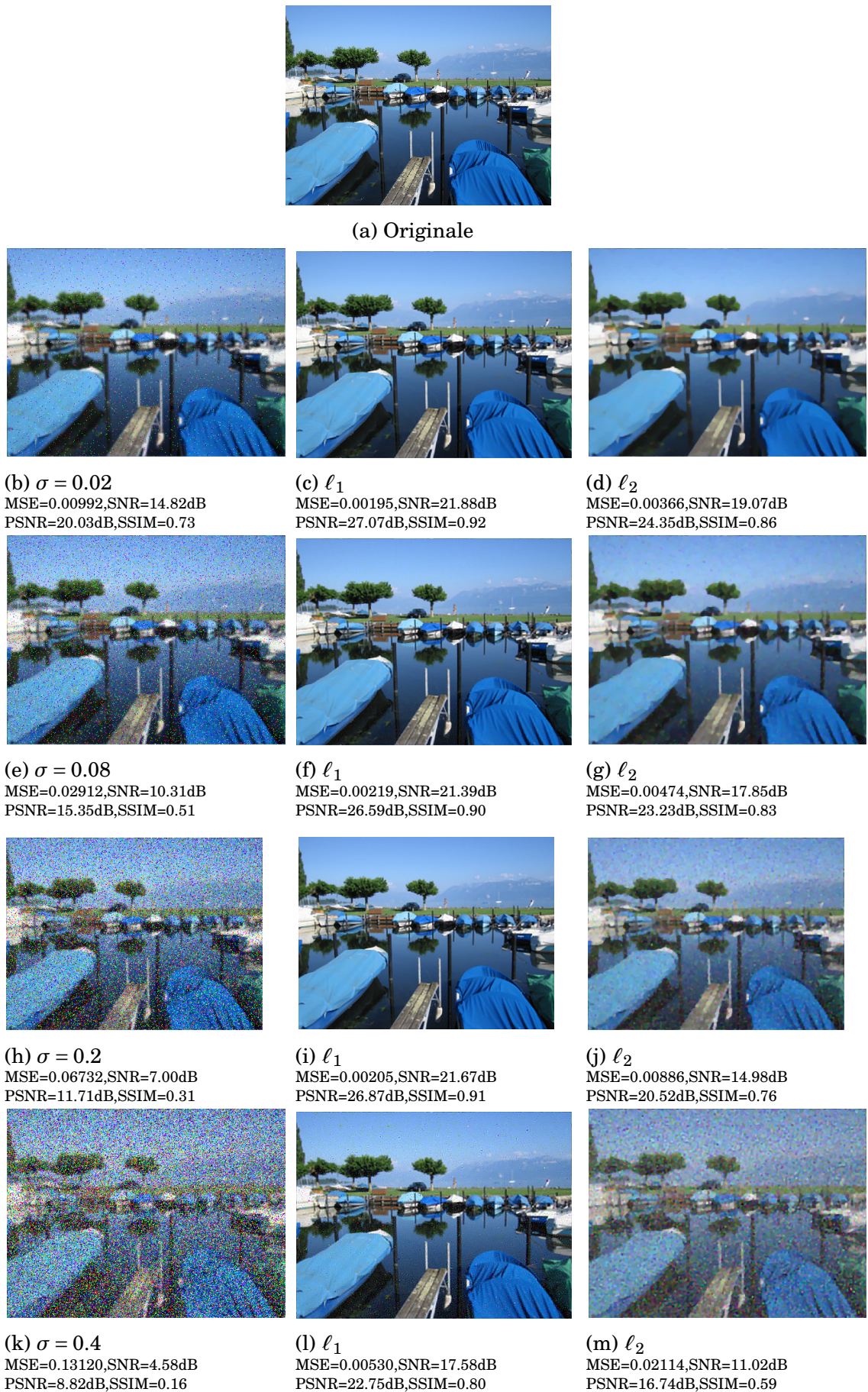


FIGURE 3.14 – Débruitage de l'image 'IM096.tif' pour différentes valeurs de la densité



(a) Originale



(b)  $d = 1$   
MSE=0.00152,SNR=22.92dB  
PSNR=22.92dB,SSIM=0.94

(c)  $\ell_1$   
MSE=0.00067,SNR=26.50dB  
PSNR=26.50dB,SSIM=0.97

(d)  $\ell_2$   
MSE=0.00051,SNR=27.67dB  
PSNR=27.67dB,SSIM=0.97



(e)  $d = 3$   
MSE=0.00448,SNR=18.19dB  
PSNR=23.48dB,SSIM=0.85

(f)  $\ell_1$   
MSE=0.00252,SNR=20.77dB  
PSNR=25.98dB,SSIM=0.90

(g)  $\ell_2$   
MSE=0.00147,SNR=23.11dB  
PSNR=28.30dB,SSIM=0.93



(h)  $d = 5$   
MSE=0.00514,SNR=17.58dB  
PSNR=22.88dB,SSIM=0.84

(i)  $\ell_1$   
MSE=0.00236,SNR=21.05dB  
PSNR=26.26dB,SSIM=0.90

(j)  $\ell_2$   
MSE=0.00131,SNR=23.61dB  
PSNR=28.80dB,SSIM=0.94



(k)  $d = 20$   
MSE=0.00554,SNR=17.26dB  
PSNR=22.56dB,SSIM=0.83

(l)  $\ell_1$   
MSE=0.00216dB,SNR=21.44  
PSNR=26.65dB,SSIM=0.91

(m)  $\ell_2$   
MSE=0.00124,SNR=23.87dB  
PSNR=29.05dB,SSIM=0.95

FIGURE 3.15 – Défloutage de l'image 'IM096.tif' pour différentes valeurs de la variance

3.4.9 Expérience 6 : Test de l'image IM008.tif

TABLEAU 3.6 – Métriques de l'image 'IM008.tif'

Métrique	$d$	Image bruitée	$\ell_1$	$\ell_2$	$\sigma$	Image floutée	$\ell_1$	$\ell_2$
MSE	0.02	0.00858	0.00183	0.00356	0.2	0.00001	0.00001	0.00001
	0.06	0.01981	0.00187	0.00397	1	0.00130	0.00068	0.00050
	0.08	0.02542	0.00191	0.00422	2	0.00298	0.00193	0.00148
	0.2	0.05909	0.00185	0.00656	3	0.00383	0.00235	0.00139
	0.4	0.11540	0.00303	0.01357	5	0.00441	0.00235	0.00123
	0.6	0.17102	0.02326	0.02339	10	0.00468	0.00229	0.00121
	0.8	0.22734	0.20198	0.03681	20	0.00475	0.00229	0.00120
SNR	0.02	14.97	21.66	18.70	0.2	41.32	42.67	41.32
	0.06	11.49	21.56	18.21	1	23.10	25.97	27.25
	0.08	10.48	21.48	17.93	2	19.48	21.43	22.59
	0.2	7.23	21.60	15.95	3	18.38	20.56	22.85
	0.4	4.92	19.4	12.74	5	17.77	20.55	23.39
	0.6	3.74	10.93	10.40	10	17.51	20.66	23.47
	0.8	2.98	3.52	8.50	20	17.44	20.67	23.48
PSNR	0.02	20.66	27.37	24.48	0.2	47.00	48.35	46.99
	0.06	17.02	27.27	24.00	1	28.83	31.65	32.93
	0.08	15.94	27.18	23.73	2	25.24	27.14	28.29
	0.2	12.28	27.30	21.82	3	24.16	26.27	28.55
	0.4	9.37	25.11	18.78	5	23.55	26.27	29.09
	0.6	7.66	16.33	16.30	10	23.29	26.38	29.16
	0.8	6.43	6.94	14.33	20	23.23	26.39	29.17
SSIM	0.02	0.43	0.84	0.72	0.2	0.98	0.99	0.98
	0.06	0.20	0.83	0.70	1	0.88	0.93	0.90
	0.08	0.15	0.82	0.69	2	0.76	0.83	0.86
	0.2	0.07	0.84	0.59	3	0.71	0.80	0.86
	0.4	0.04	0.70	0.43	5	0.69	0.79	0.86
	0.6	0.03	0.32	0.33	10	0.68	0.80	0.85
	0.8	0.02	0.07	0.27	20	0.68	0.80	0.85



(a) Originale



(b)  $\sigma = 0.02$   
MSE=0.00858,SNR=14.97dB  
PSNR=20.66dB,SSIM=0.43



(c)  $\ell_1$   
MSE=0.00183,SNR=21.66dB  
PSNR=27.37dB,SSIM=0.84



(d)  $\ell_2$   
MSE=0.00356,SNR=18.70dB  
PSNR=24.48dB,SSIM=0.72



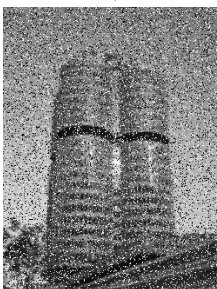
(e)  $\sigma = 0.08$   
MSE=0.02542,SNR=10.48dB  
PSNR=15.94dB,SSIM=0.15



(f)  $\ell_1$   
MSE=0.00191,SNR=21.48dB  
PSNR=27.18dB,SSIM=0.82



(g)  $\ell_2$   
MSE=0.00422,SNR=17.93dB  
PSNR=23.73dB,SSIM=0.69



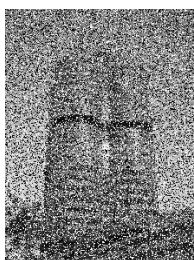
(h)  $\sigma = 0.2$   
MSE=0.05909,SNR=7.23dB  
PSNR=12.28dB,SSIM=0.07



(i)  $\ell_1$   
MSE=0.00185,SNR=21.60dB  
PSNR=27.30dB,SSIM=0.84



(j)  $\ell_2$   
MSE=0.00656,SNR=15.95dB  
PSNR=21.82dB,SSIM=0.59



(k)  $\sigma = 0.4$   
MSE=0.11540,SNR=4.92dB  
PSNR=9.37dB,SSIM=0.04



(l)  $\ell_1$   
MSE=0.00303,SNR=19.4dB  
PSNR=25.11dB,SSIM=0.70



(m)  $\ell_2$   
MSE=0.01357,SSIM=12.74dB  
PSNR=18.78dB,SSIM=0.43

FIGURE 3.16 – Débruitage de l'image 'IM008.tif' pour différentes valeurs de la densité



(a) Originale



(b)  $d = 1$   
MSE=0.00130,SNR=23.10dB  
PSNR=28.83dB,SSIM=0.88



(c)  $\ell_1$   
MSE=0.00068,SNR=25.97dB  
PSNR=31.65dB,SSIM=0.93



(d)  $\ell_2$   
MSE=0.00050,SNR=27.25dB  
PSNR=32.93dB,SSIM=0.90



(e)  $d = 3$   
MSE=0.00383,SNR=18.38dB  
PSNR=24.16dB,SSIM=0.71



(f)  $\ell_1$   
MSE=0.00235,SNR=20.56dB  
PSNR=26.27dB,SSIM=0.80



(g)  $\ell_2$   
MSE=0.00139,SNR=22.85dB  
PSNR=28.55dB,SSIM=0.86



(h)  $d = 5$   
MSE=0.00441,SNR=17.77dB  
PSNR=23.55dB,SSIM=0.69



(i)  $\ell_1$   
MSE=0.00235,SNR=20.55dB  
PSNR=26.27dB,SSIM=0.79



(j)  $\ell_2$   
MSE=0.00123,SNR=23.39dB  
PSNR=29.09dB,SSIM=0.86



(k)  $d = 20$   
MSE=0.00475,SNR=17.44dB  
PSNR=23.23dB,SSIM=0.68



(l)  $\ell_1$   
MSE=0.00229,SNR=20.67dB  
PSNR=26.39dB,SSIM=0.80



(m)  $\ell_2$   
MSE=0.00120,SNR=23.48dB  
PSNR=29.17dB,SSIM=0.85

FIGURE 3.17 – Défloutage de l'image 'IM008.tif' pour différentes valeurs de la variance

### 3.4.10 Interprétation et discussion des résultats

Les simulations informatiques des algorithmes de restauration proposés sont effectuées sous MATLAB APPDESIGNER. Cependant, six images, parmi celles couramment utilisées dans le domaine de la restauration d'images, sont utilisées dans ce mémoire, comme le montre la figure 3.3.

Les paramètres statistiques tels que  $MSE$ ,  $SNR$ ,  $PSNR$  et  $SSIM$  sont calculés pour les images filtrées obtenues à partir de chaque filtre.

Les tableaux 3.1,3.2,3.3,3.4,3.5 et 3.6 illustrent les différentes métriques pour des densités de bruit et des variances de flou variant de 0.02, 0.06, 0.08, 0.2, 0.4, 0.6, 0.8 et de 0.2, 1, 2, 3, 5, 10, et 20 respectivement.

Pour les deux types de filtres et quelque soit l'image, lorsque les valeurs  $SNR$ ,  $PSNR$  et  $SSIM$  diminuent, les valeurs de  $MSE$  et augmentent, donc l'opération de comparaison et le choix du meilleur filtre sont liés à ces paramètres statistiques. A partir des tableaux obtenues : à mesure que la variance du bruit (ou la densité du flou) augmente, le  $MSE$  augmente également et le  $SNR$ ,  $PSNR$  et  $SSIM$  diminuent. Pour des valeurs plus élevées de variance de bruit (densité du flou), la performance du filtre réduit légèrement. Sur les deux filtres utilisés,  $\ell_1$  donne de meilleurs résultats pour le débruitage et  $\ell_2$  donne des résultats plus efficaces pour le défloutage.

Les figures de 3.6 à 3.17 et les tableaux de 3.1 à 3.6 nous permettent de choisir le filtre le plus adapté à chaque type d'image et à chaque niveau de densité et de variance de bruit.

Les figures 3.6, 3.8, 3.10, 3.12, 3.14 et 3.16 montrent les images restaurées par les deux normes pour différentes valeurs de la variance. À partir de l'analyse des résultats obtenus, nous pouvons conclure que : La méthode  $\ell_1$  est plus adaptée que la méthode  $\ell_2$  pour restaurer des images bruitées par un bruit sel et poivre pour plusieurs raisons :

- Elle attribue un poids moindre aux valeurs extrêmes, réduisant ainsi leur impact sur l'image restaurée.
- La méthode  $\ell_1$  pénalise moins les grandes différences de valeurs entre les pixels voisins, ce qui permet de conserver les transitions nettes et les détails critiques de l'image.
- La méthode  $\ell_1$  favorise des solutions où la plupart des coefficients sont nuls ou proches de zéro, aidant ainsi à éliminer le bruit tout en maintenant les éléments structurels de l'image.

Les figures 3.7, 3.9, 3.11, 3.13, 3.15 et 3.17 montrent le défloutage des images par les deux normes et ce pour différentes valeurs de la densité. Le flou est caractérisé par une perte de netteté et des détails brouillés. la méthode  $\ell_2$  est plus efficace que  $\ell_1$ , elle améliore la netteté en conservant les bords. La méthode  $\ell_1$  aide à préserver les bords et certains détails sans lisser intensément les transitions, rendant les contours plus nets comparés à l'image floutée d'origine. Le succès dépend de la sévérité du flou, mais globalement, elle rend l'image plus définie.

Les résultats montrent clairement que la déconvolution par les méthodes variationnelles donne des résultats satisfaisants du point de vue visuel ou métrique. Cependant, on observe que la déconvolution par la norme 2 est meilleure que celle par la norme 1. Cela se traduit par un  $MSE$  plus petit, un  $SNR$  et un  $PSNR$  plus élevés, ainsi qu'un  $SSIM$  proche de la valeur 1.

### 3.5 Conclusion

Les méthodes variationnelles sont très intéressantes dans la restauration d'images. Les résultats trouvés par le simulateur ont montré que quelque soit l'image choisie et quelque soit sa taille, la méthode  $\ell_1$  est plus efficace pour restaurer des images bruitées par un bruit sel et poivre en raison de sa capacité à traiter les valeurs aberrantes et à préserver les détails essentiels de l'image. En revanche, la méthode  $\ell_1$  est moins efficace pour traiter le flou, car elle tend à lisser uniformément les transitions, ce qui réduit la netteté des bords et les détails fins. Elle peut exacerber la perte de netteté, rendant les bords moins définis et les détails significativement réduits. la méthode  $\ell_2$  corrige le flou mieux que la norme  $\ell_1$ .



# Conclusion

*Tout le monde est un génie. Mais si vous jugez un poisson à sa capacité de grimper à un arbre, il vivra toute sa vie en croyant qu'il est stupide.'*

# Conclusion générale

Le développement d'un simulateur de restauration variationnelle d'images par filtrage et déconvolution vise à concevoir un outil capable de restaurer des images dégradées en corrigeant le flou et réduisant le bruit tout en conservant les détails importants. Le propos de cet projet est la restauration d'images, qui consiste à reconstituer au mieux des images ayant subi des dégradations, il s'agit de supprimer ces dégradations (bruit, flou), afin de retrouver l'image originale, en optimisant un certain critère. Nous avons considéré deux problèmes de restauration d'images, exprimés comme des problèmes de minimisation de la norme en utilisant la variation totale comme terme de régularisation et la norme  $\ell_1$  et  $\ell_2$  comme termes d'attaches aux données. Ce simulateur permet de trouver un équilibre optimal entre le lissage des zones homogènes et la préservation des contours et des structures essentielles. Ce travail représente une avancée significative dans le domaine de la restauration d'images, offrant des solutions robustes pour améliorer la netteté et récupérer les détails perdus dans des images floues ou bruitées. La méthode  $\ell_1$  démontre une efficacité notable dans le débruitage d'images par contre la norme  $\ell_2$  permet de corriger le flou d'une manière satisfaisante. La comparaison entre les deux méthodes a été basée sur les mesures de qualités *MSE*, *SNR*, *PSNR* et *SSIM*.

Comme perspectives futures de ce travail nous viseros deux axes :

- L'un des avantages de la variation totale est qu'elle minimise les changements brusques dans l'image, ce qui permet une reconstruction plus douce. D'un autre côté, l'un des inconvénients de cette approche est la perte de contraste dans la reconstruction. Les transitions irrégulières peuvent devenir moins visibles. Il faut donc trouver un équilibre entre le lissage et la préservation des détails importants de l'image.
- Les résultats satisfaisants de la restauration par variation totale encourage l'extension de cette approche à la restauration de vidéos. Les vidéos souffrent des mêmes types de dégradations que les images fixes, mais ajoutent des défis supplémentaires comme la cohérence temporelle.

# Bibliographie

- [1] R. C. Gonzalez, *Digital image processing*. Pearson education india, 2009.
- [2] N. Diffellah, T. Bekkouche, and R. Hamdini, “Image denoising algorithms using norm minimization techniques,” in *Second international conference on complex systems and their applications*, 2021, pp. 242–252.
- [3] S. Kaur, “Noise types and various removal techniques,” *International Journal of Advanced Research in Electronics and Communication Engineering (IJA-RECE)*, vol. 4, no. 2, pp. 226–230, 2015.
- [4] S. S. Al-Amri, N. V. Kalyankar, and S. D. Khamitkar, “A comparative study of removal noise from remote sensing image,” *arXiv preprint arXiv :1002.1148*, 2010.
- [5] S. V. Dangeti, *Denoising techniques-a comparison*. Louisiana State University and Agricultural & Mechanical College, 2003.
- [6] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment : from error visibility to structural similarity,” *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [7] L. I. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Physica D : nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.
- [8] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, “An augmented lagrangian method for total variation video restoration,” *IEEE Transactions on Image Processing*, vol. 20, no. 11, pp. 3097–3111, 2011.
- [9] S. Adhikari and R. Baer, “Augmented lagrangian method for order-n electronic structure,” *The Journal of chemical physics*, vol. 115, no. 1, pp. 11–14, 2001.
- [10] J. Ahr, “Development of a gui to evaluate strain specific reaction parameters using matlab app designer,” Ph.D. dissertation, Hochschule für Angewandte Wissenschaften Hamburg, 2022.

## ملخص

في هذا المشروع نقوم بتطوير محاكي رسومي لتصفية وفك تحليل الصور باستخدام تقنيات تباينية. تسمح التطبيق بتحميل، إضافة ضوضاء، ومعالجة مجموعة من الصور بالألوان أو الصور الرمادية. سنقوم بتنفيذ قياسات كمية لتقييم جودة الصور المعالجة تلقائيًا.

كلمات مفتاحية: معالجة الصور، محاكي رسومي، إزالة الضوضاء، فك التحليل، قياس الجودة، أساليب تباينية .

## Résumé

DANS ce projet nous nous intéressons à la création et développement d'un simulateur graphique pour l'utilisation de la procédure de filtrage et deconvolution des images par différentes techniques variationnelles. Les différentes zones de l'application permettent de charger, bruitez et appliquer le traitement sur une collection d'images en niveaux de gris ou couleurs. Des mesures quantitatives seront étudiées et implémentées pour prévoir automatiquement la qualité perceptive de l'image traitée.

*Mots clés : traitement d'images, simulateur graphique, débruitage, deconvolution, mesure de qualité, méthodes variationnelles...*

## Abstract

In this project, we are developing a graphical simulator for filtering and deconvolution of images using variational techniques. The application allows loading, adding noise to, and processing a collection of grayscale or color images. We will implement quantitative measures to automatically assess the perceptual quality of the processed images.

*Keywords : image processing, graphical simulator, denoising, deconvolution, quality measurement, variational methods...*