



**HAL**  
open science

# Restauration d'image via une approche Bayésienne à partir d'un couple d'images d'observations prise dans différentes conditions

Mouhamad Chéhaitly

► **To cite this version:**

Mouhamad Chéhaitly. Restauration d'image via une approche Bayésienne à partir d'un couple d'images d'observations prise dans différentes conditions. Electronique. 2013. dumas-01394266

**HAL Id: dumas-01394266**

**<https://dumas.ccsd.cnrs.fr/dumas-01394266>**

Submitted on 9 Nov 2016

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

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

**CONSERVATOIRE NATIONAL DES ARTS ET METIERS**

Centre Regional Associé Du Liban

**MEMOIRE**

présenté en vue d'obtenir le

**DIPLOME D'INGENIEUR CNAM**

**SPECIALITE: Electronique - Automatique**

**OPTION: Electronique**

par

Mouhamad Chéhaitly

---

*Restauration d'image via une approche Bayésienne à partir d'un couple d'images d'observations prise dans différentes conditions.*

Soutenu le

---

**JURY**

**PRESIDENT:** M. Michel TERRE

**MEMBRES:** Mme Rima HLEISS  
M. Chaouki DIAB  
M. Kaled HITANI

année : 2013



## Remerciements

Mes premiers remerciements sont adressés à Mme Rima HLEISS, pour avoir accepté d'encadrer cette mémoire. Elle m'a permis de travailler dans des conditions optimales en étant d'une extrême disponibilité tout en me laissant une grande autonomie. Toutes les discussions que nous avons pu avoir ont toujours été très enrichissantes. J'espère, dans l'avenir, avoir à nouveau l'occasion de travailler et apprendre à ses côtés, et ainsi profiter de son expérience. Je souhaite de plus la remercier sincèrement de m'avoir accordé sa confiance. J'espère en avoir été digne.

Je remercie M. le professeur Michel Terré pour avoir bien voulu présider le jury de soutenance de ce mémoire, ainsi que les membres de ce jury.

Je souhaite également remercier toutes les personnes qui m'ont aidé et soutenu tout au long de ce projet, en particulier :

M. Chaouki DIAB pour ses conseils,

M. Rida AWADA, directeur de la société Data Link, pour m'avoir fourni les données, le matériel, et l'environnement convenable pour mener à bien ce projet, ainsi que pour ses précieux conseils.

Enfin, je remercie toute l'équipe enseignante du Cnam Liban, qui m'a permis d'effectuer des années d'étude très enrichissante au sein de cet institut.

## Liste des abréviations

|         |  |
|---------|--|
| PSF     | Point Spread Function  |
| ARMA    | Auto Regressive Moving Average   |
| MA      | Moving Average   |
| GCV     | Generalized Cross Validation (Validation Croisée Généralisée)                    |
| BV      | Bounded Variations   |
| TV      | Total Variation  |
| EDP     | Équation aux Dérivées Partielles   |
| LPA     | Local Polynomial Approximation   |
| ICI     | Intersection Confidence Intervals  |
| EM      | Expectation-Maximization   |
| NAS-RIF | Nonnegativity And Support constraints - Recursive Inverse Filtering              |
| ICA     | Independent Composant Analysis   |
| ISO     | international Organization for Standarazation.<br>(depend of light Sensitivity ) |
| KL      | Kullback-Leibler Distance  |
| MM      | Majorization-Minimisation  |
| CG      | Conjugate Gradient   |
| SNR     | Signal to Noise Ratio  |
| MSE     | Mean Square Error  |
| PSNR    | Peak Signal to Noise Ratio   |

# Table DE CONTENU

---

|   |           |
|---|-----------|
| <b>Remerciements</b> .....                                  | <b>3</b>  |
| <b>Liste des abréviations</b> .....                         | <b>4</b>  |
| <b>Table des matières</b> .....                             | <b>5</b>  |
| <b>Résumé</b> .....   | <b>9</b>  |
| <b>Abstract</b> .....                                       | <b>11</b> |
| <b>Introduction générale</b> .....                          | <b>13</b> |
| <b>I. Généralités et problématique</b> .....                | <b>16</b> |
| 1. Eléments de modélisation des images .....                | 16        |
| 1.1. Images en tant que distributions .....                 | 17        |
| 1.2. Images $L^p$ .....                                     | 17        |
| 1.3. Images de Sobolev $H^p$ .....                          | 18        |
| 1.4. Images à variations bornées .....                      | 19        |
| 2. Modélisation des images floues .....                     | 20        |
| 2.1. Images à niveaux de gris .....                         | 20        |
| 2.2. Modélisation du bruit présent .....                    | 22        |
| 3. Modélisation des flous .....                             | 22        |
| 3.1. Flou de défocalisation .....                           | 22        |
| 3.2. Flou de mouvement rectiligne .....                     | 23        |
| 4. Conclusion .....   | 23        |
| <b>II. État de l'art de la Déconvolution</b> .....          | <b>24</b> |
| 1. Déconvolution: un problème inverse mal posé .....        | 24        |
| 2. Méthode monoéchelle .....                                | 25        |
| 3. Méthode multiéchelle .....                               | 28        |
| 4. Conclusion .....   | 29        |
| <b>III. État de l'art de la Déconvolution Aveugle</b> ..... | <b>29</b> |
| 1. Introduction .....                                       | 29        |
| 2. Méthodes avec identification du flou a priori .....      | 30        |

|  |           |
|--|-----------|
| 2.1. Estimation préalable du flou .....  | 30        |
| 2.1.1. Méthodes basées sur la modélisation ARMA .....                                | 31        |
| 2.1.2. Méthodes basées sur les caractéristiques spectrales de l'image .....          | 33        |
| 2.1.3. Méthodes basées sur l'autocorrélation .....                                   | 35        |
| 2.1.4. Méthodes basées sur l'utilisation de la transformée de Radon .....            | 37        |
| 2.1.5. Autres méthodes .....   | 38        |
| 2.2. Restauration de l'image avec flou connu .....                                   | 42        |
| 2.2.1. Méthodes variationnelles .....  | 42        |
| 2.2.2. Méthodes utilisant l'analyse multi résolution et les ondelettes .....         | 49        |
| 2.2.3. Méthodes basées sur les équations aux dérivées partielles .....               | 49        |
| 2.2.4. Autres méthodes .....   | 50        |
| 2.3. Synthèse sur les méthodes avec identification du flou a priori .....            | 52        |
| 3. Méthodes d'identification jointe .....  | 52        |
| 3.1. Méthode directe de séparation de la plage des zéros .....                       | 53        |
| 3.2. Méthode d'Ayers-Dainty.....   | 54        |
| 3.3. Méthode des moindres carrés non-régularisés.....                                | 55        |
| 3.4. Méthodes à double régularisation .....  | 55        |
| 3.4.1. Approches non-paramétriques .....   | 56        |
| 3.4.2. Approches paramétriques .....   | 58        |
| 3.5. Méthodes de filtrage adaptatif .....  | 60        |
| 3.6. Méthodes basées sur l'analyse en composantes indépendantes .....                | 61        |
| 3.7. Méthodes diverses .....   | 62        |
| 3.8. Synthèse sur les méthodes d'identification jointe .....                         | 62        |
| 4. Conclusion .....  | 62        |
| <b>IV. Déconvolution Aveugle Bayésienne à partir de deux images différentes.....</b> | <b>64</b> |
| 1. Introduction .....  | 64        |
| 2. Formulation du problème .....   | 66        |
| 3. Modèle hiérarchique bayésien .....  | 66        |
| 3.1. Modèles d'observation .....   | 67        |
| 3.2. Modèle a priori de flou .....   | 68        |

|  |           |
|--|-----------|
| 3.3. Modèle a priori de l'image .....                        | 69        |
| 3.4. Hyperpriors en fonction des hyperparamètres .....       | 69        |
| 4. Inférence bayésienne variationnelle .....                 | 71        |
| 5. Calcul approximatif de la distribution a posteriori ..... | 71        |
| 6. Synthèse globale .....                                    | 77        |
| <b>V. Résultats Expérimentaux .....</b>                      | <b>79</b> |
| 1. Introduction .....  | 79        |
| 2. Dégradation d'image principe et logiciel.....             | 80        |
| 2.1. Introduction .....                                      | 80        |
| 2.2. Flou et bruit appliqués .....                           | 80        |
| 2.3. Fenêtre principale .....                                | 80        |
| 2.4. Charger une image .....                                 | 82        |
| 2.5. Flouage de l'image sélectionnée .....                   | 84        |
| 2.5.1. Charger le flou .....                                 | 84        |
| 2.5.2. Générer le flou .....                                 | 86        |
| 2.5.2.1. Flou gaussien .....                                 | 87        |
| 2.5.2.2. Flou uniforme .....                                 | 88        |
| 2.5.2.3. Flou personnalisé .....                             | 88        |
| 2.5.2.4. Flou de mouvement (Motion Blur).....                | 89        |
| 2.5.3. Situation après l'application de flou .....           | 90        |
| 2.6. Application du bruit à une image .....                  | 90        |
| 2.6.1. Bruit de Poisson.....                                 | 91        |
| 2.6.2. Bruit Gaussien additif.....                           | 92        |
| 2.6.3. Bruit Gaussien multiplicatif.....                     | 92        |
| 2.6.4. Situation après l'application bruit .....             | 93        |
| 2.7. Résultats sauvegardé .....                              | 94        |
| 2.8. Fermeture de l'application .....                        | 95        |
| 3. Résultat de restauration d'image publie dans [126] .....  | 96        |
| 3.1. Restauration des images de synthèse .....               | 96        |
| 3.2. Restauration des images réelles .....                   | 99        |

|   |            |
|---|------------|
| 4. Nos restaurations d'image: Logiciel et Résultats .....                   | 103        |
| 4.1. Fenêtre principale .....   | 104        |
| 4.2. Charger une image .....  | 105        |
| 4.3. Précision la taille de PSF .....                                       | 107        |
| 4.3.1. Charger la taille de PSF .....                                       | 107        |
| 4.3.2. Recharger la taille de PSF .....                                     | 108        |
| 4.4. Mettre en marche le programme .....                                    | 109        |
| 4.5. Fermeture de l'application .....                                       | 110        |
| 5. Restauration des images de synthèse en utilisant l'Algorithme1 .....     | 110        |
| 5.1. Restauration des images de synthèse : Modèle1 .....                    | 110        |
| 5.2. Restauration des images de synthèse : Modèle2 .....                    | 115        |
| 5.3. Restauration des images de synthèse : Modèle3 .....                    | 117        |
| 5.4. Restauration des images de synthèse : Modèle4 .....                    | 119        |
| 6. Restauration des images de synthèse en utilisant l'Algorithme2 .....     | 122        |
| 6.1. Restauration des images de synthèse : Modèle1 .....                    | 122        |
| 6.2. Restauration des images de synthèse : Modèle2 .....                    | 123        |
| 7. Restauration avec fonction déconvolution Aveugle définie en Matlab ..... | 125        |
| 7.1. Déconvolution Aveugle en Matlab.: Modèle1 .....                        | 125        |
| 7.2. Déconvolution Aveugle en Matlab.: Modèle2 .....                        | 126        |
| 8. Conclusions .....  | 128        |
| <b>Conclusion générale .....</b>  | <b>129</b> |
| <b>Annexe A .....</b>   | <b>132</b> |
| <b>Bibliographie .....</b>  | <b>135</b> |

## Résumé

À travers ce mémoire d'ingénieur, le problème de "Computational Photography" est abordé. L'objectif de surpasser les limites d'un appareil photographique ordinaire en utilisant des techniques informatiques permettant d'améliorer la qualité visuelle de l'image produite.

Deux solutions sont étudiées: la déconvolution avec ou sans régularisation, et la déconvolution aveugle.

Malgré les avantages qui ont suscité l'intérêt pour ces techniques, la déconvolution présente des inconvénients importants, tels que l'existence de problème inverse, la présence d'extrema locaux associés à des solutions parasites, et l'instabilité de système. Des méthodes de restauration sont donc à employer afin de pallier les défauts des dégradations, dans le but que l'observation traduise alors au mieux possible la scène originale de la réalité du terrain à analyser.

Dans nos expérimentations, nous nous sommes placés dans le contexte d'un appareil photographique embarqué sur une voiture ou sur un drone d'hélicoptère. Il va de soi que ces images doivent être d'une bonne qualité afin d'être exploitables. Or un tel appareillage monté sur aéronef est soumis à beaucoup d'aléas: climatiques, vibrations du moteur, problèmes de bouge durant l'acquisition photo, ..., etc.

Des dégradations se traduisant par des flous et des bruits sont donc parfois observées sur les images obtenues, réduisant leur qualité, et rendant plus délicate l'extraction d'informations pertinentes.

Dans ce travail, nous présentons une nouvelle méthode de restauration utilisant une paire d'images d'une observation unique (paire d'images prise avec des durées d'exposition différentes, par exemple), et basée sur le principe déconvolution totalement aveugle (Noyau de débrouillage inconnu) pour construire des images de haute qualité en combinant les deux images de qualité médiocre: la première brouillée (Slow Shutter Speed image) et la seconde bruitée (Fast Shutter Speed image).

Nous formulons le problème dans un cadre bayésien hiérarchique en utilisant les connaissances a priori sur l'image inconnue et le flou, et aussi sur la dépendance entre les deux images observées. En intégrant une analyse entièrement bayésienne, l'algorithme développé estime tous les paramètres utiles du modèle de sorte qu'aucune intervention de l'utilisateur n'est nécessaire.

De plus, nous utilisons une procédure d'inférence bayésienne variationnelle, qui permet la compensation statistique des erreurs qui se produisent à différentes étapes de la restauration, et fournit également des incertitudes sur les estimations.

Les résultats expérimentaux avec des images synthétiques et réelles montrent que la méthode proposée fournit des résultats de restauration à qualité très élevée et se compare

favorablement aux méthodes existantes, sans qu'aucune supervision utilisateur ne soit nécessaire.

Mots-clés : Restauration d'images, Méthodes bayésiennes, Déconvolution aveugle, Stabilisation d'image, hyper paramètres, Approximations variationnelles de distribution.

# Abstract

Through this memory engineering, the problem of "Computational Photography" is discussed. The goal is to overcome the limits of a traditional camera using informatics techniques to improve the visual quality of the image produced.

Two solutions are studied: deconvolution with or without regularization, and blind deconvolution.

Despite the benefits of these techniques, deconvolution criteria also present important drawbacks, such as the existence of the inverse problem, the presence of local extrema associated with parasites solutions, and system instability. So too many restoration methods are used in order to overcome the shortcomings of the damage, where the target is to contain an image like the original scene to can analyze.

In our experiments, we place in the context of a camera embarks on a car or a helicopter drone. It goes without saying that these images must be of good quality to be usable. But such equipment mounted on aircraft is subject to much riskiness: climate, engine vibration, moving the camera during the acquisition, etc...

Degradation resulting of blurring and noise are sometimes seen on the images obtained, reducing their quality and making it more difficult to extract suitable information.

In this framework, we present a new restoration method that uses a pair of images of a single observation as the pair of images is taken with different exposure times. Our contribution is to establish a method of blind déconvolution (Blurring kernel unknown) for the construction of high-quality images by combining two images taken under poor qualities, e.g. taken under different conditions: first blurred (slow shutter speed image) and the second noise ( fast shutter speed image).

We formulate the problem in a hierarchical Bayesian framework by using a prior knowledge on the unknown image and blur, and also on the dependency between the two observed images. By incorporating a fully Bayesian analysis, the developed algorithm estimates all necessary model parameters along with the unknown image and blur, such that no user-intervention is needed or required.

Moreover, we use a variational Bayesian inference procedure, which allows for the statistical compensation of errors occurring at different stages of the restoration, and also provides uncertainties of the estimates.

Experimental results with synthetic and real images demonstrate that the proposed method provides very high quality restoration results and compares favorably to existing methods even though no user supervision is needed.

Keywords: Image Restoration, Bayesian methods, blind deconvolution, Image Stabilization, hyper parameters, Approximations variational distribution.

## Introduction générale

Le traitement d'image connaît de nos jours un essor considérable. En effet, si d'une part les ordinateurs, téléviseurs, téléphones portables et autres produits de grande consommation, dont les enjeux commerciaux et économiques sont immenses, apportent de nouveaux défis, des domaines plus "fondamentaux" soulèvent eux aussi de nombreux problèmes. En effet, nous pouvons par exemple citer la médecine et l'utilisation des radiographies ou échographies, l'observation de l'infiniment petit, de l'infiniment grand... Dans chacun des cas précités, tout l'enjeu est de pouvoir extraire de l'image le maximum d'information. Cela nécessite généralement un traitement particulier: restauration, décomposition, segmentation, etc.

L'objectif essentiel de recouvrement d'image est de restaurer la scène originale dégradée par l'intermédiaire d'observations de qualité mauvaise.

Principalement le recouvrement d'image consiste en trois problèmes :

- a. Restauration d'image, où la dégradation due au système d'acquisition. Dans ce cas, le masque de dégradation est connu ou obtenu expérimentalement.
- b. Déconvolution aveugle, où seulement l'image observée est connue.
- c. Amélioration de la résolution d'image, où l'objectif est de restaurer une image à haute résolution à partir d'une ou de plusieurs images de résolutions plus basses.

En plus, de nouveaux problèmes de recouvrements sont étudiés face à l'avènement de nouvelles applications dans les domaines médical, du nano-image, de compétition photographique et de compacte sensation.

Ce projet vise à étudier, avec une orientation pratique (dans l'objectif d'une implantation sur FPGA), le problème de déconvolution aveugle d'images dégradées, et ce à partir de dernières méthodes proposées dans ce domaine.

Le sujet de la restauration vise, en général, à récupérer une image originale de bonne qualité, à partir d'une image dégradée (i.e. image bruitée et/ou floue). Dans le contexte de l'application pratique que nous avons choisie (images acquises par un appareil monté sur un support en mouvement), les images considérées souffrent d'un flou et/ou d'un bruit, ce qui peut rendre leur exploitation délicate, voire impossible. Dans l'extrême majorité des cas, ces opérations de brouillage sont modélisées par une convolution entre une image nette et des opérateurs de flou. Le but étant alors de pouvoir effectuer l'opération inverse, ce qui résulte en la dénomination couramment utilisée, et plus précise, de déconvolution.

Le qualificatif d'aveugle proviendra alors du cas particulier dans lequel l'opérateur de flou n'est pas explicitement connu, et doit également être estimé, rendant le problème autrement plus délicat.

Cette étude se justifie aussi par le fait qu'un grand nombre des approches proposées dans la littérature, utilisant des outils mathématiques très avancées, n'ont - à notre connaissance - pas

encore été sérieusement considérées dans un cadre applicatif.

## **Organisation du mémoire**

Les deux premiers chapitres constitueront l'introduction à notre thématique d'étude.

Le Chapitre 1 présente en détail le processus de formation des images observées, afin de comprendre comment celles-ci sont dégradées par le flou et par le bruit.

Le chapitre 2 comporte une revue des principales techniques de déconvolution existantes, d'un point de vue algorithmique, permettant de définir des outils qui nous seront utiles ultérieurement, et de situer notre travail par rapport à l'état de l'art, en particulier en ce qui concerne la déconvolution aveugle.

Le Chapitre 3 présente l'état de l'art actuel de déconvolution aveugle d'images. Nous présentons: les approches dites à identification du flou a priori, procédant en deux étapes découplées d'estimation de la dégradation, puis de restauration, et les approches à identification jointe, procédant de manière itérative et simultanée sur la reconstruction de l'opérateur de flou et de l'image restaurée.

Au chapitre 4, Nous proposons un nouvel algorithme de déconvolution, permettant de mettre en application les modèles proposés au chapitre précédent. Nous présentons différentes techniques de déconvolution aveugle d'images basé sur le principe de double régularisation à variation totale, c'est-à-dire par une technique variationnelle, soit de fonctionnelle à minimiser. Sous un modèle hiérarchique bayésien, exact et hiérarchique, mais également approche permettant d'effectuer un traitement rapide, voir temps réel. L'algorithme construit été hybride, permettant de combiner efficacement plusieurs approches différentes afin d'optimiser à la fois la qualité des résultats et le temps de calcul.

Dans la plupart des cas, il est nécessaire de connaître les dégradations subies par l'image observée. Lorsque celles-ci ne sont pas connues, nous effectuons une déconvolution dite aveugle. Pour cela nous estimons les paramètres instrumentaux grâce à un nouvel algorithme (hyperparamètres, cf. paragraphe IV.3.4), que nous avons développé en décrivant le spectre des images naturelles par un modèle fractal. Cet algorithme permet de résoudre un problème particulièrement difficile, dans de nombreux cas (notamment pour des images naturelles), et apporte une solution précise, là où les techniques classiques de déconvolution aveugle échouent.

Enfin, le chapitre 5 montre les résultats de déconvolution d'images dans plusieurs domaines applicatifs, comme l'imagerie au plein air, dans une chambre à mauvais éclairage et des images artificielles. Les résultats obtenus sont de très bonne qualité, autant lorsque les dégradations ne le sont pas connues.

Nous montrons la performance de notre algorithme proposé par effectuer plusieurs comparaisons avec des approches de l'état de l'art. En plus notre méthode mène à des résultats très souvent supérieurs, que cela soit visuellement, ou qualitativement par utilisation d'indicateurs standardisés.

Enfin, on présent par conclusion les résultats tirent dans ce travail vis-à-vis aux considérations pratiques envisagées. Pour le travail ultérieur, on donnera un certain nombre d'améliorations potentielles, ou d'autres pistes à creuser en perspectives de développement, qui peuvent mener à de haut performances (ex. : temps de calcul, environnement de simulation, informations connues a priori, forme du flou, etc.).

# Chapitre 1

## Généralités et problématique

Dans ce chapitre, Nous présentons dans le premier paragraphe les outils de base associés à la problématique de la restauration d'images. Nous commençons par des éléments de modélisation des images et le processus de formation des images floues. (Qui est en fait, processus résultat d'opération de convolution).

Ensuite, nous précisons les modèles de flous et d'images considérées, ceux-ci étant directement liés à l'application considérée.

Cette étape de modélisation est fondamentale si l'on souhaite une restauration efficace. Cette étape permet alors de mieux « guider » ultérieurement le processus aveugle dans sa recherche d'une solution possible.

### 1. Eléments de modélisation des images

Par modélisation d'une image, on entend une façon de décrire mathématiquement, et de trouver des voies pour analyser de telles images. Il s'agit donc d'une étape très importante en traitement des images, et tout particulièrement en déconvolution aveugle, où l'image et le flou que l'on vise à reconstruire doivent nécessairement être recherchés dans une classe de fonctions particulière, afin de borner l'espace de recherche, et surtout d'imposer à l'image fonction et au flou à estimer des caractéristiques particulières. On note qu'aucune modélisation n'est ultime, la « meilleure », et surtout pas parfaite, dans la mesure où, tout d'abord, une « bonne » représentation varie fortement d'une application particulière à une autre, et même d'une image représentative d'une scène bien spécifique à une autre. Surtout, il ne s'agit toujours dans tous les cas que d'une approximation, qui se révèle être plus ou moins éloignée de la réalité absolue, mais en aucun cas totalement concordante avec celle-ci.

C'est qui est évidente : La définition d'une méthode de restauration d'images soit modélisé la plus fiable possible du problème rencontré. Nous donnons donc ici des éléments communément admis de formation et de modélisation des images floues, que nous utiliserons par la suite.

Parmi l'ensemble des modèles, il en est un type qui sera utilisé tout au long de ce rapport, et qui est celui des représentations dites déterministes basées sur des espaces de fonctions. Celles-ci sont parmi les plus couramment utilisées dans la littérature scientifique. Précisons cependant qu'il en existe d'autres, telles que les représentations multi-échelles (par exemple par ondelettes), par ensemble de niveau, ou encore par champs aléatoires (Gibbs, Markov, etc.). Pour davantage d'informations concernant les différents modèles d'images pouvant être considérés, on peut consulter avec profit [1], dont nous nous inspirons pour ce qui suit.

## 1.1. Images en tant que distributions

De manière générale, concevoir une image comme une distribution est l'approche la plus répandue en modélisation déterministe. Commençons par fournir quelques éléments concernant les définitions des distributions, autrement appelées fonctions généralisées. Pour cela, notons  $\Omega$  un domaine 2D ouvert et borné. L'ensemble  $D$  des fonctions dites tests est défini, sur un espace  $\Omega$ , par

$$D(\Omega) = \{\phi \in C^\infty(\Omega) | \text{supp } \phi \subseteq \Omega\} \quad (1.1)$$

Chaque fonction test  $\phi$  peut être vue comme un système de capture linéaire de signaux d'image. Une image  $f$  sur  $\Omega$  est alors une fonction linéaire sur  $D(\Omega)$ :

$$f: \phi \rightarrow \langle f, \phi \rangle \quad (1.2)$$

L'espace image est alors l'espace des distributions, noté traditionnellement  $D'(\Omega)$ , qui est le dual topologique de  $D(\Omega)$ .

Bien qu'une telle image puisse ne pas sembler très familière, elle exhibe en fait bien une réponse  $\langle f, \phi \rangle$  pour toute fonction senseur  $\phi \in D(\Omega)$ , qui vise à percevoir l'existence et les caractéristiques de  $f$ . D'autre part, la perception est linéaire, et la notion de distribution est si souple que l'on peut alors considérer des dérivées de n'importe quel ordre.

Parmi les avantages de cette modélisation par fonctions généralisées figurent entre autre le fait que celles-ci permettent de représenter des images très diversifiées de nombreuses disciplines, et que leur définition semble bien en accord avec l'interprétation physique d'une image. En effet, une image n'existe pas en soi ; elle est plutôt acquise ou observée à travers sa réponse à un senseur, que ce soit le système de vision humain ou tout autre capteur technologique (caméra CCD par exemple).

En revanche, de telles images basées sur des distributions présentent aussi le défaut d'être si nombreuses, variées, que l'on peut alors difficilement en extraire quelque caractéristique bien particulière que ce soit. Des conditions de régularité bien spécifiques doivent alors être incorporées, afin de les modéliser de manière à ce qu'elles soient plus facilement utilisables en pratique.

## 1.2. Images $L^p$

Les images  $L^p$ , pour tout  $p \in [0, \infty]$ , portent une structure et contiennent une information bien plus forte que des images basées sur des distributions de manière générale. La définition des images  $L^p$  se rapporte naturellement à celle des espaces éponymes, qui est la suivante:

$$L^p(\Omega) = \{f: \int_\Omega |f|^p < \infty\} \quad (1.3)$$

A cet espace, on associe la norme

$$\|f\|_{L^p} = \left[ \int_{\Omega} |f|^p \right]^{\frac{1}{p}} \quad (1.4)$$

Les images  $L^p$  sont également des images de type distributions. On peut montrer que  $\forall q \geq p \geq 1$ , on a

$$L^q(\Omega) \subseteq L^p(\Omega) \quad (1.5)$$

On a donc, par exemple,  $L^2 \subseteq L^1$ . On constate alors, qu'en ce sens, les images  $L^1$  forment la classe la plus générale parmi les images  $L^p$ . Comme sous-ensemble, la classe des images  $L^2$  a formé un groupe couramment utilisé dans les tous premiers travaux en restauration d'images. Néanmoins, il est vite apparu que l'utilisation de la norme associée

$$\|f\|_{L^2} = \sqrt{\int_{\Omega} |f|^2} \quad (1.6)$$

amenait en restauration à des images pas suffisamment régulières, c'est-à-dire simplement pas suffisamment lissées, contenant en général trop de bruit.

### 1.3. Images de Sobolev $H^n$

Pour pallier le défaut de régularité des images  $L^2$ , sont introduites les images basées sur les espaces dits de Sobolev [2]. Commençons ici par donner quelques définitions concernant ces espaces. Pour tout  $n \in \mathbb{N}$  et pour tout  $1 \leq p < \infty$ , on définit l'espace de Sobolev  $W^{n,p}(\Omega)$

$$W^{n,p}(\Omega) = \{f \in L^p(\Omega) | \forall \alpha, |\alpha| \leq n, \exists v_{\alpha} \in L^p(\Omega), v_{\alpha} = \partial^{\alpha} f \text{ au sens faible}\}, \quad (1.7)$$

où  $\partial^{\alpha}$  représente la dérivation d'ordre  $\alpha$ .

On se limite le plus souvent dans la définition présente au cas où  $p = 2$ . On a alors la définition suivante des espaces  $H^n = W^{n,2}$ :

$$H^n(\Omega) = \{f \in L^2(\Omega) | \forall \alpha, |\alpha| \leq n, \exists v_{\alpha} \in L^2(\Omega), v_{\alpha} = \partial^{\alpha} f \text{ au sens faible}\}, \quad (1.8)$$

Ces espaces présentent l'intérêt d'être en fait des espaces de Hilbert (d'où la notation H), c'est-à-dire des espaces spécifiques dont la norme découle d'un produit scalaire. On introduit alors sur  $H^n$  le produit scalaire

$$\langle f, v \rangle_n = \sum_{|\alpha| \leq n} \partial^{\alpha} f \partial^{\alpha} v \quad (1.9)$$

et la norme associée  $\|f\|_n = \sqrt{\langle f, f \rangle_n}$  (où  $\langle ., . \rangle$  désigne le produit scalaire usuel

sur  $L^2(\Omega)$ ).

Parmi ces espaces  $H^n$ , l'espace  $H^1$  a rapidement supplanté  $L^2$  pour la modélisation d'images en restauration. Une image est dite « Sobolev  $H^1$  » si ses dérivées au premier ordre (au sens des distributions) appartiennent également à  $L^2$ , de même, ou de manière équivalente, si son gradient  $\nabla f$  appartient à  $L^2 \times L^2$ . Sa norme associée est alors

$$\|f\|_1 = \|f\|_{H^1} = \sqrt{\langle f, f \rangle_1} = \sqrt{\int_{\Omega} |f|^2 + \int_{\Omega} |\Delta f|^2} \quad (1.10)$$

On peut également écrire que

$$\|f\|_{H^1}^2 = \|f\|_{L^2}^2 + \|\nabla f\|_{L^2}^2 \quad (1.11)$$

La norme  $H^1$  n'est cependant pas communément utilisée sous ces deux dernières formes, et seul le terme

$$\|\nabla f\|_{L^2} = \sqrt{\int_{\Omega} |\Delta f|^2} \quad (1.12)$$

est en général employé pour mesurer l'information (au sens de la régularité) contenue dans une image  $H^1$ . Ceci cependant, par abus de langage, souvent sous le nom de norme  $H^1$ , alors qu'il s'agit en toute rigueur de la semi-norme<sup>1</sup> sur  $H^1$ .

#### 1.4. Images à variations bornées.

Des distributions globales aux images  $L^p$ , puis aux images de Sobolev  $H^n$ , nous avons à chaque fois plus de régularité ajoutée. Cette suite d'espaces devient donc plus spécifique, et ainsi plus facilement appréhendable et utilisable en pratique. Il y a cependant un compromis à trouver entre les propriétés de régularité et la fidélité de la modélisation des images. Une solution adéquate est de disposer d'un espace qui fournit un équilibre entre ces deux aspects, c'est-à-dire qui soit mathématiquement « malléable », et pratiquement fidèle dans la représentation des caractéristiques essentielles des images générales.

Ceci est vérifié par l'espace à variations bornées BV (angl. Bounded variations), dont le rôle en traitement d'images a pris une importance notable ces dernières années.

Du point de vue des distributions, une image  $f \in L^1$  appartient à  $BV(\Omega)$  si et seulement si son gradient au sens des distributions  $\nabla f$  (parfois écrit  $Df$ ) vérifie

<sup>1</sup> C'est-à-dire une application disposant de presque toutes les propriétés lui conférant le statut de norme, sauf celle de séparation, qui assure que la norme d'un vecteur est nulle seulement si ce vecteur est nul.

$$\sup_{\phi \in D(\Omega); \|\phi\|_{\infty} \leq 1} \langle \nabla, \phi \rangle < \infty \quad (1.13)$$

ou  $\phi = (\phi_1 + \phi_2) \in D(\Omega) \times D(\Omega)$  et  $\|\phi\|_{\infty} = \sup_{\Omega} (\phi_1^2 + \phi_2^2)^{\frac{1}{2}}$ .

Si c'est le cas, alors ce supremum est appelé « variation totale » (TV: total variation) de  $f$ , et que l'on notera parfois plus loin  $TV(f)$ .

Dans le cas où  $f$  est une fonction différentiable,  $TV(f)$  peut être définie via une intégrale basée sur la fonction donnée, au lieu du supremum de (1.13) :

$$TV(f) = \int_{\Omega} |\nabla f| \quad (1.14)$$

Parmi les propriétés remarquables de l'espace BV figure le fait que toutes les images  $W^{1,1}$  sont aussi des images BV, mais que seulement ces dernières permettent des discontinuités, c'est-à-dire des sauts d'intensité. Ce besoin de représentation des discontinuités d'une image, de ses bords, est devenu crucial dans les recherches actuelles, et même incontournable dans certaines thématiques. D'autre part, bien que les images  $L^p$  autorisent aussi la présence de discontinuités, elles ne conservent pas les irrégularités oscillatoires locales, aux différences des modèles BV.

## 2. Modélisation des images floues

Après nous être attachés à donner des modèles d'images claires, nettes, intéressons-nous à la modélisation du processus d'obtention d'images dégradées par un noyau de flou.

Nous commencerons par donner le modèle de formation des images monochromes, modélisé le plus couramment utilisé dans la littérature scientifique.

### 2.1. Images à niveaux de gris

De manière la plus générale, une image continue  $g$  obtenue à travers un système, par exemple optique, possédant une fonction de dispersion (angl. Point Spread Function « PSF »)  $h$  variante suivant les coordonnées spatiales  $x$  et  $y$ , et dépendant de l'image source  $f$ , peut être donnée par

$$g(x, y) = [h(x, y; s, t, f) * f(x, y)] \diamond b(x, y) \quad (1.15)$$

$$= \left[ \int h(x - s, y - t; s, t, f) f(x - s, y - t) ds dt \right] \diamond b(x, y) \quad (1.16)$$

Où  $*$  est utilisé ici pour définir une convolution spatialement variante,  $\diamond$  un opérateur générique soit d'addition soit de multiplication,  $b$  un bruit éventuellement présent dans l'image observée, et  $s$  et  $t$  les coordonnées spatiales. Ici, le flou  $h$  est donc dépendant de l'image originale  $f$ .

Cependant, cette dépendance du flou  $h$  avec l'objet  $f$  observé n'est en général pas

fondamentale et peut être négligée. On arrive alors à l'intégrale de superposition [3]

$$g(x, y) = [h(x, y; s, t, f) * f(x, y)] \diamond b(x, y) \quad (1.17)$$

$$= \left[ \int h(x - s, y - t; s, t) f(x - s, y - t) ds dt \right] \diamond b(x, y) \quad (1.18)$$

De même, une hypothèse de variance spatiale du flou est seulement induite par la présence de phénomènes particuliers, tels que mouvement rotatif, et autres. Nous avons pour cette raison choisi de l'éliminer, dans la mesure où ce type de dégradation ne nous semble pas pouvoir être fréquent dans le cadre de l'application visée. Mais également pour des questions de simplicité, étant donné le caractère très délicat que présente un problème aveugle, avec de tels noyaux. Le modèle de formation d'images résultant est alors beaucoup plus simple à traiter et se réduit à

$$g(x, y) = [h(x, y) * f(x, y)] \diamond b(x, y) \quad (1.19)$$

$$= \left[ \int h(x, y) f(x - s, y - t) ds dt \right] \diamond b(x, y) \quad (1.20)$$

$$= \left[ \int h(x - s, y - t) f(x, y) ds dt \right] \diamond b(x, y) \quad (1.21)$$

où \* est l'opérateur usuel de convolution.

Pour une image  $f$  numérisée, qui est le cas qui nous intéresse ( $f, g$  et  $h$  étant alors des matrices), nous obtenons alors des relations précédentes le modèle convolutif discret suivant :

$$g(x, y) = [h(x, y) * f(x, y)] \diamond b(x, y) \quad (1.22)$$

$$= \left[ \sum_k \sum_l h(x - k, y - l) f(k, l) \right] \diamond b(x, y) \quad (1.23)$$

$$= \left[ \sum_k \sum_l h(k, l) f(x - k, y - l) \right] \diamond b(x, y) \quad (1.24)$$

On peut réordonner lexico-graphiquement les éléments de  $f$  dans un vecteur  $f$ , et la relation (1.24) peut alors se présenter sous la forme d'un produit matrice vecteur, souvent plus simple à manipuler, dans lequel  $h$  prend la forme d'une matrice de Toeplitz par blocs avec blocs de Toeplitz (abrégé BTTB par la suite)  $H$ :

$$g = H \cdot f \diamond b \quad (1.25)$$

## 2.2. Modélisation du bruit présent

En ce qui concerne les caractéristiques du bruit d'observation, nous le considérons de bruit additif, car c'est le type le plus communément rencontré dans les systèmes optiques de type appareil photographique ou caméra vidéo. L'équation de modélisation retenue pour notre

modèle sera donc, dans le cas monocanal :

$$g(x, y) = h(x, y) * f(x, y) + b(x, y) \quad (1.26)$$

$$= \sum_k \sum_l h(x - k, y - l) f(k, l) + b(x, y) \quad (1.27)$$

ou, sous forme d'un produit matrice-vecteur

$$g = Hf + b \quad (1.28)$$

La distribution du bruit peut être considérée comme gaussienne (Voir [4] pour plus de détails). Une étude approfondie du système imageur utilisé peut estimer les paramètres de manière plus approfondie, mais ceci n'est pas fondamental ici.

### 3. Modélisation des flous

Portons maintenant attention au flou  $h$ . Deux types principaux peuvent être rencontrés dans les situations de prises d'images de l'application qui nous intéresse : flou de mouvement, dû au bougé pendant la phase d'acquisition, et ou de défocalisation, dû à un mauvais réglage de la focale de l'appareil utilisé. Une difficulté qui peut de même être rencontrée est que ces deux types de flou peuvent par moment affecter simultanément les images acquises.

Ceci est un obstacle important, résultant en des noyaux très généraux, et qui est rarement traité dans la littérature scientifique. En posant que  $h = h_1 * h_2$ , notre modèle (1.28) s'écrirait alors de manière plus précise

$$g = h_1 * h_2 * f + b \quad (1.29)$$

Le flou de mouvement sans autre contrainte n'est pas caractérisable autrement que par une courbe paramétrique. En revanche, si l'on se ramène à un modèle uniforme linéaire (parfois valable en première approximation),  $h_2 = h_2(l, \theta)$  dépend alors de deux paramètres : orientation  $\theta$ , et longueur du support  $l$ . Le noyau de défocalisation (parfaite)  $h_1 = h_1(\rho)$  est lui toujours paramétrable et dépend d'un paramètre unique  $\rho$ .

#### 3.1. Flou de défocalisation

Le flou de défocalisation est un opérateur transformant l'image d'un point en un disque de rayon  $\rho$ . Il est déterminé par un unique paramètre  $\rho$ , son rayon, et est donné par

$$h_1(\rho) = \begin{cases} \frac{1}{\pi\rho^2} & \text{si } \sqrt{x^2 + y^2} \leq \rho \\ 0 & \text{ailleurs} \end{cases} \quad (1.30)$$

Avec  $x$  et  $y$  sont les coordonnées spatiales d'une image obtenue.

### 3.2. Flou de mouvement rectiligne

La fonction de dispersion (ou réponse impulsionnelle) d'un tel noyau dépend, comme nous l'avons dit, de deux paramètres, sa longueur  $l$  et son orientation  $\theta$ , et s'écrit

$$h_2(l, \theta) = \begin{cases} \frac{1}{l} & \text{si } \sqrt{x^2 + y^2} \leq \frac{l}{2} \text{ et } \tan \theta = \frac{y}{x} \\ 0 & \text{ailleurs} \end{cases} \quad (1.31)$$

Avec  $x$  et  $y$  sont les coordonnées spatiales d'une image obtenue.

Ce noyau convolué avec l'image d'un point résulte en sortie en un segment de droite, de longueur  $l$  et d'angle  $\theta$ . Mais comme nous l'avons dit, le flou de bougé n'est pas exclusivement rectiligne uniforme (c'est au contraire plutôt un cas restrictif de la réalité physique), et le cas général serait plutôt de considérer une ligne courbe avec valeurs variables le long de celle-ci, ce qui est néanmoins bien plus délicat à traiter pour des problèmes de déconvolution.

### 4. Conclusion

Dans ce chapitre, nous présentons les outils des bases d'un tel processus de traitement d'image.

Où il faut noter qu'en général, la plupart des expérimentations effectuées dans la littérature consacrée au problème de déconvolution aveugle, utilisent des noyaux de flou très petits.

Ceci pour deux raisons, qui, à notre avis, impliquent des conditions très favorables :

- ❖ Inversion du noyau mieux conditionnée.
- ❖ Problème plus simple de par la faible quantité d'information perdue dans l'image.

En plus globalement, il apparaît clairement que déflouer en aveugle de petits noyaux, comme cela est fait dans la grande majorité des travaux publiés, est un cas tout à fait favorable d'une part, et de l'autre peu représentatif de l'ensemble des problèmes réels susceptibles de se poser dans des applications pratiques.

## Chapitre 2

### État de l'art de la Déconvolution

On présente ici la problématique fondamentale des problèmes inverses, dont la déconvolution est un cas particulier. L'équation de modélisation retenue pour notre modèle sera (1.28) sous forme d'un produit matrice-vecteur ou sous forme produit de convolution

$$g = Hf + b$$

$$g = h * f + b \quad (1.28)$$

En effet, on pourrait naïvement penser que le fait de disposer de la connaissance du noyau de flou affectant une image est suffisante pour pouvoir « annihiler » ce défaut, en effectuant une opération inverse. Il n'en n'est en fait rien, et la restauration, aveugle ou non, s'avère un phénomène instable, très facilement perturbable, et que l'on appelle « problème mal-posé ».

Après avoir posé le problème de la déconvolution, nous effectuons une classification algorithmique de l'état d'art sur ce sujet. Les méthodes présentées cherchent pour la plupart à reconstruire une solution qui présente une certaine régularité. Celle-ci peut être posée de différentes manières, et réalisée au moyen d'approches monoéchelle ou multiéchelle, par des algorithmes directs ou itératifs, linéaires ou non linéaires, déterministes ou stochastiques.

#### 1. Déconvolution: un problème inverse mal posé

La déconvolution consiste à reconstruire une fonction, représentant la scène observée, à partir d'une transformation intégrale (une convolution dans le cas présent) suivie d'une discrétisation.

Il s'agit donc d'un problème inverse. Comme le processus d'observation est stochastique (l'observation étant bruitée), le problème appartient à la classe des problèmes inverses mal posés au sens de Hadamard [5], car :

- ❖ la solution peut ne pas exister.
- ❖ elle n'est pas nécessairement unique.
- ❖ elle peut être instable: une faible incertitude sur les données entraîne une forte incertitude sur l'image reconstruite.

Comme les données observées sont discrètes, nous ne cherchons pas ici à reconstruire une fonction à support continu, mais une fonction discrète – et qui comporte le même nombre d'inconnues que de données, autrement dit nous gardons la même résolution. Cela ne résout pas pour autant notre problème, mais permet de le reformuler sous forme d'inversion d'un opérateur discret – et l'on parle alors de problème discret mal posé.

Même s'il existe un inverse, la solution reconstruite par inversion n'est pas stable, car cet inverse possède des valeurs propres très grandes, ce qui a pour effet d'amplifier considérablement le bruit d'observation. L'incertitude sur la solution est en général

inacceptable, même si l'on augmente autant que possible la précision des observations. Il est alors nécessaire de régulariser le problème en contraignant la solution, soit en la forçant à appartenir à un certain ensemble, soit en lui attribuant une densité de probabilité donnée, tout cela traduisant les connaissances a priori dont on dispose. Cela permet de stabiliser la solution et de reconstruire une image physiquement acceptable, car on lui impose de vérifier certaines propriétés, par exemple la régularité ou la positivité. Ce type d'approche a donné naissance à un grand nombre d'algorithmes, dont nous allons présenter une classification dans les paragraphes suivants. On peut trouver une revue des principaux algorithmes dans [6, 7, 8, 9, 10], et concernant plus particulièrement les images astronomiques dans [11].

Nous considérons d'abord que le noyau de convolution  $h$  (ou la FTM correspondante) est connu (ses coefficients sont donnés, ou les paramètres instrumentaux sont connus, ainsi que le modèle de dégradation). Ensuite, nous abordons un cas plus général, celui de la déconvolution aveugle (cf. chapitre 3).

## 2. Méthodes monoéchelle

Lorsque la notion d'échelle n'est pas utilisée de manière explicite dans un algorithme, nous parlerons de méthode monoéchelle. Les méthodes de déconvolution les plus courantes appartiennent à cet ensemble.

### - Filtrage linéaire, en une seule passe:

I - Filtrage inverse, pseudo-inverse et Wiener. Le filtre de Wiener est l'estimateur qui minimise l'erreur quadratique moyenne (MMSE).

II - DVS tronquée. La décomposition en valeurs singulières (DVS) tronquée permet, de limiter l'amplification du bruit.

III - Filtres RIF. Le filtrage peut être effectué dans l'espace image au moyen d'une convolution par un filtre à réponse impulsionnelle finie (RIF)).

### - Filtrage récursif et semi-récursif

Les filtres récursifs permettent de réaliser des réponses impulsionnelles infinies à partir de filtres de support spatial fini. Ces techniques sont particulièrement utiles lorsque le modèle d'observation ou d'image varie spatialement. Le plus célèbre est: Filtre de Kalman [12].

### - Techniques itératives semi-convergentes

On appelle méthode semi-convergente toute méthode de restauration qui converge en l'absence de bruit et diverge en présence de bruit.

Nous décrivons maintenant les techniques itératives, nécessitant l'application d'un filtre de manière récursive sur un grand nombre d'itérations pour devenir une solution au problème de déconvolution.

#### 1- Méthodes sans contraintes.

Une des techniques les plus simples est la méthode de Landweber [13], algorithme de type "Expectation-Maximization"(EM) a été introduit dans [14]. Indépendamment, la même méthode a été définie dans [15], portant le nom d'algorithme de Richardson-Lucy, très populaire en Astronomie.

#### 2- Méthodes avec contraintes.

Il est parfois très utile d'incorporer des contraintes lors de la recherche itérative de solution au maximum de vraisemblance, et ceci quel que soit le modèle de bruit utilisé [6]. On distingue essentiellement deux types de contraintes: les contraintes sur  $X$ , et les contraintes sur le résidu  $Y-HX$ . Ces contraintes sont définies soit dans l'espace image, soit dans l'espace fréquentiel.

#### 3- Approches itératives par EDP

Les équations aux dérivées partielles (EDP) ont trouvé récemment une application à la déconvolution des images. Utilisées auparavant pour le débruitage, elles permettent, moyennant quelques améliorations, de traiter aussi le cas des images floues. L'idée principale est d'effectuer une diffusion afin de lisser le bruit qui tend à apparaître lors de l'opération d'inversion. Le plus simple consiste à effectuer une diffusion isotrope, cela a pour effet de lisser l'image (donc le bruit) dans toutes les directions. Le lissage du bruit se fait, dans ce cas, au détriment des contours. La combinaison avec un terme d'attache aux données permet alors de construire une méthode itérative dans laquelle la diffusion joue le rôle de régularisation. Ce cas étant équivalent à la méthode de Tikhonov [9].

#### 4- Algorithmes de régularisation déterministes

##### Principe

L'approche générale, consistant à rechercher une solution lisse approchée compatible avec l'équation d'observation, a été proposée dans [16], sous le nom de régularisation.

L'idée de base de la régularisation des problèmes mal posés est de considérer une famille de solutions approchées qui dépendent d'un paramètre positif, appelé paramètre de régularisation.

La recherche d'une solution approchée au lieu de la solution inverse permet non seulement d'imposer des propriétés de régularité évitant ainsi l'amplification du bruit, mais aussi de stabiliser le problème: une solution unique et stable existe, sous certaines conditions.

## Méthodes linéaires

Les algorithmes de régularisation font appel soit à techniques de filtrage linéaire, soit à méthodes itératives. Lorsque la fonction à minimiser est quadratique, et lorsque les opérateurs intervenant dans cette fonctionnelle sont circulants par blocs (convolution, dérivation, etc.) si l'on considère une extrapolation de l'image périodique alors la minimisation est effectuée par filtrage linéaire dans l'espace de Fourier. C'est le cas notamment de la régularisation de Tikhonov [9].

On note l'opérateur linéaire correspondant au gradient  $\nabla$ , au laplacien  $\Delta$  ou encore au Hessien  $\nabla^2$ .

## Méthodes non linéaires

Lorsque l'on cherche à préserver les discontinuités (grandes variations en discret) de l'image que l'on reconstruit, les techniques de régularisation classiques comme celle de Tikhonov [16] échouent. De nombreux travaux ont été effectués à partir des années 90 pour définir des termes de régularisation non quadratiques, l'idée de base étant de ne plus utiliser la norme qui introduit des contraintes globalement, mais de préserver les discontinuités là où elles existent tout en lissant ailleurs.

Naturellement, n'importe quel algorithme proposé doit également vérifier des conditions particulières pour permettre la convergence. Notamment l'évolution de la forme des critères doit être suffisamment lente pour éviter de tomber dans un minimum local.

Cela se traduit par un allongement considérable du temps de calcul par rapport aux techniques citées précédemment. On entre alors peu à peu dans le monde des algorithmes coûteux en temps de calcul, mais efficaces en terme de convergence, celui des méthodes stochastiques.

### - Algorithmes stochastiques

Au lieu d'imposer des contraintes sous forme de projections, ou de régulariser le problème en ajoutant un terme supplémentaire au critère des moindres carrés, on peut travailler en terme de probabilités et utiliser l'approche bayésienne. La densité de probabilité  $P(X)$  correspond à la connaissance a priori de l'image que l'on veut obtenir. C'est ici que l'on introduit les contraintes sur les contours et les zones homogènes de l'image, autrement dit la régularisation, mais cette fois-ci sous forme stochastique (une image n'ayant pas les propriétés de régularité désirées est peu probable).

Le modèle a priori étant défini, il s'agit maintenant d'exprimer, toujours en terme de probabilité, l'image originale  $X$  en fonction de l'observation  $Y$ . Cela se traduit par la densité de probabilité a posteriori. La loi de Bayes permet de l'exprimer :

$$P\left(\frac{X}{Y}\right) \propto P\left(\frac{Y}{X}\right) P(X) \quad (2.1)$$

Sans entrer dans la théorie des estimateurs (qui sera abordée le chapitre ultérieur), on peut

mentionner plusieurs manières d'obtenir  $\tilde{X}$  à partir de cette loi a posteriori. L'une des plus répandues est de calculer le maximum a posteriori (MAP). On remarque que la densité de probabilité a la forme  $e^{-U(X)/2\sigma^2}$  : maximiser cette densité équivaut à minimiser le critère discret  $U(X)$ . La régularisation variationnelle (formulée en discret) et la régularisation stochastique se rejoignent donc, correspondant au même optimum, mais à des algorithmes pouvant être très différents.

L'inconvénient majeur de ces méthodes est leur lenteur qui les rend inutilisables lorsque le support du noyau de convolution est grand (c'est le cas même pour des noyaux  $11 \times 11$ ). D'autres algorithmes permettent de contourner cette difficulté en travaillant de manière alternée dans l'espace de Fourier (pour diagonaliser l'opérateur de convolution) et dans l'espace image (pour incorporer le modèle a priori).

Une autre manière d'estimer l'image inconnue est d'utiliser l'estimateur du MPM (mode des marginales a posteriori) [17], qui donne souvent de meilleurs résultats que le MAP et plus rapidement.

L'approche stochastique fournit un cadre théorique permettant l'estimation des paramètres du modèle a priori. Lorsque l'on calcule le MAP et que le paramètre inconnu  $\phi$  est défini de la même manière qu'en régularisation fonctionnelle, ce cadre permet l'estimation des paramètres des modèles variationnels. Il est donc possible de rendre automatiques les algorithmes fondés sur toutes ces approches, en associant à la recherche de la solution  $\tilde{X}$  l'estimation des paramètres du modèle. Souvent, l'estimation et la restauration sont réalisées de manière conjointe, alternant étape de déconvolution et étape de mise à jour des paramètres. C'est le cas notamment de la méthode EM, utilisée selon différentes variantes dans [6,18]. On peut encore citer [19, 20].

### **3. Méthodes multiéchelles**

On appelle méthode multiéchelle toute méthode mettant en jeu de manière explicite soit les approximations, soit les détails, soit tous les éléments, d'une décomposition multirésolution. Les approximations et détails multirésolution sont calculés, pour des images à support continu, par une transformée en ondelettes [21].

Semblable au cas monoéchelle, la technique de déconvolution multiéchelle repose tous sur l'utilisation d'algorithmes de décomposition comme ceux cités précédemment, leur différence réside dans la manière d'utiliser les coefficients de la décomposition et parfois dans le choix de l'algorithme d'analyse.

Parmi toutes les techniques citées ci-dessus, seule une partie d'entre elles ont été appliquées à la déconvolution, la voie est ainsi ouverte pour proposer de nouveaux algorithmes. D'autre part, comme nous le verrons dans les chapitres suivants, il est nécessaire d'améliorer les modèles existants car peu d'entre eux permettent une reconstruction correcte des images aériennes prisées ou satellitaires, la plupart des approches n'étant pas adaptées à ce type d'images, fortement non stationnaires, et contenant de nombreux détails très fins et orientés.

Pour terminer l'état de l'art en matière de déconvolution, nous allons dans la suite de ce chapitre faire un tour sur un autre type de déconvolution dite aveugle.

#### **4. Conclusion**

Dans ce chapitre, nous sommes présentés le processus de déconvolution et les principes méthodes utilisent de l'art. Où nous voudrions donner des informations le plus complet possible, sans pouvoir être très exhaustif.

Parmi tous les algorithmes proposés dans la littérature de la déconvolution, il n'en existe aucun qui puisse être appliqué avec succès sur des images des scènes quelconques. À part la méthode décrite au ce chapitre aucune technique ne peut fournir, de manière fiable et précise, une estimation de la PSF ou du noyau de convolution.

Il est donc nécessaire de formuler de nouvelles approches, tenant compte de la spécialité des données à traiter. Et ce qui discute dans le chapitre suivant : déconvolution aveugle.

## Chapitre 3

# État de l'art en Déconvolution Aveugle

Dans ce chapitre, nous allons présenter l'ensemble des techniques existantes en déconvolution aveugle d'images, suivant leurs différentes approches. Étant donné que le caractère aveugle de ces méthodes de restauration peut être vu comme une extension des techniques de restauration « simples », ou « classiques », c'est-à-dire à flou connu, nous allons voir qu'il est nécessaire de donner également un aperçu de ces dernières approches, en plus de celles bien spécifiquement axées sur l'approche aveugle. En effet, nombre d'algorithmes, qu'ils soient récents ou plus anciens, se basent en fait sur des techniques « simples », auxquelles sont adjointes des procédures plus spécifiques d'identification de la dégradation, afin de pallier le caractère aveugle du problème.

### 1. Introduction

On peut classifier de manière générale les approches de déconvolution aveugle en deux catégories, suivant la façon dont on va identifier le flou: soit a priori, c'est-à-dire de manière disjointe à la procédure de restauration, préalablement à celle-ci, et indépendamment, soit de manière jointe à cette même étape de restauration de l'image, la plupart du temps dans le cadre d'une méthode itérative, alternant entre l'estimation du noyau de flou et de l'image restaurée, jusqu'à une convergence escomptée.

#### Méthodes avec identification du flou a priori.

Ces méthodes visent à estimer la réponse impulsionnelle (angl. PSF « Point Spread Function ») à partir de l'image dégradée uniquement. Cette estimation est ensuite utilisée dans un schéma d'inversion classique. Les premières techniques de déconvolution aveugles se sont basées sur cette approche [22, 23, 24], en partie en raison de leur grande simplicité, et donc aussi de leur relative rapidité, à une époque où le calcul numérique était encore balbutiant. Cette approche a donc pour avantage sa rapidité, mais présente de l'autre côté le défaut d'être de fonctionnement assez aléatoire en pratique. En effet, si l'estimation obtenue à l'identification n'est pas de qualité suffisante, alors le processus de déconvolution de l'image ne pourra qu'être médiocre, sans aucune possibilité de correction a posteriori. Notons que les meilleurs résultats obtenus avec ce genre de techniques le sont dans les cas de modèles de flous paramétriques, pour lesquels on considère alors connue la classe paramétrique du noyau, ce qui contraint donc fortement leur estimation. Notons enfin que certaines méthodes plus récentes, et qui sont assimilables à cette catégorie proposent d'estimer directement l'inverse de la dégradation. L'ensemble de ces approches est examiné dans le paragraphe 2.

#### Méthodes d'identification jointe.

La plupart des méthodes actuelles rentrent dans cette deuxième catégorie. L'image restaurée et le flou sont estimés simultanément, même si cette appellation est un peu impropre, car dans les faits, le processus est très souvent alterné sur ces deux données. Ceci en raison du fait que le calcul d'une solution jointe est souvent problématique, par exemple à cause de difficultés posées par la minimisation de fonctionnels non convexes. Ces méthodes d'identification

jointe sont algorithmiquement bien plus complexes, nécessitant de plus longs temps de calculs, mais sont mieux à même de traiter des problèmes généraux avec des noyaux non-paramétriques, dans lesquels on cherche à reconstruire tout un opérateur de flou (soit une surface 2D), en raison de leur plus grande robustesse. Ces méthodes feront l'objet du paragraphe 3. Et bien sur notre algorithme dans le chapitre 4.

## 2. Méthodes avec identification du flou a priori

Nous donnons dans cette section un aperçu des méthodes existantes dans le domaine de la déconvolution aveugle par identification a priori du flou. Ces méthodes reposent principalement sur une estimation préalable du flou à partir de la seule connaissance de l'image observée, puis d'une étape d'inversion de ce flou estimé, ceci par une des innombrables méthodes existantes pour la restauration « classique » d'images, c'est-à-dire à flou connu.

En bref, ces méthodes contiennent donc globalement deux étapes :

- ❖ Estimation préalable du flou.
- ❖ Restauration de l'image avec flou connu.

Le point essentiel se situe de manière claire dans la première étape d'estimation, car c'est la qualité de cette estimée obtenue qui conditionnera la bonne marche ou non du processus de restauration par inversion consécutif. Notons que c'est ce genre d'approches qui a été utilisé dans les premières techniques de déconvolution d'images telles que [23] et [24].

En ce qui concerne la deuxième étape de restauration à partir du flou estimé, nous donnerons principalement les techniques générales parmi les très nombreuses méthodes existantes, car une énumération exhaustive ne relèverait pas de notre propos.

### 2.1. Estimation préalable du flou

Comme nous l'avons dit plus haut, les méthodes initiales de déconvolution aveugle d'images rentrent dans cette catégorie. Leur grande simplicité et donc leur rapidité étaient idéales du point de vue des besoins (et des possibilités) en capacité de calcul. En revanche, leur robustesse n'était en général pas avérée.

Un certain nombre des approches ici présentées procèdent en deux sous-étapes: estimation de la taille du support, puis calcul des coefficients. Si l'estimation de la taille du support du flou diffère de sa taille réelle, les coefficients ne peuvent pas être estimés de manière fiable. Cette estimation préalable du flou permet dans certains cas précis d'arriver à de bien meilleurs résultats que ne le fait une procédure de déconvolution à identification jointe, et plus rapidement ; mais bien entendu uniquement dans le cas d'une estimation suffisamment exacte.

Plusieurs parmi ces méthodes sont conçues pour être utilisées avec un type de flou à chaque

fois bien particulier, cela pouvant être un noyau gaussien, de mouvement, de défocalisation, gaussien mixte, etc. Cependant, le point négatif de ces approches est que, dans leur grande majorité, elle ne peuvent fonctionner qu'avec des modèles de flou paramétriques, ce qui exclut donc a priori la plupart des images réelles observées en pratique, pour lesquelles le flou ne peut pas être forcément toujours modélisée de la sorte de manière fiable. En effet, celui-ci n'est alors dans ce cas que plus ou moins proche d'un modèle paramétrique.

Précisons que ce type d'approches est en pratique majoritairement utilisé dans le cadre d'une démarche expérimentale: on collecte une ou plusieurs images d'un point source, et on en déduit la fonction de dispersion du système imageur. C'est par exemple ce qui a été fait dans le cas du télescope spatial Hubble. De même, si la physique du système imageur peut être modélisée de façon précise, alors on peut en déduire une estimation a priori du flou qui affectera les images issues de ce système. Ceci peut par exemple être fait en microscopie, télédétection, etc.

Parmi les défauts inhérents à ces méthodes d'estimation a priori, on pourra citer le manque de réalisme du modèle d'image sous-jacent, souvent basé sur la réalisation d'un processus aléatoire. En outre, quand la scène contient des bords nets (ce qui est quasi systématiquement le cas), ou des textures variant spatialement, ces modèles ne permettent pas une estimation efficace du noyau de flou. En effet, pour que ces méthodes fonctionnent de manière efficace, il faut absolument que la modélisation globale (du flou et de l'image) soit suffisamment fiable.

Les méthodes que nous allons présenter ci-dessous ont été regroupées en 5 catégories. Celles-ci, cependant, ne sont pas totalement hermétiques les unes aux autres, et il peut arriver que certaines soient assimilables à l'une ou l'autre catégorie que celle à laquelle nous l'avons associée. Ces catégories seront les :

- ❖ Méthodes basées sur la modélisation ARMA.
- ❖ Méthodes basées sur les caractéristiques spectrales de l'image.
- ❖ Méthodes basées sur l'autocorrélation
- ❖ Méthodes basées sur la transformée de Radon.
- ❖ Autres méthodes.

### **2.1.1. Méthodes basées sur la modélisation ARMA**

Ces approches se basent sur la modélisation ARMA (angl. Auto Regressive Moving Average) des images floues. Cette modélisation repose sur deux processus: un processus AR pour le modèle d'image, et un processus MA (angl. Moving Average) pour le modèle de flou, ce qui permet de distinguer les effets du flou de ceux de l'image. En effet, les images floues exhibent généralement les caractéristiques d'un lissage passe-bas, d'où l'intérêt du processus MA 2D utilisé pour représenter cet effet. Le processus MA identifié est alors pris pour être la description de la fonction de dispersion du système imageur. Le problème d'identification du flou affectant une image revient donc à l'identification d'un modèle ARMA 2D.

La modélisation de l'image discrète originale (AR 2D), est donnée par l'équation

$$f(x, y) = \sum_{k,l \in R_a; (k,l) \neq (0,0)} a_{k,l} f(x - k, y - l) + u(x, y) \quad (3.1)$$

où les  $a_{k,l}$  sont les coefficients du modèles, pris sur le demi-plan asymétrique<sup>1</sup> (NSHP)  $R_a$ , et  $u$  est un bruit blanc de moyenne nulle. On peut aussi utiliser la notation équivalente  $f = Af + u$ , par ordonnancement lexicographique des éléments de  $f$  et  $u$ . L'équation décrivant l'image dégradée (MA 2D) s'écrit elle

$$g(x, y) = \sum_{(k,l) \in R_d} h_{k,l} f(x - k, y - l) + b(x, y), \quad (3.2)$$

ou, sous forme plus compacte:  $g = Hf + b$ .

Ces deux expressions (3.1) et (3.2) peuvent être mises sous une forme réduite à une équation:

$$g = H(I - A)^{-1}u + b \quad (3.3)$$

De manière à spécifier complètement la manière dont les paramètres identifiés doivent être utilisés dans le schéma de restauration, on doit établir un critère définissant le résultat de la restauration. Il est en général choisi pour cela une régularisation quadratique, menant à une minimisation de la forme

$$\min_f \|Hf - g\|^2 + \alpha \|Lf\|^2 \quad (3.4)$$

où  $L$  est un opérateur de régularisation, et  $\alpha$  son paramètre de contrôle. Le résultat de restauration pour ce schéma est alors donné par

$$\tilde{f} = (H^T H + \alpha L^T L)^{-1} H^T g \quad (3.5)$$

Sur base de (3.3) et (3.5), on voit donc bien que pour obtenir  $\tilde{f}$ , il faut tout d'abord identifier les paramètres  $H$  du modèle MA, et  $A$  du modèle AR, ce qui, pour ce dernier cas, n'est pas trivial.

Validation croisée généralisée. Le critère de la validation croisée généralisée (angl. GCV) a tout d'abord été utilisé par Reeves [121] pour l'estimation des paramètres d'un modèle ARMA. Le principe de la validation croisée est simple: pour une valeur fixée des paramètres du modèle, une image « restaurée » est déterminée en utilisant toutes les valeurs provenant de l'image observée  $g$  sauf une, et l'image « restaurée » est renfloué afin de prédire l'observation qui avait été mise de côté pour la restauration.

<sup>1</sup> C'est-à-dire le demi-plan défini pour tout  $(m,n) \in Z^2$  par  $S(m;n) = \{(k,l) \in Z^2: k < m\} \cup \{(m,l) \in Z^2: l > n\}$ .

La formule  $V(\Theta)$  du critère du GCV, à minimiser afin d'identifier les paramètres  $\Theta$  recherchés, est donnée par

$$V(\Theta) = \frac{\frac{1}{N} \|(I - B(\Theta))g\|^2}{[\frac{1}{N} \text{tr}(I - B(\Theta))]^2} \quad (3.6)$$

où  $N$  est le nombre de pixels de l'image discrétisée, et

$$B(\Theta) = H(H^T H + \alpha(I - A)^T(I - A))^{-1} H^T \quad (3.7)$$

Les paramètres minimisant le critère  $V(\Theta)$  sont considérés comme les estimations des paramètres du modèle. Cependant, cette minimisation peut être délicate, car des minima locaux peuvent apparaître, de même que l'approche demande de grandes ressources en calculs.

Dans [25], il est proposé un schéma numérique efficace, différent de celui développé dans [26], pour la minimisation du critère du GCV, fournissant ainsi une première réponse au problème des capacités de calcul requises. L'approche des auteurs se base sur l'utilisation de règles de quadrature avec l'algorithme de Lanczos, afin d'approximer les numérateurs et dénominateurs de (3.6). Des applications intéressantes à la restauration et à la super résolution sont présentées, avec des résultats encourageants.

Autres méthodes. Notons que la modélisation ARMA a encore été utilisée assez récemment par Chan et Yap dans [49], afin d'en tirer deux estimateurs, utilisant des propriétés de l'autocorrélation du noyau, afin d'estimer non le flou lui-même, mais la taille de son support. Cette approche est en fait une extension partielle de [27], visant à pouvoir s'attaquer à toutes sortes de supports 2D généraux. Les estimateurs proposés se basent sur la conception d'un filtre adaptatif, obtenu d'après un modèle AR. Cette approche peut par exemple être utilisée en conjonction avec une méthode de déconvolution à identification jointe, de manière à ce que l'on n'ait alors plus qu'à estimer la valeur des coefficients sur le support de flou identifié, fournissant ainsi une contrainte très pertinente.

Pour davantage de détails sur ces questions d'estimation des flous dans le cadre de la modélisation ARMA, on pourra consulter [4]. Le défaut fondamental de ces approches utilisant la modélisation ARMA est qu'elles ne peuvent être fondamentalement adaptées qu'à des images relativement lisses, ce qui n'est que rarement le cas pour des scènes naturelles. Les modèles AR sont en effet incapables de reproduire des phénomènes de discontinuité, dont la prise en compte efficace est devenue incontournable dans les techniques contemporaines de traitement d'images.

### 2.1.2. Méthodes basées sur les caractéristiques spectrales de l'image

Ces méthodes comptent parmi les premières introduites en déconvolution aveugle d'images. Elles considèrent l'image observée comme une réalisation particulière d'un processus stochastique, et sont bien adaptées quand la réponse en fréquence du système optique

présente une forme paramétrique connue qui est complètement caractérisée par ses zéros dans le domaine fréquentiel. Ceci est par exemple le cas pour les flous de mouvement linéaires, de même que pour les noyaux circulaires de défocalisation. En revanche, ce n'est plus le cas par exemple pour des mouvements accélérés et des vibrations basses-fréquences.

Si l'on réécrit le modèle convolutif de floutage  $g = h * f$  dans le domaine fréquentiel,

$$\hat{g} = \hat{h} \cdot \hat{f} \quad (3.8)$$

alors, il apparaît qu'en raison de la multiplication des spectres, les zéros de  $\hat{g}$  sont la combinaison de ceux de  $\hat{h}$  et  $\hat{f}$ . Le problème revient donc à identifier les zéros de  $\hat{g}$  qui appartiennent à  $\hat{h}$ . L'utilisation de modèles paramétriques rend ceci possible: en effet, la transformée de Fourier de flous linéaires et circulaires est une fonction de type sinus cardinal ou de Bessel, qui présente des caractéristiques de périodicité dans la distribution de ses zéros spectraux. Pour une défocalisation par exemple, l'espacement de ces zéros dépend de son rayon. Si ce genre de caractéristique peut être détecté, alors les paramètres du noyau peuvent également être estimés.

En pratique, ce type d'approche se révèle cependant souvent inopérant en raison de la présence dans l'image de bruit, dont le spectre est approximativement plat, masquant ainsi les caractéristiques périodiques. La méthode homomorphique, ou l'approche cepstrale [23] tentent d'exploiter l'effet non stationnaire de l'image et l'effet stationnaire du flou pour contourner cette difficulté. L'approche homomorphique consiste à partitionner l'image en blocs  $f_i$ , de taille plus large que celle de la fonction de dispersion (ce qui suppose de disposer a priori d'une bonne estimation de son support). Chaque bloc  $g_i$  peut alors s'écrire  $g_i = f_i * h_i + b_i$ . Un opérateur logarithmique peut alors être appliqué à la transformée de Fourier de l'image floue, ce qui a pour effet de convertir la convolution originelle en une addition. La moyenne sur les blocs (en en supposant N) est alors :

$$\frac{1}{N} \sum_{i=1}^N \log \hat{g}_i \approx \frac{1}{N} \sum_{i=1}^N (\log \hat{f}_i + \log \hat{h}_i) \quad (3.9)$$

Cette sommation consiste ici en la moyenne des contributions provenant des blocs  $\hat{h}_i$ , qui sont supposés être égaux, et des blocs  $\hat{f}_i$ , qui eux ne le sont pas, puisque ceux-ci ont des contenus spectraux variantes. D'après ceci, la composante de flou devrait tendre à dominer dans cette expression.

Comme alternative, le cepstre de puissance peut être utilisé. Celui consiste à prendre la transformée de Fourier du logarithme du spectre de l'image, ou du spectre de puissance<sup>1</sup>. En combinaison avec la méthode de séparation des blocs décrite ci-dessus, le résultat est qu'un pic important apparaît dans le domaine cepstral où qu'il y ait une distribution périodique des zéros dans le domaine de Fourier original. La distance entre ce pic et l'origine représente

<sup>1</sup> Le spectre de puissance d'un signal est le module au carré de la transformée de Fourier de ce signal.

l'espacement entre les zéros et peut donc être utilisée pour identifier les paramètres du flou. L'importante contribution du terme de flou, répété dans chaque bloc, domine. Le bruit (haute fréquence) et le contenu spectral moyenné de l'image tendent alors à être séparés du pic dans le domaine cepstral.

Ce type de méthode a également été étendu à l'utilisation du bispectre de puissance<sup>1</sup> au lieu du spectre (classique) ou du cepstre, permettant des performances accrues dans des conditions de rapport signal/bruit faible. Cependant, la limitation à des noyaux paramétriques s'applique dans tous les cas dans ce genre d'approches.

Une méthode plus originale, basée sur l'utilisation des propriétés spectrales d'une classe large (mais cependant limitée) de flous pouvant s'exprimer comme des convolutions de densités de probabilités 2D symétriques stables au sens de Lévy, a été proposée par Carasso dans [28,29]. L'approche développée détecte la fonction de dispersion affectant une image, à partir de son analyse de Fourier unidimensionnelle. Elle peut être appliquée aux noyaux dont la transformée de Fourier, notée  $\hat{h}(u, v)$ , est de la forme

$$\hat{h}(u, v) = \exp^{-\alpha(u^2+v^2)^\beta} \quad (3.10)$$

avec  $\alpha > 0$  et  $0 < \beta < 1$ .

Si l'image floue  $g = h * f$  est obtenue par convolution, alors on a dans le domaine de Fourier

$$\hat{g}(u, v) = \hat{f}(u, v) \cdot \hat{h}(u, v) \quad (3.11)$$

$$= \hat{f}(u, v) \cdot \exp^{-\alpha(u^2+v^2)^\beta} \quad (3.12)$$

L'idée derrière la méthode d'identification du flou est alors d'ajuster la fonction  $\alpha|u|^2\beta$  au algorithme de la transformée de Fourier de l'image observée moins une estimée de l'image réelle (nette). Cet ajustement n'est possible qu'avec une classe d'images particulières (dites à « bon comportement »; cf. [28] pour une définition précise), incluant cependant une grande variété d'images numérisées. De même, une importante limitation de la méthode est l'impossibilité de traiter des flous de défocalisation ou de mouvement, étant donné leur non-appartenance à la classe des noyaux supposés.

### 2.1.3. Méthodes basées sur l'autocorrélation

Pour les flous de mouvement. Yitzhaky et Kopeika ont proposé dans [27] une approche pour l'identification des flous de mouvement rectilignes, à partir de la seule connaissance d'une unique image dégradée. La technique est limitée à l'identification des paramètres d'orientation et de longueur et ne peut s'appliquer à d'autres types de noyaux. L'idée générale est d'identifier certains paramètres importants par lesquels on peut ensuite caractériser la fonction de dispersion. On se base pour cela sur le fait que les caractéristiques de l'image le long de la

<sup>1</sup> Le bispectre est la transformée de Fourier de la triple corrélation.

direction du mouvement (soit l'angle du noyau flou) sont différentes de celles dans les autres directions.

L'identification de la direction du flou se fait en mesurant l'orientation dans laquelle le spectre de puissance de la dérivée de l'image est le plus faible, et la longueur du support est identifiée en utilisant des propriétés de la fonction d'autocorrélation de la dérivée de l'image dans la direction précédemment trouvée.

Concrètement, la dérivée horizontale (d'angle  $0^\circ$ ) au point  $f(i, j)$  se calcule très simplement par  $\Delta f(i, j)_{[0^\circ]} = f(i + 1, j) - f(i, j)$ , tandis que celle orientée de  $k^\circ$  sera donnée par  $\Delta f(i, j)_{[k^\circ]} = f(i', j') - f(i, j)$ , où  $f(i', j')$  est un pixel virtuel dans la direction  $k$  à partir du pixel  $f(i, j)$ , dont l'intensité est fixée à partir de celle de  $f(i + 1, j)$  et  $f(i + 1, j + 1)$ . L'intensité totale  $I(\Delta f)$  de la dérivée de l'image dans la direction  $k$  est alors la somme des valeurs absolues des pixels dans  $\Delta f(i, j)$ , c'est-à-dire

$$I(\Delta f)_{[k^\circ]} = \sum_1^{n-1} \sum_1^{m-1} |\Delta f(i, j)_{k^\circ}| \quad (3.13)$$

L'angle  $k$  minimisant cette fonction est alors identifié comme l'angle du noyau de mouvement. Une fois ce paramètre disponible, on calcule l'autocorrélation  $R_l(x)$  dans la direction du mouvement des dérivées par ligne  $l$  de l'image floue :

$$R_l(t) = \sum_{s=-M}^{+M} l(s + t)l(s) \quad \text{avec } t \in [-M, +M] \quad (3.14)$$

où  $l(s) = 0$  pour  $s \notin [0, +M]$ .

La moyenne des autocorrélations des lignes des dérivées de l'image  $\bar{R}_{\Delta f}$  est alors utilisée pour calculer l'étendue du flou, qui est la distance entre la localisation du minimum de  $\bar{R}_{\Delta f}$  et de son point de décalage en zéro  $\bar{R}_{\Delta f}(0)$ .

Cette approche s'est montrée être efficace pour différents types de mouvements, autant uniforme qu'accélérés, que ce soit sur des images synthétiques ou réelles. Dans [30], la même approche a été présentée, sous un angle légèrement différent, par lequel la direction du mouvement est obtenue par un filtrage passe-haut de l'image floue dans toutes les directions. Ceci est en fait équivalent au calcul du plus faible spectre de puissance dans une direction. Beaucoup d'applications à mouvement linéaire accélère ont été traitées dans ce cas. En outre, la même technique a été appliquée pour des modèles vibratoires sinusoïdaux (autant à basse qu'à haute fréquence).

Notons finalement que pour l'étape de restauration, les auteurs se sont limités à l'utilisation d'un filtre de Wiener, sans présumer de sa capacité à gérer ou non d'éventuelles erreurs

d'estimation du flou. Dans [31], ils ont proposé une comparaison de leur méthode - dénommée « blanchiment » (angl. whitening) - avec la méthode homomorphique de Stockman [24], de même qu'avec la méthode cepstrale de Cannon [23] (qui suppose-t-elle un noyau de mouvement uniforme). Par rapport à cette dernière, l'approche de Yitzhaky et Kopeika s'est révélée très légèrement inférieure pour des images présentant un faible SNR; de l'autre côté, la méthode cepstrale de Cannon s'est révélée incapable d'estimer des flous à grande longueur de support. Tout ceci pour des mouvements uniformes bien entendu. Dans tous les autres cas de mouvement traités (non-uniformes), la comparaison de l'approche de blanchiment avec l'approche homomorphique a montré des résultats d'identification supérieurs.

#### 2.1.4. Méthodes basées sur l'utilisation de la transformée de Radon

La transformée de Radon est une approche qui a été utilisée dans les techniques les plus récentes en matière d'identification des flous de mouvement. Pour une fonction  $f(x, y)$ , la transformée de Radon est définie par

$$R(f, \rho, \theta) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) \delta(\rho - x \cos \theta - y \sin \theta) dx dy \quad (3.15)$$

Citons par exemple le travail de Lun et al. [32], qui considèrent des flous symétriques généraux bidimensionnels (gaussien, défocaltion, etc.). Leur approche se base fondamentalement sur la modélisation ARMA et l'utilisation du critère GCV présente ci-avant. Mais l'utilisation d'une transformée discrète périodique de Radon leur permet de transformer le problème 2D (traité par GCV) en plusieurs problèmes 1D (traités par GCV aussi). Ceci permet alors de réduire sérieusement la complexité calculatoire nécessaire à la mise en œuvre du critère du GCV. La technique proposée peut donc être vue comme une version rapide du GCV 2D, et elle en partage donc globalement les mêmes limitations et avantages.

Une autre approche utilisant la transformée de Radon, mais seulement pour des flous de mouvement uniformes rectilignes, a été proposée par Moghaddam et Jamzad dans [33]. Leur méthode consiste en fait en l'identification de l'angle seul du mouvement par cette transformée, alors que la longueur du support est déduite grâce à l'information contenue dans le bispectre de l'image dégradée. L'estimation de l'angle est donnée par la valeur  $\theta$  permettant de maximiser la transformée de Radon. Notons que cette approche a été développée dans le cadre particulier des images fortement bruitées; sa robustesse dans ces cas étant sa principale contribution. Cependant, elle ne peut fonctionner que pour des noyaux à support relativement long, de telle sorte que l'image devient suffisamment lisse dans leur direction, menant alors à un maximum de la transformée. Enfin, rajoutons que, de nouveau, les auteurs se sont limités à l'utilisation d'un filtre de Wiener pour l'étape de restauration consécutive à l'identification des paramètres du flou.

Une des techniques les plus récentes utilisant la transformée de Radon a été présentée par Oliveira et al. dans [34], également dans le seul cas des noyaux de mouvement uniformes rectilignes. L'approche est fondamentalement très proche de celle de Moghaddam, mais

présente quelques améliorations, permettant de circonvenir à quelques défauts, tels que la nécessité d'être en présence de noyaux de longueur conséquente pour pouvoir obtenir un maximum dans la transformée et donc d'estimer l'angle du mouvement. Ici, même des flous de faible longueur sont identifiables. Pour cela, les auteurs utilisent une transformée de Radon exacte, prise sur des intervalles similaires. Au lieu de calculer la transformée pour toute l'image, on intègre sur le carré le plus grand qui puisse être inscrit dans l'image, ceci résultant en la formulation

$$\text{Rd}(f, \rho, \theta) = \int_{-\infty}^{+\infty} f(\rho \cos \theta - s \sin \theta, \rho \sin \theta - s \cos \theta) ds \quad (3.16)$$

Où  $d = \frac{\sqrt{2}}{2} n$ , avec  $n$  la taille de l'image (supposée carrée).

L'estimation de l'angle du flou est alors donnée par

$$\tilde{\theta} = \arg \max_{\theta} V \{ \text{Rd}(\log |\hat{g}|, \rho, \theta) \} \quad (3.17)$$

où  $V$  est l'opérateur de variance ; ceci étant déduit des propriétés de  $\log |\hat{g}|$ .

Pour l'estimation de la longueur du support, les auteurs utilisent les caractéristiques de  $\text{Rd}(\log |\hat{g}|, \rho, \tilde{\theta})$ , ce qui est aussi l'approche suivie dans [33]. L'algorithme proposé est cependant dit être plus simple, et ceci à précision équivalente. En introduisant la fonction  $N(w) = \text{Rd}(\log |\hat{g}|, w, \tilde{\theta})$ , ils montrent que les minima de cette fonction sont aussi ceux de la transformée de Fourier d'une pulsation rectangulaire<sup>1</sup> de longueur  $L$ . En supposant alors que  $N(w)$  est en effet la transformée de Fourier d'une pulsation de longueur  $L'$ , soit

$$N(w) = \exp^{i\psi(w)} \frac{\sin \frac{wL'}{2}}{\sin \frac{w}{2}} \quad (3.18)$$

Le but est alors de trouver  $L'$  en déterminant le premier zéro positif de  $N(w)$ .

### 2.1.5. Autres méthodes

Il existe plusieurs autres méthodes permettant l'identification a priori d'un flou dans des images dégradées, méthodes basées sur des concepts ou approches très divers. Loin de vouloir en fournir une liste exhaustive, nous en donnons quelques exemples ci-dessous.

Méthode utilisant une fonction de filtrage extraite. Dans [35,36], il est proposé une méthode utilisant la transformée de Fourier de l'image dégradée, mais dont le principe est assez différent des méthodes spectrales/cepstrales présentées si dessus.

<sup>1</sup> Pulsation rectangulaire à rapprocher de la modélisation du flou de mouvement rectiligne uniforme.

La fonction de dispersion est considérée avoir a priori une certaine forme et est extraite de l'image, ceci en passant par le domaine de Fourier. Dans le domaine fréquentiel, les auteurs postulent que pour un système linéaire et invariant spatialement, la fonction de transfert peut être déterminée à partir de la puissance  $\alpha$ -ième du spectre de puissance lissé de l'image dégradée. Ils posent donc que

$$\hat{h}(u, v) = K_{\hat{h}} S \left\{ \left| P_{\hat{g}(u,v)} - P_{\hat{b}(u,v)} \right|^{\frac{\alpha}{2}} \right\} \quad (3.19)$$

où  $K_{\hat{h}}$  est une constante assurant que  $|\hat{h}(u, v)| \leq 1$ ,  $S$  une fonction « lissante » (non précisée par les auteurs),  $\alpha$  un paramètre à estimer, et  $P$  l'opérateur de densité spectrale de puissance, définie pour  $\hat{f}$  par  $P_{\hat{f}} = \frac{|\hat{f}(u,v)|^2}{n^2}$ , où  $n$  est la taille de l'image.

Il est cependant montré que pour que cette condition soit valable, il faut que  $\hat{f}$  et  $\hat{h}$  soient reliés par la relation

$$\hat{h}^{1-\alpha}(u, v) = K_{\hat{h}} S P_{\hat{f}}^{\alpha/2}(u, v) \quad (3.20)$$

Assez obscures sont au final les raisons de la similarité entre la fonction de dispersion réelle et la fonction extraite, et il est fort probable que cette approche soit uniquement applicable dans un nombre de cas très limités.

Méthode utilisant les statistiques de l'image. Levin [37] utilise les statistiques de l'image (distribution des dérivées de l'image floue) pour estimer la longueur du support d'un flou de mouvement. L'auteur met en œuvre l'approche de Kopeika [30] pour déterminer l'angle du noyau. Cette approche, qui n'est valable que pour un mouvement uniforme, se base sur les travaux de Fergus et al. [38] concernant la restauration par apprentissage et l'intelligence artificielle, travaux que nous évoquerons plus loin.

Méthode basée sur la carte topographique de l'image. Ladjal [39] a proposé une méthode à la base destinée à quantifier localement le flou d'une image, en analysant ses zones de transition. Son approche permet seulement dans certains cas de revenir à l'expression du noyau de convolution, si celui-ci vérifie une hypothèse de symétrie radiale, ce qui n'est pas le cas de la plupart des noyaux de mouvement. L'auteur utilise dans ce cas une inversion de la transformée de Radon inverse, à partir de la largeur calculée du flou.

Méthodes basées sur l'analyse des transparences. Jia a proposé [40] une méthode également limitée aux flous de mouvements, utilisant l'information contenue dans les translucidités des zones floues entre deux régions opaques. Son approche est supervisée, dans le sens où l'utilisateur doit sélectionner lui-même les zones de transparence pertinentes.

L'estimation du noyau de flou est formulée comme la résolution d'un problème de maximum a posteriori avec une vraisemblance et a priori sur la transparence. La restauration est ensuite effectuée avec un algorithme de Richardson-Lucy [89] ou simplement avec un filtre inverse.

Méthodes basées sur les filtres de choc. Une approche permettant d'obtenir rapidement une estimée du flou d'une image par utilisation de filtres de choc a été proposée par Money et Kang dans [41].

Le filtrage de choc est une technique qui a été introduite par Osher et Rudin dans [42]. Celle-ci consiste de manière très générale à rehausser les contours d'une image. Pour cela, l'idée initiale était de s'inspirer de la physique des chocs et des fronts d'onde. Par résolution d'une équation aux dérivées partielles d'évolution en temps, l'image traitée voit ses bords (adoucis par le flou) accentués.

Cette image rehaussée peut alors être utilisée comme une approximation de l'image restaurée, que l'on peut noter  $g_r$ . On peut alors calculer à partir de cette image de référence et de l'image observée un noyau de convolution de référence  $h_r$ , en utilisant un schéma classique de déconvolution. Les auteurs ayant travaillé avec des flous de défocalisation et de mouvement (donc des noyaux discontinus) proposent de minimiser le critère pénalisé suivant

$$\min_h \frac{1}{2} \|h * g_r - g\|_{L^2}^2 + \int_{\Omega} |\nabla h| \quad (3.21)$$

Le noyau obtenu estimé est ensuite « raffiné », en raison de la présence de bruit dans son estimation. Pour cela, il est proposé une procédure en plusieurs étapes: passer le noyau estimé au filtre de choc, le normaliser, effectuer un premier débruitage par seuillage, avec une petite constante. Ensuite, une technique de débruitage adaptative, basée sur les travaux de [43], est mise en œuvre en calculant l'échelle de l'objet (i.e. du noyau), c'est-à-dire le rapport de son périmètre par sa surface, afin de pouvoir retrouver une forme correspondant bien à un mouvement rectiligne ou à un disque de défocalisation.

Bien que l'idée de l'utilisation d'un filtre de choc afin de rehausser les contours de l'image semble novatrice, la procédure de raffinement du noyau comporte tout de même une démarche très heuristique semblant bien peu à même de fonctionner dans des cas réels.

Méthodes basées sur le kurtosis. Le kurtosis  $k$  d'une variable aléatoire  $X$  est défini comme son moment centré du quatrième ordre :

$$k(X) = \frac{E\{X - \mu\}^4}{\sigma^4} \quad (3.22)$$

où  $\mu$  est la moyenne de  $X$  et  $\sigma$  son écart-type.  $E$  est la fonction d'espérance mathématique.

Cette caractéristique a été utilisée par Li et al. dans [44]. Li et al. employant cette statistique afin de mesurer le caractère pointu (la pointicité) de la distribution des niveaux de gris d'une image. Ils utilisent le fait que, pour une distribution dite platykurtique, c'est-à-dire une distribution avec une queue courte, pour lesquelles  $k < 3$ , plus grand est le kurtosis, plus lisse est cette distribution. Soit  $g$  une image bruitée donnée, à partir de la connaissance de la forme paramétrique du flou pour la fonction de dispersion, et en se plaçant dans un espace

raisonnable, on cherche la meilleure estimation du paramètre, en mesurant à chaque fois le kurtosis de l'image restaurée. À chaque étape de la boucle, les auteurs restaurent l'image avec un filtre de Wiener (car c'est une technique très rapide), et le kurtosis est mémorisé. L'image présentant la plus petite valeur  $k$  est alors choisie comme l'image restaurée, et le paramètre correspondant comme caractérisant le flou associé.

Bien entendu, cette approche n'est pas totalement une approche d'identification de flou, puisqu'elle doit être couplée à une étape de restauration. Néanmoins, cette étape de restauration ne peut pas être ici trop évoluée (et donc gourmande en ressources de calculs), dans la mesure où toutes les valeurs possibles des paramètres du noyau doivent être testées. Il est alors préférable d'utiliser, une fois que le noyau a été identifié, une procédure de restauration plus avancée, au choix de l'utilisateur.

Méthode basé sur la maximisation de la variance. Li et al. ont également utilisé l'analyse en composante principale pour obtenir un filtre de déflouage maximisant la variance de l'image restaurée [45]. Il a montré que, pour cela, l'estimation de l'image idéale peut être obtenue par

$$\tilde{f} = G\mu_1 + \psi \quad (3.23)$$

où  $G$  est une matrice à  $M$  colonnes comportant une certaine représentation des éléments  $\bar{g}_i$  de l'image sans bruit  $\bar{g}$ ,  $\mu_1$  est le vecteur propre correspondant à la plus grande valeur propre de la matrice de covariance  $G^T G$ , et  $\psi = \frac{1}{M} \sum_{i=1}^M \bar{g}_i$ . Le vecteur  $\mu_1$  est d'ailleurs en fait le filtre de déflouage.

Seuls des tests sur des images affectées par de très légers flous, sans perte forte de contenu dans l'image, ont été effectués par les auteurs, et il est à craindre que des images réelles soient difficilement appréhendables avec cette approche, dont la robustesse paraît limitée. D'autre part, cette méthode semble nécessiter la connaissance (du moins approximative) de la taille du support du flou, ce qui est assez gênant dans le cadre d'applications pratiques.

Méthodes basées sur les techniques d'apprentissage. Une approche prometteuse, pour le moment limitée à l'estimation de flous provenant de bougés (humains) d'appareils photographiques, a été proposée par Fergus et al. [38]. Celle-ci exploite les statistiques des images naturelles (comme la technique de [37]), et en particulier de la distribution des gradients. L'approche est également supervisée, nécessitant le choix d'une zone rectangulaire pertinente de l'image, et certaines hypothèses a priori sur le noyau à estimer (orientation et taille maximum du support). Un algorithme bayésien, permettant une prise en compte des incertitudes sur les données, est alors mis en œuvre, permettant d'identifier le flou à partir d'une distribution d'images probables. L'algorithme d'apprentissage employé pour résoudre le problème de séparation de ces images est originellement proposé dans [46] (et est relié à l'analyse en composantes indépendantes). Les comparaisons effectuées par les auteurs par rapport à l'algorithme deconvblind de MATLAB® ont montré la supériorité de leur approche.

Bronstein et al. ont également proposé une méthode permettant l'estimation d'un filtre de déflouage, sur la base d'une technique d'apprentissage. Pour cela, une approche basée sur des

estimateurs de quasi maximum de vraisemblance du filtre recherché est utilisée, en considérant une modélisation de la distribution des niveaux de gris d'une image de signaux creux, par une famille de fonctions convexes lisses

$$\varphi_\lambda(s) = |s| - \lambda \log\left(1 + \frac{|s|}{\lambda}\right) \quad (3.24)$$

La méthode est intéressante, mais les auteurs n'ayant pas effectué de tests sur des images traditionnellement utilisées dans la littérature, ni de comparaisons avec d'autres approches, il est bien difficile de se prononcer quant à ses capacités de manière générale.

## 2.2. Restauration de l'image avec flou connu

La deuxième étape dans le déroulement d'une méthode de déconvolution aveugle avec identification a priori du flou, est l'inversion de celui-ci, c'est-à-dire l'étape de restauration proprement dite. Fondamentalement, n'importe quelle approche peut être utilisée, et la quasi-totalité des auteurs des méthodes présentées ci-dessus pour l'estimation de noyau ne cherche pas à utiliser une méthode spécialement adaptée, par exemple à l'incertitude qui peut demeurer sur le noyau identifié ou sur sa forme.

Par la suite, nous allons donner un aperçu des types de méthodes existantes, que nous classerons dans les catégories suivantes :

- ❖ Méthodes variationnelles.
- ❖ Méthodes utilisant l'analyse multirésolution et les ondelettes.
- ❖ Méthodes basées sur les équations aux dérivées partielles.
- ❖ Autres méthodes.

Notons encore une fois qu'il existe une quasi-infinité d'approches, et qu'il n'est pas de notre propos d'en donner une liste exhaustive. Nous nous limiterons donc à fournir un aperçu général, avec une description plus en profondeur de certaines méthodes récentes ou présentant un intérêt particulier.

### 2.2.1. Méthodes variationnelles

Parmi les méthodes de restauration d'images floues, la classe des méthodes dites variationnelles (c'est-à-dire basées sur l'analyse des variations d'une fonctionnelle donnée), joue un rôle de premier plan. Un modèle générique variationnel, peut s'écrire

$$\min_f \mathcal{J}(f) = \min_f \int_{\Omega} \Phi(h * f - g) + \alpha R(f) \quad (3.25)$$

où  $\Phi$  une norme caractérisant la fidélité aux données du problème,  $R$  est un opérateur de régularisation, et  $\alpha$  le paramètre de contrôle de la régularisation.

Méthodes itératives basiques. Dans le cas où  $\Phi$  est la norme  $L^2$  et  $R = 0$ , on a le schéma

classique des moindres carrés, dont la solution peut être vue comme un simple filtre inverse. Ce filtre inverse peut être calculé explicitement par une approche comme l'itération de van Cittert [47], une des méthodes de déconvolution les plus anciennes, et qui s'exprime par

$$f_{i+1} = g + (I - h) * f_i \quad (3.26)$$

où  $I$  est l'opérateur identité et  $f_0 = g$ . On montre que cette itération ne converge cependant pas pour des flous discontinus, ce qui l'exclut donc pour traiter des images affectées par des noyaux de défocalisation ou de mouvement.

Une autre approche assez similaire, grand classique de la déconvolution, est la méthode de Landweber [48] :

$$f_{i+1} = f_i + \gamma h^{\otimes} * (h * f_i - g) \quad (3.27)$$

Méthode de Tihonov. Les premiers travaux concernant la régularisation des problèmes inverses, menés dans un cadre totalement mathématique (équation de Fredholm du premier ordre) et hors de toute considération de traitement d'images, ont été effectués par Tihonov dans les années soixante.

Ses premières études ont consisté à considérer pour la régularisation la norme  $L^2$  de la fonction  $f$  recherchée, soit la fonctionnelle

$$\mathcal{J}(f) = \|h * f - g\|^2 + \alpha \int_{\Omega} |f|^2 \quad (3.28)$$

L'interprétation de ce modèle est de fournir un compromis entre la proximité avec la solution des moindres carrés, et une solution de norme minimale. Ce compromis est contrôlé par l'intermédiaire du paramètre  $\alpha$ , dont le réglage adéquat est en soi un problème délicat. Dans un deuxième temps, Tihonov a proposé de minimiser la norme de la dérivée de la fonction, soit la semi-norme  $H^1$  de  $f$ , résultant en le modèle

$$\mathcal{J}(f) = \|h * f - g\|^2 + \alpha \int_{\Omega} |\nabla f|^2 \quad (3.29)$$

Celui-ci est plus à même de pénaliser les irrégularités locales de l'image, bien qu'étonnamment, dans ses premiers travaux en traitement d'images, souvent inconnus de la communauté scientifique occidentale, Tihonov et al. continuent à considérer la minimisation de (3.28) plutôt que celle de (3.29).

La minimisation de tels critères quadratiques (3.28) et (3.29) présente le gros avantage de pouvoir être mise en œuvre par des techniques rapides. On citera par exemple la transformée de Fourier rapide [49], ou les approches multigrilles [50].

Bien que cela ne soit pas du tout la façon usuelle de le présenter, le filtre de Wiener peut également être relié à une méthode variationnelle et, plus particulièrement, à la minimisation de (3.29). En effet, il revient aussi à considérer la minimisation de cette fonctionnelle, mais

dans le domaine fréquentiel, soit de

$$\mathcal{J}(f) = \|\hat{h}f - \hat{g}\|^2 + \alpha \int_{\Omega} |\nabla \hat{f}|^2 \quad (3.30)$$

où  $\alpha$  dépend alors de la variance  $\alpha^2$  du bruit présent dans l'image.

L'utilisation de ces critères quadratiques présente cependant le lourd défaut de trop pénaliser les discontinuités des images lors de l'inversion. Or celles-ci revêtent une importance fondamentale dans la quasi-totalité des applications, puisqu'une image est quasi toujours composée de zones homogènes séparées par de franches discontinuités. Un effet de lissage est donc observé à chaque utilisation de tels schémas, résultant en un léger flou persistant, très gênant visuellement. Pour contrecarrer cette difficulté, des termes de pénalisation autres que les normes  $L^2$  et  $H^1$  ont été considérés par nombre d'auteurs.

Méthodes avec prise en compte des discontinuités. De nombreux travaux ont été menés ces dernières années, portant sur la définition de termes de régularisation nonquadratiques d'une part, et sur les méthodes de minimisation nécessaire au traitement efficace des fonctionnelles résultantes de l'utilisation de ce type d'approches.

Norme  $L^1$  et variation totale. Une des premières approches naturelles permettant de moins pénaliser les fortes discontinuités est de ne plus considérer un terme de pénalisation en  $\|\nabla f\|^2$  mais plutôt la norme  $L^1$  vert  $\nabla f$  du gradient, ce qui résulte en une fonctionnelle à minimiser du type

$$\mathcal{J}(f) = \|h * f - g\|^2 + \alpha \int_{\Omega} |\nabla f| \quad (3.31)$$

Mathématiquement, la solution à ce problème appartient à un espace analogue à celui de Sobolev  $W^{1,1}$ , et est également trop douce pour modéliser des images formées de régions régulières séparées par des contours. C'est pourquoi, il est plus judicieux de se positionner dans l'espace des fonctions à variation bornée BV, qui, lui, permet des points de discontinuités sur des courbes. On arrive donc à des fonctionnelles utilisant comme régularisateur la variation totale  $TV(f) = \int_{\Omega} |Df|$ , où  $D$  est le gradient au sens des distributions :

$$\mathcal{J}(f) = \|h * f - g\|^2 + \alpha \int_{\Omega} |Df| \quad (3.32)$$

En pratique néanmoins, une telle formulation au sens des distributions est difficilement utilisable, et on se rabattra sur l'approximation

$$TV(f) = \int_{\Omega} |Df| \approx \int_{\Omega} |\nabla f| \quad (3.33)$$

qui, en théorie, n'est valable que pour  $f$  différentiable, et donc continue. Ceci ramenant au final à (3.31).

Notons qu'en variables discrétisées, la différence entre la norme  $L^1$  du gradient et sa variation totale donne souvent lieu à des confusions, et il y a ambiguïté dans la littérature à ce sujet. Très souvent, le régularisateur  $l^1(f) = \sum_i |\Delta_i^{hor} f| + |\Delta_i^{ver} f|$  est utilisé comme approximation de la variation totale<sup>1</sup>, ou même considéré lui-même à tort comme le régularisateur TV, dont le modèle isotrope discret s'écrit plutôt  $TV(f) = \sum_i \sqrt{(\Delta_i^{hor} f)^2 + (\Delta_i^{ver} f)^2}$  [51, 52, 53], ceci en raison de la consistance, au sens de l'analyse numérique, associée à ce dernier schéma de discrétisation. On reviendra plus longuement à cette question de discrétisation au chapitre suivant.

Autour de la variation totale, plusieurs méthodes ont proposé l'introduction de paramètres ou informations supplémentaires.

On citera par exemple la méthode de Combettes et Pesquet de contrainte sur la variation totale [54]. Les auteurs y proposent de ne plus utiliser exclusivement la variation totale comme un objectif à minimiser sous certaines contraintes, mais introduisent une formulation alternative dans laquelle la variation totale est utilisée elle-même comme une contrainte parmi d'autres dans un contexte d'optimisation convexe. Ceci leur permet de faire face au problème d'effet d'escalier souvent rencontré dans les formulations classiques du type (3.31), et de ne pas avoir besoin de connaître l'environnement du bruit. Des expérimentations ont été effectuées sur des images satellitaires avec des résultats convaincants.

Minimisation de la variation totale. Pour ce qui est de la minimisation de (3.32), un grand nombre d'approches a été proposé, que ce soit en débruitage [55, 56, 57, 58], ou en déconvolution [59, 60]. On pourra citer pour ce deuxième cas la récente méthode de Bioucas-Dias [52], qui propose une manière originale et efficace de traiter le problème de restauration avec régularisation TV. L'algorithme de minimisation, au lieu de traiter directement ce problème difficile, le remplace par une suite de problèmes plus simples, en utilisant des arguments de convexité.

Soit  $f(t)$  le calcul de  $f$  à l'instant  $t$ . Les auteurs cherchent un majorant quadratique de la fonctionnelle  $L(f) = \|Hf-g\|_2 + \alpha TV(f)$ , et montrent pour cela que

$$QTV(f|f(t)) = TV(f(t)) + \frac{\alpha}{2} \sum_i \frac{[(\Delta_i^h f)^2 - (\Delta_i^v f)^2]}{\sqrt{(\Delta_i^h f(t))^2 + (\Delta_i^v f(t))^2}} + \frac{\alpha}{2} \sum_i \frac{[(\Delta_i^v f)^2 - (\Delta_i^h f)^2]}{\sqrt{(\Delta_i^h f(t))^2 + (\Delta_i^v f(t))^2}} \quad (3.34)$$

en est un pour  $TV(f)$ . On a donc  $TV(f) \leq QTV(f|f(t))$  pour  $f \neq f(t)$  et  $TV(f) = QTV(f|f(t))$  pour  $f = f(t)$ .

<sup>1</sup>Les  $\Delta_i^{hor}$  et  $\Delta_i^{ver}$  sont les opérateurs aux différences finies du premier ordre, respectivement horizontal et vertical, d'indice  $i$ .

En remarquant que le premier terme de  $L(f)$  est également quadratique, cette fonctionnelle a alors comme borne quadratique  $Q(f, f(t)) = \|Hf - g\|^2 + QTV(f, f(t))$ . La minimisation de cette dernière expression menant par suite à

$$f^{(t+1)} = (H^T H + D^T W^{(t)} D)^{-1} H^T g \quad (3.35)$$

où  $W^{(t)} = \text{diag}(w^{(t)}, w^{(t)})$  avec

$$w^{(t)} = \left[ \frac{\alpha/2}{\sqrt{(\Delta_i^h f^{(t)})^2 + (\Delta_i^v f^{(t)})^2}} \right], i = 1, 2, \dots \quad (3.36)$$

et  $D = [(Dh)^T (Dv)^T]^T$  avec  $Dh$  et  $Dv$  les matrices telles que  $D^h f$  et  $D^v f$  donnent les différences horizontales et verticales du premier ordre.

Cette méthode a surclassé toutes celles auxquelles elle a été comparée, méthodes récentes et basées sur les ondelettes pour la plupart: Bioucas-Dias et al. [61], Figueiredo et Nowak [62], Banham et Katsaggelos [63], Mignotte [64], Neelamani et al. [65]. Les résultats se sont avérés convaincants et il s'agit d'une des méthodes parmi les plus performantes. Notons qu'une adaptation intéressante a été proposée dans [51,53], où le paramètre de régularisation  $\alpha$  est automatiquement réglé, avec cependant des performances légèrement amoindries.

Fonctions  $\varphi$  et conditions de préservation des discontinuités. De manière plus générale, les conditions permettant de définir des opérateurs de régularisation à préservation des discontinuités, dans le cadre du formalisme continu. La fonctionnelle de restauration considérée dans ce cas s'écrit

$$\mathcal{J}(f) = \|h * f - g\|^2 + \alpha \int_{\Omega} \varphi |\nabla f| \quad (3.37)$$

avec  $\varphi$  fonction paire définie sur  $\mathbb{R}$ , et croissante sur  $\mathbb{R}^+$ . Les conditions à respecter par  $\varphi$  sont alors les suivantes :

$$\left\{ \begin{array}{l} \lim_{t \rightarrow 0} \frac{\varphi'(t)}{2t} = 1 \text{ (lissage dans les régions à faible gradient)} \\ \lim_{t \rightarrow \infty} \frac{\varphi'(t)}{2t} = 0 \text{ (préservation des grands gradients aux discontinuités)} \\ \frac{\varphi'(t)}{2t} \text{ est strictement croissante} \end{array} \right. \quad (3.38)$$

De nombreuses fonctions présentant de telles caractéristiques ont été proposées, parmi lesquelles la fonction de Green  $\varphi(t) = 2 \log(\cosh(t))$ , la fonction de Geman-MacClure  $\varphi(t) = \frac{t^2}{1+t^2}$  [66] ou encore la fonction de Perona-Malik  $\varphi(t) = 1 - \exp(-t^2)$  [67].

Régularisation convexe et non-convexe. Parmi les fonctions  $\varphi$  évoquées ci-dessus, certaines pourront être convexes, soit  $\varphi'' \geq 0$ , et d'autres non. Si  $\varphi$  est convexe, alors la fonctionnelle (3.37) à minimiser l'est aussi, facilitant ainsi grandement la tâche d'optimisation. D'autre part, sous certaines conditions, des résultats d'existence et d'unicité du minimum existent. Si  $\varphi$  est non-convexe, alors la minimisation est bien plus délicate, et l'existence d'un minimum n'est pas garantie. De manière globale, on pourra dire qu'une fonctionnelle non-convexe reconstruira des discontinuités de manière plus instable qu'une fonctionnelle convexe, mais ces discontinuités présenteront un caractère plus net.

Notons que dans un schéma de déconvolution aveugle, on utilise souvent le terme de régularisation convexe  $\varphi(t) = \sqrt{1 + t^2}$  (comme approximation de la TV), puis d'un terme non convexe  $\varphi(t) = \min(t^2, 1)$ .

Minimisation semi-quadratique. Parmi les algorithmes utilisés pour la minimisation de critères pénalisés avec prise en compte des discontinuités, les techniques dites semi-quadratiques tiennent une place importante. L'idée principale est d'introduire et d'optimiser une nouvelle fonction coût, ayant le même minimum que la fonctionnelle originale non-quadratique.

On peut montrer que les fonctions  $\varphi$  définies ci-dessus vérifient  $\varphi(t) = \min_{b \in [0,1]}(bt^2 + \psi(b))$ , où  $\psi$  est une fonction strictement convexe. A partir de là, il découle que la minimisation de (3.37) est équivalente à celle de la première forme (ou forme multiplicative) semi-quadratique

$$\mathcal{J}^{\triangleright}(f, b) = \int_{\Omega} |h * f - g|^2 + \alpha(\int_{\Omega} b |\nabla f|^2 + \int_{\Omega} \psi(b)) \quad (3.39)$$

dans laquelle la variable  $b$  représente la présence ou l'absence d'une discontinuité, selon que sa valeur est (respectivement) proche de zéro ou de un. On peut voir que dans ce deuxième cas, la régularisation devient quadratique, alors que le terme en  $\|\Delta f\|$  disparaît en présence d'un bord, d'où l'appellation semi-quadratique. Une deuxième forme (ou forme additive) de telle régularisation peut être obtenue en remarquant que  $\varphi(t) = \min_{b \in \mathbb{R}^+}((t - b)^2 + \xi(b))$ , où  $\xi$  est encore une fonction strictement convexe, cette fois définie par  $\xi(b) = \min_{s \in \mathbb{R}}(\varphi(s) - (s - b)^2)$ . Alors, de même que pour le premier cas, minimiser (3.37) revient à minimiser

$$\mathcal{J}^{\triangleleft}(f, b) = \int_{\Omega} |h * f - g|^2 + \alpha(\int_{\Omega} (|\nabla f|^2 - b)^2 + \int_{\Omega} \xi(b)) \quad (3.40)$$

Influence du terme de fidélité. La quasi-totalité des méthodes de restauration par minimisation de critères pénalisés considèrent des termes de fidélité aux données en norme  $L^2$ , par soucis de s'approcher le plus possible d'une solution de type moindres carrés, mais surtout par simplicité, puisque l'utilisation d'une norme quadratique pose beaucoup moins de difficultés théoriques. Néanmoins, cette approche n'est pas la seule possible, et certains auteurs ont commencé ces dernières années à envisager l'utilisation de termes de fidélité en norme  $L^1$ , ou même  $L^\infty$ .

Lintner et Malgouyres [68] ont par exemple proposé un modèle utilisant la norme  $L^\infty$  pour le terme de fidélité aux données  $\Phi$ . Leur modèle se place dans le cadre de la régularisation par ondelettes, et s'écrit

$$\left\{ \begin{array}{l} \min_f \int_{\Omega} |\nabla f| \\ \|h * f - g\|_{L^\infty, D \leq \tau} \end{array} \right\} \quad (3.41)$$

où le terme de fidélité est en fait donné par la norme  $L^\infty$  du produit scalaire avec les éléments d'un dictionnaire de caractéristiques  $D$ :  $\|f\|_{L^\infty, D} = \sup_{\psi \in D} (\langle f, \psi \rangle)$ .

L'utilisation de cette norme est dite porter plus d'informations et permettre de réduire l'espace de recherche pour le minimum de la variation totale dans la régularisation. La minimisation est alors effectuée en utilisant une méthode dérivée de celle d'Uzawa, et résulte en une meilleure préservation des textures qu'avec le modèle de Rudin-Osher-Fatemi [56].

Parmi les travaux en restauration utilisant d'autres termes de fidélité, on pourra citer ceux de Bar et al. [69], qui envisagent l'approximation de la norme  $L^1$  (approximation permettant d'éviter l'apparition de singularités en zéro), traitant ainsi le modèle

$$\mathcal{J}(f) = \int_{\Omega} \sqrt{(h * f - g)^2 + \eta} + \alpha R(f) \quad (3.42)$$

Celui-ci s'est montré efficace dans la restauration d'images affectées par des bruits impulsifs et autres observations aberrantes (angl. outliers).

De même, dans [70, 71, 72], une norme  $L^1$  est également utilisée comme terme de fidélité aux données :

$$\mathcal{J}(f) = \|h * f - g\|_{L^1} + \alpha R(f) \quad (3.43)$$

Dans [71], les auteurs visent comme intérêt de ce terme le fait de ne plus supposer d'a priori gaussien, ni sur le bruit, ni sur la distribution des niveaux de gris de l'image.

Dans [70], on envisage plutôt la norme  $L^1$  comme un moyen d'éviter la perte de contraste caractérisant la restauration des images effectuées par des modèles du type  $L^2 + TV$ <sup>1</sup>, tandis que dans [72], on considère davantage l'incertitude caractérisant le flou disponible pour effectuer l'étape de déconvolution.

<sup>1</sup> C'est-à-dire le modèle à fidélité aux données en norme  $L^2$  et régularisation TV.

Nikolova étudie dans [73] le modèle générique à terme de fidélité non-lisse

$$\mathcal{J}(f) = \int_{\Omega} \psi(h * f - g) + \alpha R(f) \quad (3.44)$$

avec une emphase sur le cas  $\psi(t) = |t|^p$ , ce qui résulte alors en le modèle

$$\mathcal{J}(f) = \int_{\Omega} |h * f - g|^p + \alpha R(f) \quad (3.45)$$

Ce type de fonctionnelle en est, dans son utilisation, encore au stade des balbutiements, mais il est à penser que son usage se développera dans les années à venir.

Moindres carrés totaux et moindre norme totale. Les moindres carrés totaux sont une approche qui étend les moindres carrés traditionnels, et permet de gérer une incertitude sur l'image et la fonction de dispersion. Dans cette méthode, la fonction de dispersion est supposée contenir une composante déterministe, de même qu'une composante stochastique:  $h = \bar{h} + \delta h$ . Ceci résulte alors en le modèle de dégradation  $\bar{g} + \delta g = (\bar{h} + \delta h) * f$ , dans lequel  $\bar{g}$  est la composante déterministe de l'image observée, et  $\delta g$  sa composante stochastique. Le problème est alors formulé comme la minimisation de  $\delta g$  et  $\delta h$ , avec les contraintes de l'équation de modélisation précédente.

Ce type de technique a par exemple été utilisé dans [74] pour la déconvolution aveugle. Cependant, la méthode des auteurs requiert une connaissance relativement précise de la fonction de dispersion, ce qui l'exclut immédiatement s'il y a nécessité d'une reconstruction complète, ce qui est le cas dans la vaste majorité des problèmes aveugles. L'absence de comparaisons avec d'autres méthodes rend également l'évaluation des possibilités de restauration délicate.

Dans [75], un algorithme assez proche est proposé, basé sur l'utilisation de la moindre norme totale. Celui-ci n'est cependant pas très convaincant, car la technique APEX/SECB [28] décrite plus avant fait mieux ou aussi bien dans tous les cas, et ce sans connaissance préalable du noyau de dégradation.

### 2.2.2. Méthodes utilisant l'analyse multirésolution et les ondelettes

La transformée continue en ondelettes permet de décomposer un signal en un espace d'échelles/positions qui sépare les composantes de différente taille. L'analyse des caractéristiques significatives dans cet espace mène à des filtrages adaptatifs.

### 2.2.3. Méthodes basées sur les équations aux dérivées partielles

Ces méthodes se basent sur la résolution d'un problème se ramenant à une équation aux dérivées partielles (EDP). Les modèles de diffusion sont les plus couramment utilisés. Une des difficultés principales est de choisir judicieusement le terme de diffusivité, de manière homologue au choix des opérateurs de régularisation pour les approches variationnelles.

On citera par exemple le schéma de Welk et al. [72, 76], qui présente une méthode de déconvolution se basant à la fois sur le formalisme des EDP et celui variationnel, et utilisant des tenseurs de diffusion anisotropes [77].

En partant du modèle variationnel général de déflouage suivant,

$$\mathcal{J}(f) = \int (h * f - g)^2 + \alpha \psi(|\nabla|^2) \quad (3.46)$$

les auteurs obtiennent, en appliquant la condition d'Euler, une EDP de diffusion-réaction:

$$\partial_t f = h^{\odot} * (h * f - g) + \alpha \operatorname{div}(u(|\nabla|^2) \nabla f) \quad (3.47)$$

dans laquelle le terme de diffusion  $u(s^2) = \Psi'(s^2)$  est relié au terme de régularisation dans la fonctionnelle d'énergie (3.46). Ils choisissent d'utiliser des diffusivités de type TV :

$u(s^2) = 1/|s|$ , souvent utilisée dans sa forme « régularisée »  $u(s^2) = 1/\sqrt{s^2 + \varepsilon^2}$ , où  $\varepsilon$  doit être petit, et une diffusivité de type Perona-Malik :  $u(s^2) = (1 + s^2/\lambda^2)$ , où  $\lambda$  est un paramètre à fixer. Le choix de ces termes de diffusivité provenant de leur qualité reconnue à bien faire ressortir des bords nets dans les images. Un schéma numérique de type différences finies est ensuite appliqué pour la résolution de cette équation.

Du point de vue du SNR, la restauration avec diffusivité de type Perona-Malik s'avère meilleure qu'avec la diffusivité TV; cependant, l'aspect visuel est beaucoup plus discutable.

## 2.2.4. Autres méthodes

Nous présentons ici quelques méthodes basées sur des approches plus originales, et/ou présentant des résultats intéressants, et qu'il est difficile de classer dans les catégories ci-dessus.

Il existe une quasi infinité d'autres algorithmes, et comme nous l'avons déjà dit, il est impossible et hors de notre propos d'en dresser un recensement exhaustif. Néanmoins, on pourra citer à titre d'exemple les méthodes utilisant des réseaux de neurones [78], la méthode CLEAN [79] et des approches dérivées en radioastronomie [80], ou encore les méthodes basées sur divers filtres adaptatifs [81]. Dans [81], on calcule en fait un filtre de restauration à partir de la carte des bords de l'image dégradée, en utilisant des modèles de référence pour la distribution statistique des niveaux de gris. Cependant, pour plusieurs de ces techniques à filtrage adaptatif, force est de constater qu'il s'agit souvent plus d'améliorer l'image floue que de la restaurer à proprement parler.

Parmi les autres approches que nous pensons destinées à se développer, figurent celles basées sur des traitements locaux de l'image dégradée. Nous en donnons quelques éléments dans les paragraphes ci-dessous, en présentant un cas particulier.

Déconvolution locale/adaptative. Katkovnik et al. [82] ont proposé une méthode utilisant la

régression non-paramétrique. L'idée se base sur une approximation locale polynomiale de l'image (dénommée LPA) et un paradigme d'intersection d'intervalles de confiance (ICI), qui est appliqué pour définir des échelles adaptatives variantes de l'estimateur LPA. L'algorithme est non-linéaire et spatialement adaptable à la douceur et aux irrégularités des images corrompues par un bruit additif.

On considère un paramètre d'échelle  $e$ , et un noyau (filtre passe-bas d'ordre  $m$ )  $u_e$  défini tel que  $f_e = u_e * f$ . Sa bande passante se réduit quand l'échelle  $e$  augmente. Si celle-ci est petite, la bande passante de l'estimateur est large et le signal  $f$  est reproduit sans distorsions. À l'inverse, si l'échelle est grande, la bande passante est étroite et seules les composantes basses fréquences de  $f$  peuvent être observées en sortie. La conception du filtre  $u_e$  est effectuée grâce à la méthode LPA présentée dans [83].

L'idée basique de l'algorithme est d'utiliser l'image lissée  $f_e$  au lieu de  $f$  comme solution du problème inverse, et d'exploiter l'estimateur du noyau  $u_e$  muni du paramètre d'échelle  $e$  dans le but de supprimer autant que possible le bruit en préservant les détails de la fonction objet  $f$ . En appliquant l'opérateur de noyau  $u_e$  aux deux membres de  $g = h * f + b$ , on obtient dans le domaine fréquentiel  $\hat{g}_e = \hat{f}_e \hat{h} + \hat{b}_e$  ou  $\hat{g}_e = \hat{f} \hat{h}_e + \hat{b}_e$

Deux types de « solutions » à ces expressions sont alors utilisées comme estimées de  $f_e$  : l'inverse régularisée.

$$\tilde{f}_e = \frac{\hat{h}^\oplus}{(\hat{h})^2 + \varepsilon_1^2} \hat{g}_e = \frac{\hat{h}^\oplus \hat{u}_e}{(\hat{h})^2 + \varepsilon_1^2} \hat{g} \quad (3.48)$$

et l'inverse de Wiener régularisée

$$\tilde{f}_e = \frac{\hat{h}^\oplus (\hat{f})^2 \hat{u}_e}{|\hat{h} \hat{f}|^2 + \varepsilon_2^2 \sigma^2} \hat{g} \quad (3.49)$$

La régression non paramétrique LPA avec le paramètre optimal  $e$  permet alors une haute qualité d'approximation de  $f$ . La règle ICI pour la sélection de l'échelle adaptative est obtenue à partir de la précision de LPA. On consultera [84] pour le fonctionnement détaillé de ICI.

Les résultats obtenus par les auteurs se sont révélés être très bons, et les comparaisons effectuées avec l'algorithme ForWaRD de Neelamani et al. et EM de Figuereido et Nowak [85] en faveur de leur approche dans tous les cas traités.

Takeda et al. ont proposé par la suite [86] ce qu'on pourrait qualifier d'évolution de la technique de Katkovnik, qu'ils qualifient de suboptimale, en raison de la présence de deux étapes dans son processus de déflouage: débruitage et déconvolution.

Leur algorithme est basé sur le choix d'une fonctionnelle spécifique à minimiser (en ce sens il rentre dans la catégorie des méthodes variationnelles), utilisant un terme d'a priori sur l'image

(c'est-à-dire un opérateur de régularisation) particulier, basé sur les caractéristiques de celle-ci, et qualifié de « variation totale à noyau adaptatif ».

De très bons résultats ont été observés par les auteurs par rapport à l'approche de Katkovnik, ceci en faisant ainsi probablement l'une des méthodes d'état de l'art à l'heure actuelle en déconvolution à noyau connu.

### **2.3. Synthèse sur les méthodes avec identification du flou a priori**

Nous avons donné dans les sections ci-dessus un aperçu, que nous espérons le plus complet possible, des différentes manières envisageables pour traiter un problème de déconvolution aveugle en deux étapes disjointes: estimation du flou (ou de son inverse) à partir de la seule image dégradée, puis restauration par une des méthodes traditionnelles de restauration à noyau connu.

Le choix de la procédure d'identification du flou se fera suivant le type de noyau à identifier. Étant donné que nous avons à traiter des images dégradées par des noyaux de défocalisation (entre autre), des procédures telles que l'utilisation du kurtosis, ou le filtrage de choc, nous semblent être parmi les plus adaptées.

Globalement, on a pu constater un choix quasi-illimité pour ce qui est de la restauration d'images à niveaux de gris avec connaissance parfaite du flou. Cependant, dans un schéma de déconvolution aveugle à identification du flou a priori, on privilégiera plutôt pour l'étape de restauration une approche robuste permettant de compenser l'erreur d'estimation sur le flou, inévitable en pratique, telle que l'utilisation de normes non-quadratiques pour le terme de fidélité aux données.

Pour ce qui est de la régularisation, on veillera à utiliser des termes à préservation des discontinuités, la manière la plus simple étant de choisir des opérateurs convexes, ou pouvant être approximés par une fonction convexe.

En bref, un schéma de restauration variationnel basé sur l'utilisation d'une fonctionnelle comprenant un terme de fidélité aux données en norme  $L^1$  (pour la robustesse aux incertitudes du flou identifié), et une régularisation TV approximée (pour sa capacité bien connue à la préservation de bords, et sa relative simplicité de mise en œuvre), nous paraît être un choix judicieux.

### **3. Méthodes d'identification jointe**

Le problème de déconvolution à identification jointe cherche à déterminer de manière parallèle le noyau de flou et l'image originale, nette donc. On pourra alors également le voir qualifié de « conjoint » ou « aveugle ». Cependant, en réalité, beaucoup de ces méthodes s'y rapportant utilisent en fait une approche alternée entre la reconstruction de  $h$  et  $f$ , plutôt que de calculer à proprement une solution jointe.

On trouvera également parfois la dénomination « myope » ou « semi-aveugle », ceci

provenant du fait que, d'une manière ou d'une autre, des hypothèses sont toujours faites sur certaines données du problème, dans le but de ne pas se trouver dans une configuration qui reviendrait à traiter sans information supplémentaire une (seule) équation à deux inconnues (de une le flou, et de deux l'image nette).

De manière générale, on pourra scinder cette classe d'algorithmes en deux groupes: méthodes directes, et méthodes itératives, alternant de manière successive la reconstruction de  $h$  et  $f$ , et ce jusqu'à (éventuelle) convergence.

Notons que la grande majorité des méthodes rentre dans le deuxième groupe, soit celui des approches itératives. De même, elles sont pour la plupart issues du formalisme variationnel. Nous donnerons dans ce qui suit un exemple de méthode directe, tandis que toutes les autres présentées seront itératives. Ces méthodes seront listées selon l'ordre suivant :

- ❖ Méthode directe de séparation de la plage des zéros.
- ❖ Méthode d'Ayers-Dainty.
- ❖ Méthode des moindres carrés non-régularisés.
- ❖ Méthodes à double régularisation.
- ❖ Méthodes de filtrage adaptatif.
- ❖ Méthodes basées sur l'analyse en composantes indépendantes.
- ❖ Méthodes diverses.

### 3.1. Méthode directe de séparation de la plage des zéros

Un des premiers algorithmes conjoint de restauration des signaux multidimensionnels est dû à Lane et Bates [87], qui ont proposé une approche basée sur le principe de séparation de la plage des zéros de l'image flouée  $g$ .

Ils ont montré que tout signal issu de multiples convolutions entre d'autres signaux est en théorie automatiquement déconvoluable si sa dimension est supérieure à un. Cet argument repose sur les propriétés de la transformée en  $Z$  de signaux  $n$ -dimensionnels à support fini, qui prend nécessairement la valeur zéro sur des hypersurfaces de dimension  $2n - 2$  dans un espace de dimension  $2n$ .

L'approche suppose l'absence de bruit additif, ce qui résulte en

$$\check{g}(z_1, z_2) = \check{h}(z_1, z_2)\check{f}(z_1, z_2) \quad (3.50)$$

dans le  $Z$ -domaine, avec  $\check{f}$  la transformée en  $Z$  de  $f$ .

Elle consiste alors à séparer la transformée en  $Z$  de l'image floue  $\check{g}(z_1, z_2)$  en les deux facteurs de la convolution. Comme  $\check{g}(z_1, z_2)$  est un polynôme, la solution est équivalente à en trouver les racines qui appartiennent à chaque composante.

Pour cela, on effectue plusieurs hypothèses: support fini compact pour  $\check{h}(z_1, z_2)$  et  $\check{g}(z_1, z_2)$ ;

la transformée en  $Z$  de chaque facteur doit être nulle sur une seule surface continue; et ces plages de zéros ne peuvent se couper qu'en des points discrets.

En pratique, l'applicabilité de cette méthode est malgré tout limitée, en raison de sa grande complexité calculatoire, et également de sa sensibilité au bruit. On préférera alors s'orienter pour plus de robustesse vers une approche itérative.

### 3.2. Méthode d'Ayers-Dainty

Autrement appelé algorithme de « déconvolution aveugle itérative » (angl. Iterative blind deconvolution), cette approche est la plus connue des approches aveugles à estimation conjointe du flou et de restauration de l'image dégradée. Elle consiste à itérer les estimations dans le domaine de Fourier et dans le domaine spatial, en imposant des contraintes dans ces deux espaces. Ces contraintes, la façon dont elles sont utilisées, de même que la manière de mettre en œuvre cet algorithme ont donné lieu à différentes versions. On pourra consulter [88] pour l'approche originelle par Ayers et Dainty.

L'estimation dans le domaine de Fourier est faite à partir d'un filtre assimilable à celui de Wiener, et est donnée, à la  $n^{\text{ième}}$  itération par

$$\hat{h}_n = \frac{\hat{g}\hat{f}_{n-1}^{\circledast}}{|\hat{f}|^2 + \frac{\alpha}{|\hat{h}_{n-1}|^2}} \quad (3.51)$$

et

$$\hat{f}_n = \frac{\hat{g}\hat{h}_{n-1}^{\circledast}}{|\hat{h}|^2 + \frac{\alpha}{|\hat{f}_{n-1}|^2}} \quad (3.52)$$

Il existe une adaptation de cette méthode proposée par Holmes [90], qui, au lieu d'utiliser itérativement ces formules de Wiener, se base sur l'algorithme dit « EM » (angl. Expectation-Maximization), largement utilisé dans un grand nombre d'applications d'imagerie, et ceci sous différents noms, dont le plus connu est sûrement celui de Richardson-Lucy [89]. Dans le cas d'un modèle à flou connu  $g = h * f$ , linéaire et spatialement invariant, l'itération EM procède de la manière suivante :

$$f_{n+1} = f_n h_n^{\circledast} * \left( \frac{g}{h * f_n} \right) \quad (3.53)$$

Pour un modèle aveugle, on obtient un algorithme à double itération en « échangeant » l'image et la fonction de dispersion, ce qui résulte alors en le schéma alterné suivant :

$$f_{n+1} = f_n h_n^{\circledast} * \left( \frac{g}{h * f_n} \right) \quad (3.54)$$

et

$$h_{n+1} = \frac{h_n}{\int_{\Omega} g} f_n^{\odot} * \left( \frac{g}{h * f_n} \right) \quad (3.55)$$

Cependant, bien que la convergence de l'algorithme EM soit bien établie, celle de la double itération n'est, elle, pas garantie, ce qui peut mener à un fonctionnement assez aléatoire suivant les données traitées.

Notons que cette approche n'est pas à proprement parler régularisée, et qu'elle peut être vue comme la minimisation d'un critère standard des moindres carrés, sous certaines contraintes. Certains auteurs ont alors proposé, afin de s'affranchir - ou tout du moins de limiter- les problèmes de convergence de cet algorithme, d'employer des formulations explicites des moindres carrés avec (ou non) régularisation.

### 3.3. Méthode des moindres carrés non-régularisés

McCallum [91] a par exemple proposé de simplement minimiser la fonctionnelle

$$\min_{f,h} \mathcal{J}(f, h) = \min_{f,h} \int (h * f - g)^2 \quad (3.56)$$

dans laquelle  $f$  et  $h$  sont supposés positifs et de support connu. L'optimisation est mise en œuvre par une technique appelée « recuit simulé » (angl. simulated annealing), consistant à rechercher le minimum global par des perturbations aléatoires des variables: si celles-ci mènent à une diminution de  $\mathcal{J}$ , elles sont acceptées, sinon elles sont rejetées avec une loi de probabilité particulière, dépendant du nombre d'itérations déjà effectuées.

Le gros inconvénient de cette approche réside cependant dans sa vitesse de convergence plutôt lente, la rendant ainsi difficilement utilisable avec des images de taille courante, présentant une résolution relativement élevée, comme c'est le cas pour celles issues d'appareils photographiques modernes.

### 3.4. Méthodes à double régularisation

Depuis quelques années sont apparues des méthodes basées sur la minimisation de fonctionnelles de type moindres carrés, avec régularisation portant à la fois sur l'image à restaurer et sur le noyau à reconstruire. Ces approches sont une extension au problème aveugle de la régularisation simple, dont nous avons parlé auparavant, et qui est utilisée pour la déconvolution d'images à flou connu. Cette dernière consiste à minimiser une fonctionnelle dont la forme générique est donnée (avec les mêmes notations que précédemment) par

$$\mathcal{J}(f) = \int_{\Omega} \Phi(h * f - g) + \alpha R1(f) \quad (3.57)$$

où  $\Phi$  est une norme pour le terme de fidélité aux données,  $R1$  l'opérateur de régularisation (autrement appelé « fonction potentiel ») visant à contrôler l'amplification du bruit lors du

processus d'inversion, et  $\alpha$  le paramètre de régularisation permettant de contrôler la balance, l'équilibre, entre la fidélité aux données dans le premier terme, et la contrainte a priori introduite par l'opérateur de régularisation. Ceci sur le domaine  $\Omega \subset R^2$  de l'image.

En imaginant que nous soyons dans une configuration inversée dans laquelle nous disposerions des images floue et nette, mais pas de la dégradation, alors celle-ci pourrait être estimée par le schéma similaire suivant:

$$\mathcal{J}(h) = \int_{\Omega} \Phi(h * f - g) + \alpha R^2(h) \quad (3.58)$$

dans lequel le terme de régularisation porte ici dorénavant sur  $h$ .

Pour un problème aveugle, qui est alors doublement mal posé (à la fois sur l'image et sur le flou), on peut considérer la « somme » des deux problèmes (3.57) et (3.58). Ceci ramène alors à la minimisation sur  $f$  et  $h$  de la fonctionnelle

$$\mathcal{J}(f, h) = \int_{\Omega} \Phi(h * f - g) + \alpha_1 R^1(f) + \alpha_2 R^2(h) \quad (3.59)$$

où  $\Phi$  est comme précédemment la norme pour le terme de fidélité aux données,  $R^1$  l'opérateur de régularisation sur l'image,  $R^2$  celui sur le flou, et  $\alpha_1$  et  $\alpha_2$  les paramètres de régularisation respectifs sur  $f$  et  $h$ . L'un des principaux problèmes de cette approche réside dans le choix des opérateurs  $R^1$ ,  $R^2$  et des paramètres  $\alpha_1$  et  $\alpha_2$ , de même que dans la conception d'un algorithme efficace pour la minimisation de la fonctionnelle (3.59), algorithme dont la complexité dépendra alors de la régularisation retenue.

Globalement, on pourra scinder ces approches de double régularisation en deux groupes. Un premier groupe de méthodes qu'on pourra qualifier de « non-paramétriques », dans lesquelles l'opérateur de flou  $h$  et l'image  $f$  sont considérées comme étant de forme générale, indépendante de toute représentation via des paramètres. Ensuite un deuxième groupe, dans lequel  $h$  et/ou  $f$  pourront satisfaire un certain degré de structure paramétrique, comme par exemple en considérant des flous de défocalisation, de mouvement, ou encore gaussiens.

### 3.4.1. Approches non-paramétriques

Lane [92] a par exemple considère l'extension suivante de (3.28),

$$\mathcal{J}(f, h) = ||h * f - g||^2 + \alpha_1 \int_{\Omega} |f|^2 + \alpha_2 \int_{\Omega} |h|^2 \quad (3.60)$$

avec des contraintes concernant le support de  $f$  et  $h$ , de même qu'une contrainte sur leur positivité. Il s'agit donc ici du cas  $R^1(f) = \int_{\Omega} |f|^2$  et  $R^2(h) = \int_{\Omega} |h|^2$ . La minimisation s'effectue alors à partir d'une technique de type gradient conjugué. Cette approche - probablement l'une des premières à mettre en œuvre un principe de double régularisation - a également permis de mettre en évidence les difficultés propres à ce type de méthodes: influence du bruit sur la reconstruction de l'image solution, critère d'arrêt à choisir, prise en compte rigoureuse des contraintes considérées, etc.

You et Kaveh [93] ont eu proposé d'utiliser  $R^1(f) = f_{\Omega} |\nabla f|^2$  et  $R^2(h) = f_{\Omega} |\nabla h|^2$ , en conjonction avec une norme  $L^2$  pour le terme de fidélité aux données  $\phi$ , soit une extension de (3.29). Malheureusement, l'utilisation de cette semi-norme  $H^1$  pour les opérateurs de régularisation mène au problème bien connu de reconstruction d'image et de noyau trop lisses, comme nous l'avons présenté dans la Section 2.2.1. Notons qu'en fait, les auteurs considèrent en outre, à des fins d'affinage de la régularisation, des matrices de pondération  $\lambda_f$  et  $\lambda_h$ , menant en définitive au traitement du problème suivant:

$$\min_{f,h} \mathcal{J}(f, h) = \min_{f,h} \frac{1}{2} \|h * f - g\|_{L^2}^2 + \alpha_1 \int_{\Omega} \lambda_f |\nabla f|^2 + \alpha_2 \int_{\Omega} \lambda_h |\nabla h|^2 \quad (3.61)$$

L'optimisation mise en œuvre se base sur une minimisation alternée suivant  $f$  et  $h$ . La pertinence de ces matrices de pondération est cependant discutable, et leur définition bien peu aisée, puisque celles-ci sont estimées à partir de l'image observée seulement. En outre, en pratique, on est limité à des flous de support (très) réduits, de l'ordre de quelques pixels.

Par la suite, il a été proposé, comme dans le cadre de la restauration à noyau connu, des termes de régularisation prenant en compte les discontinuités inhérentes à la plupart des images naturelles. Ceci peut par exemple être fait en utilisant la norme TV, comme cela a été mis en œuvre dans le contexte de la déconvolution aveugle par Chan et Wong [94]. Leur modèle de déflouage étant alors

$$\min_{f,h} \mathcal{J}(f, h) = \min_{f,h} \frac{1}{2} \|h * f - g\|_{L^2}^2 + \alpha_1 \int_{\Omega} |\nabla f| + \alpha_2 \int_{\Omega} |\nabla h| \quad (3.62)$$

Le principe de minimisation utilisé est fondamentalement le même que celui de You et Kaveh, à savoir une minimisation selon des directions alternées. La prise en compte de termes de régularisation non-convexes pose alors des difficultés théoriques, résolues par une approche de linéarisation des composantes du gradient de la fonctionnelle.

Notons que You et Kaveh ont également proposé un schéma utilisant ce type de régularisation, pour un modèle de flou anisotrope [95].

Ces dernières années, plusieurs auteurs ont essayé de pallier les défauts des approches de You et Kaveh ou de Chan et Wong (convergence globale, influence de l'initialisation, paramètres de régularisation, etc.). He, Marquina et Osher, dans [96] ont par exemple considéré un raffinement de la technique introduite par Chan et Wong, et basée sur le principe de l'itération de Bregman. Ils modifient la fonctionnelle d'énergie (3.62), en y ajoutant des contraintes sur le noyau de convolution et l'image originale, dans le but de se défaire du problème de non-unicité de la solution.

Une régularisation itérative est ensuite appliquée en utilisant la distance de Bregman pour le nouveau problème de déconvolution. Celle-ci consiste à remplacer l'image  $g$  par  $g + v_n$  à l'itération  $n + 1$  de l'algorithme alterné, avec

$$v_n = f + v_{n-1} - h * f_n \quad (3.63)$$

Une contrainte de la forme  $\int f = w = \int g$  est également introduite afin de pallier la non-unicité de la solution du modèle de Chan et Wong. Le modèle modifié obtenu, c'est-à-dire la fonctionnelle à minimiser, se trouve alors être

$$\begin{aligned} \mathcal{J}(f, h) = & \frac{1}{2} \|h * f - g\|_{L^2}^2 + \alpha_1 TV(f) + \alpha_2 TV(h) + \\ & \frac{\lambda_1}{2} (\int_{\Omega} f - w)^2 + \frac{\lambda_2}{2} (\int_{\Omega} |f| - w)^2 + \frac{\lambda_3}{2} (\int_{\Omega} h - 1)^2 + \frac{\lambda_4}{2} (\int_{\Omega} |h| - 1)^2 \end{aligned} \quad (3.64)$$

Un schéma explicite est ensuite obtenu afin de minimiser cette fonctionnelle, schéma utilisant la régularisation itérée de Bregman présentée ci-haut.

Les résultats présentés par les auteurs semblent très légèrement supérieurs à ceux obtenus par Chan et Wong, et les temps de calcul très raisonnables (de l'ordre de quelques minutes pour l'ensemble du processus). L'itération de Bregman semble permettre de faire apparaître plus de détails dans les images que ne le peut la méthode de Chan et Wong initiale, mais risque toujours de légèrement plus amplifier le bruit. D'autant plus qu'aucun résultat n'a été présenté sur des images réelles, mais seulement sur une unique image à support fini, exhibant ainsi des propriétés particulières que nous évoquerons dans le chapitre suivant. Au final, il est à craindre que cette méthode soit principalement basée sur un artifice mathématique qui ne change rien à la capacité de reconstruction du noyau de flou, élément fondamental et primordial de la problématique aveugle.

Précisons que, dans [97], il est proposé une reformulation du modèle ci-dessus comme un problème d'espace d'échelle inverse. Les méthodes d'espace d'échelles consistent globalement à lisser les petites caractéristiques plus rapidement que les grandes par un modèle d'équation d'évolution en temps. Les mêmes limitations que dans l'approche variationnelle s'appliquent cependant également, dans la mesure où la modélisation en elle-même ne change aucunement.

### 3.4.2. Approches paramétriques

Dans [98], il est proposé un modèle qualifié de « semi-aveugle » qui fait l'hypothèse que l'on ne connaît pas précisément la fonction de dispersion, mais uniquement à quelle classe de fonctions paramétriques elle appartient, en l'occurrence à celle des noyaux gaussiens, paramétrables par un unique paramètre. La technique se base de nouveau sur le modèle de Chan et Wong [94]. Les auteurs choisissent d'utiliser un terme de régularisation de type Mumford-Shah pour l'image, ce qui permet de résoudre simultanément le problème de segmentation (soit de partitionnement de l'image) et de restauration. Le principe de  $\Gamma$ -convergence [99] est appliqué pour l'approximation discrète. Le modèle variationnel considéré est alors

$$\min_{f,v,\sigma} \frac{1}{2} \int_{\Omega} (h_{\sigma} * f - g)^2 + G_{\epsilon}(f, v) + \gamma \int_{\Omega} |\nabla h_{\sigma}|^2 \quad (3.65)$$

où  $\sigma$  est le paramètre du flou, et  $G_{\epsilon}(f, v)$  l'opérateur de régularisation défini par

$$G_{\epsilon}(f, v) = \beta \int v^2 |\nabla f|^2 + \alpha \int \left( \epsilon |\nabla v|^2 + \frac{(v-1)^2}{4\epsilon} \right) \quad (3.66)$$

Ici,  $\epsilon$  est le paramètre assurant la  $\Gamma$ -convergence, qui doit être choisi proche de zéro pour que celle-ci soit vérifiée, et  $v$  l'opérateur donnant une approximation de la carte des discontinuités de l'image. Cette fonctionnelle est minimisée de manière alternée par rapport à  $f$ ,  $v$  et  $\sigma$ , en utilisant les conditions d'Euler-Lagrange, avec des conditions aux bords de type Neumann. A chaque étape de la procédure itérative, on minimise donc par rapport à une fonction en gardant les deux autres fixées.

Chen et Yap ont proposé dans [100] une approche basée sur [101], qualifiée de déconvolution par double régularisation paramétrique, visant à minimiser l'expression

$$\mathcal{J}(f, h) = \frac{1}{2} \|Hf - g\|^2 + R(f, h) + S(f, h) \quad (3.67)$$

$R(f, h)$  représente ici une fonctionnelle de régularisation qui introduit de la stabilité dans la solution en imposant des contraintes de douceur sur l'image et la fonction de dispersion. Elle est définie par

$$R(f, h) = \frac{1}{2} \|\Lambda \Delta f\|^2 + \frac{1}{2} \|\Psi \Delta f\|^2 \quad (3.68)$$

où  $\Delta$  est l'opérateur laplacien (passe-haut) standard, comme utilisé par You et Kaveh, et  $\Lambda$  et  $\Psi$  sont des matrices de régularisation adaptatives qui ont deux buts :

Assigner des paramètres de régularisation spatialement variantes à différentes régions de l'image pour encourager la préservation des détails dans les régions texturées, et supprimer le bruit dans les zones lisses fournir un compromis entre la fidélité aux données et la fidélité au modèle.

$S(f, h)$  représente lui un terme de modélisation paramétrique pour l'image et le flou .

Il joue le rôle d'un terme de renforcement d'apprentissage pour intégrer une structure paramétrique potentiellement utile de l'image et du flou. La plupart des flous satisfont en effet à une certaine forme de structure paramétrique, du moins en théorie. Ce terme a donc la forme

$$S(f, h) = \frac{1}{2} \|\Gamma(h - h_f)\|^2 \quad (3.69)$$

où  $h_f$  est un estimateur pour le flou, et  $\Gamma$  une matrice diagonale qui assigne différentes

emphases basées sur des différences relatives entre  $h$  et  $h_f$ .

La procédure d'optimisation est basée, comme pour les modèles précédents, sur l'algorithme classique de minimisation alternée.

### 3.5. Méthodes de filtrage adaptatif

Plusieurs méthodes basées sur des techniques de filtrage adaptatif ont également été proposées afin de traiter le problème de déconvolution aveugle. Une des plus connues parmi celles-là est peut-être la méthode NAS-RIF (angl. Nonnegativity And Support constraints Recursive Inverse Filtering) développée par Kundur et Hatzinakos [102]. Celle-ci peut être rapprochée de la technique de déconvolution aveugle itérative d'Ayers et Dainty, et cherche à minimiser une fonctionnelle, en mettant à jour à chaque étape un filtre de restauration  $w$  (inverse du noyau de dégradation), qui est convolué avec  $g$  afin de fournir l'estimée de  $f$ . Contrairement aux approches de double régularisation évoquées ci-dessus, on ne minimise donc pas ici par rapport à  $h$ , mais par rapport à son inverse. La fonction coût utilisée comprend les mêmes contraintes que l'algorithme d'Ayers-Dainty, excepté pour ce qui est des contraintes sur la fonction de dispersion. Celle-ci doit ici seulement être d'inverse absolument sommable. Comme pour un grand nombre d'approches, le gros défaut de cette méthode est néanmoins qu'elle ne soit applicable qu'à des images à support fini, c'est-à-dire entièrement incluses dans un fond uniforme noir. Elle n'est donc pas utilisable en l'état pour notre problématique et sera plutôt réservée à des domaines tels que l'observation spatiale ou la microscopie.

Dans [103], il est proposé une variante de l'algorithme NAS-RIF, reprenant globalement le même concept, mais y appliquant un principe de régularisation, rendant la technique (qui, on le rappelle, était à la base seulement régularisée par un arrêt des itérations de la méthode) plus robuste au bruit. Une première régularisation est effectuée par un principe de troncature des valeurs propres sur le filtre inverse estimé, tandis qu'une seconde possibilité est introduite, permettant d'utiliser un schéma de régularisation par minimisation de la variation totale de l'image. Bien qu'améliorant le schéma initial de Kundur et Hatzinakos, celui-ci reste quand même très sensible à l'initialisation du filtre, rendant en ce sens l'algorithme délicat à utiliser en pratique.

Un certain nombre d'autres approches peuvent être vues comme des variantes de l'algorithme NAS-RIF. On pourra citer à titre d'exemple [104], qui propose la construction itérative d'un filtre inverse, par utilisation de diverses contraintes (entre autre de non-négativité), ou encore [105], qui utilise l'information de la transformée de Radon des bords de l'image pour mettre en œuvre une banque de filtres de déconvolution, résultant en une technique appelée algorithme de type « Bussgang », efficace apparemment surtout pour des observations multiples de la même scène, bien que, en théorie, l'approche puisse fonctionner avec une seule image.

Une approche autorégressive adaptative plus originale a été publiée par Vural et Sethares dans [106, 107]. Celle-ci se base sur la minimisation d'un critère de dispersion, qui est défini, pour une image  $f$ , par

$$\gamma = \frac{E\{f^4\}}{E^2\{f^2\}} \quad (3.70)$$

où  $E$  est l'espérance mathématique. Le critère à minimiser est issu des techniques de séparation aveugle de source et s'écrit

$$\mathcal{J}(f) = \frac{1}{4}(f^2 - \gamma)^2 \quad (3.71)$$

En notant que l'estimée  $\tilde{f}$  de  $f$  est donnée par  $\tilde{f} = w^T g$ , où  $w$  est le filtre adaptatif de déconvolution (soit le filtre inverse de  $h$ ), et en utilisant le fait que

$$\frac{\partial \mathcal{J}}{\partial w} = \frac{\partial \mathcal{J}}{\partial \tilde{f}} \frac{\partial \tilde{f}}{\partial w} = [\tilde{f}^2 - \gamma] \tilde{f} g \quad (3.72)$$

on obtient alors le schéma itératif suivant sur  $w$ ,

$$w_{i+1} = w_i - \mu \frac{\partial \mathcal{J}}{\partial w_i} \quad (3.73)$$

où  $\mu$  est une constante petite (à fixer), garantissant la stabilité de l'algorithme.

### 3.6. Méthodes basées sur l'analyse en composantes indépendantes

Le principe de l'analyse en composantes indépendantes (ICA) pour le problème de déconvolution a été introduit par Umeyama dans [108]. Cette technique est basée sur l'utilisation d'une banque de filtres de Gabor 2D, appliquée à l'image floue  $g$  observée. Les images en sortie de ces filtres, de même que l'image dégradée originale sont alors soumises à une analyse en composantes indépendantes. Cette analyse est menée en utilisant un algorithme de gradient stochastique, et, en sélectionnant de manière appropriée un paramètre d'apprentissage, l'ICA estime l'image déflouée comme étant la première composante indépendante, quand celle-ci provient d'une scène naturelle.

Cette approche a été reprise par Kopriva et al. [109], car Umeyama suppose une condition critique qui est que l'image source et ses dérivées spatiales doivent être statistiquement indépendantes. Or ceci n'est pas vrai en général. La qualité de l'image restaurée par la méthode d'Umeyama dépend donc surtout du niveau auquel chaque image traitée vérifie cette condition d'indépendance statistique. Kopriva et al. montrent que le problème de déconvolution aveugle d'une image unique peut être formulé en tant que problème de factorisation de matrice non-négative, avec des contraintes de caractère creux (angl.

sparseness) imposées.

Dans [110], il a été proposé une nouvelle extension se basant sur l'analyse en composantes indépendantes par décomposition en sous-bandes (par ondelettes), technique issue d'approches utilisées en séparation de sources statistiquement dépendantes. Une méthode pour résoudre ce type de problèmes est de relâcher la contrainte d'indépendance statistique et de supposer que les signaux à bande large sont dépendants, mais qu'il existe des sous-bandes plus étroites pour lesquels ils sont indépendants. L'avantage de cette approche par rapport à celle de la factorisation matricielle non-négative de [110] est que l'on évite alors des problèmes potentiels associés à la non-unicité de la décomposition matricielle et à la sélection des contraintes de caractère creux. Une fois la sous-bande avec les composantes les moins dépendantes identifiées, l'algorithme de décomposition en composantes indépendantes peut être utilisé pour obtenir soit une estimation de l'inverse de la matrice base, ou une estimation de la matrice base elle-même.

### **3.7. Méthodes diverses**

Comme pour le cas des méthodes à estimation du flou a priori, il existe dans le cadre de la déconvolution à identification jointe de nombreuses autres approches présentant des degrés divers d'originalité et/ou de performances réelles. On pourra citer à titre indicatif les méthodes basées sur l'apprentissage et l'intelligence artificielle, tel que l'algorithme d'évolution présenté dans [111], ou encore les méthodes prenant en compte des sources multiples de la même image nette à des fins de restauration plus performante. Dans ce dernier domaine, on citera les travaux de Sroubek [112] qui font référence dans le domaine, les travaux de Molina [20] qui présente une nouvelle approche Bayésienne hiérarchique, et par exemple [113] qui présente une méthode spécialisée pour le traitement des flous de mouvement.

### **3.8. Synthèse sur les méthodes d'identification jointe**

Nous avons présenté dans ce qui précède un aperçu des différentes catégories de méthodes de déconvolution aveugle fonctionnant suivant le principe d'identification jointe de l'image restaurée et du noyau de flou.

Parmi les possibilités qui s'offrent à nous, nous avons choisi d'étudier dans le chapitre suivant une méthode de déconvolution aveugle basée sur les travaux de Wang et Molina [20]. Ceci en raison de l'aspect relativement « évoluée » de leur approche, souvent utilisée dans des comparatifs de techniques, et également en raison du fait que des algorithmes de minimisation évolués et bien adaptés existent pour traiter ce modèle, qui préserve les discontinuités des images.

## **4. Conclusion**

Un état de l'art des méthodes récentes en déconvolution aveugle d'images a été effectué ici. Celui-ci nous permet d'avoir un aperçu global des techniques existantes et de pouvoir en

retirer les plus performantes. Il nous permet également de dégager un certain nombre de pistes à explorer.

Il ressort de cela, et de par les informations dont nous disposons quant aux dégradations affectant les images acquises par un aéronef en vol, ou une voiture a grand vitesse pour le cadre de notre application, que les méthodes les plus intéressantes à considérer rentrent dans la catégorie des approches d'identification jointe, et plus particulièrement des méthodes bayésiennes. En effet, ce sont quasi exclusivement les seules qui permettent une résolution du problème aveugle.

D'opérateur de régularisation convexe tels que variation totale, sera utilisé dans le prochain chapitre. Les phases de test comporteront ensuite des résultats sur images simulées, contenant des caractéristiques et des éléments que l'on peut chercher à visualiser plus en détail.

## Chapitre 4

# Déconvolution aveugle Bayésienne de deux différentes images

Dans ce chapitre, nous présentons l'algorithme adopté dans ce travail. Nous considérons deux images de la même scène afin d'obtenir une estimation de la scène originale par l'intermédiaire d'un nouvel algorithme de déconvolution aveugle. Nous formulons le problème dans un cadre bayésien hiérarchique en utilisant les connaissances a priori sur l'image inconnue et le flou, et aussi sur la dépendance entre les deux images observées.

### 1. Introduction

Prise photographie de haute qualité sous conditions de faible éclairage est vraiment un challenge.

Un temps d'exposition plus long que d'habitude sert à obtenir une image avec peu de bruit, mais tout mouvement de l'appareil photo lors de l'exposition entraîne la présence de flou dans l'image enregistrée. D'autre part, un temps d'exposition court se traduira par une image avec un très haut niveau de bruit.

Éventuelles solutions matérielles comprennent l'augmentation de la sensibilité à la lumière (angl. Increasing the light sensitivity (ISO)) du capteur de la caméra, ce qui augmente le niveau de bruit; l'augmentation de l'ouverture, ce qui conduit à une plus petite profondeur de champ dans l'image acquise, et l'utilisation d'un trépied pour stabiliser l'appareil qui n'est pas pratique dans de nombreux cas et surtout dans notre cas applicatif. En outre, de nombreux appareils photo numériques incluent des stabilisateurs d'image optique, que ce soit à l'intérieur du boîtier de l'appareil ou à l'intérieur de la lentille, qui aident à réduire considérablement les dégradations causées par photographie prise en main. Cependant, lorsque les temps d'exposition sont trop longs, ces solutions matérielles basées ne peuvent pas fournir des résultats satisfaisants. Dans ces cas, les méthodes numériques de stabilisation d'image, appliquées à un stade de post-traitement, constituent un moyen vigoureux pour obtenir des images de haute qualité en utilisant les observations de faible qualité.

Dans le cas de photographie prise en utilisant un temps d'exposition prolongé sous faible éclairage, le flou dans l'image obtenue peut être enlevé en utilisant un algorithme de déconvolution aveugle d'une image unique. Un certain nombre de méthodes sont proposées pour la déconvolution aveugle d'une seule observation, et le cas particulier de la restauration des images dégradées par le mouvement de camera est abordé dans [29]-[115]. Toutefois, en raison de la nature complexe du problème, l'obtention d'un résultat de restauration de haute qualité est très difficile dans la plupart des cas et nécessite de supervision. En outre, en raison de la difficulté d'estimer la dégradation d'image prise par camera située sur voiture ou sur aéro, les images restaurées présentent généralement de déconvolution artefacts comme sonnerie.

Une autre approche consiste à utiliser un temps d'exposition court pour éviter le flou au détriment de bruit élevé, puis appliquer des algorithmes de débruitage sur l'image pour

éliminer le bruit. De nombreuses méthodes de débruitage avancées sont disponibles dans cette littérature. Cependant, le niveau de bruit dans ces courts exposés des images est généralement très haut, et donc les caractéristiques de l'image sont cachées, et les algorithmes de débruitage ne peuvent pas facilement séparer l'image et le bruit. Un problème supplémentaire, et peut-être plus important, c'est qu'en raison de la brève durée d'exposition, les images ont généralement une faible luminosité et les couleurs peuvent être partiellement perdues.

Récemment, des méthodes de déconvolution ont été proposées qui utilisent une paire d'images au lieu d'une seule observation, où deux images sont prises sous différents temps d'exposition [116]. Certains appareils photo numériques ont des caractéristiques de « bracketing » d'exposition qui permettent à l'utilisateur d'acquérir des photos consécutives avec différents réglages d'exposition, qui a été principalement développées pour des applications à haute gamme dynamique [117], mais peut aussi être utilisé pour la stabilisation d'image. En utilisant deux images permet de réduire le problème mal-posé de déconvolution, et se traduit généralement par des restaurations de bien meilleure qualité que les méthodes utilisant une seule observation. De grand nombre d'algorithmes existe pour le problème général de la multiframe de déconvolution aveugle [112], [114], [118]. Le cas spécifique de déconvolution aveugle d'une paire de court et de long-exposition des images a été considéré dans [116], [119], [120]. Dans [116], la fonction de flou (PSF) est d'abord identifiée à l'aide de l'image à longue exposition et la version débruitée à courte exposition de l'image, où régularisation de Tikhonov et seuillage par hystérésis est utilisé pour régulariser la solution. Le PSF estimé est ensuite utilisé dans un procédé de restauration d'image classique [15] afin d'obtenir une estimation de l'image originale à partir de l'image prise sous longue exposition.

Une méthode d'identification conjointe est proposée dans [119], où l'image et le PSF inconnue sont estimés simultanément. L'image est modélisée en utilisant la distribution a priori avec l'expression de TV, et les PSF de flou sont estimés en imposant la contrainte que le flou dans l'image avec une exposition courte est très faible.

En effet, aucun modèle explicite de flou n'est utilisé dans ce travail, et débruitage est appliqué aux estimations par seuillage flou afin d'améliorer les estimations. Enfin, la loi a priori avec des contraintes sur les flous sont utilisés dans [120], et l'image est modélisée suivant la loi à mélange de gaussiennes (angl. Mixture-of-Gaussians) sur les dérivés d'image. Cependant, l'algorithme résultant a beaucoup de paramètres à régler, ce qui le rend difficile d'appliquer sur un large type d'images.

Mon projet aborde le problème de la déconvolution aveugle d'une paire d'images à courte et à long durée d'exposition. On a proposé une modélisation systématique des inconnus dans une nouvelle formulation bayésienne hiérarchique et de développer un algorithme de déconvolution aveugle qui estime conjointement l'image inconnue et le flou. Il utilise TV-prior sur l'image pour modéliser les statistiques d'images naturelles et à assurer la robustesse de l'algorithme. Le flou dans l'image à long durée d'exposition est modélisé en utilisant un mélange préalable qui impose à la fois la mince (angl. sparsity) et la positivité du PSF. Le flou estimé également modéliser le couplage entre les images à long et à court exposé en utilisant un modèle d'observation additionnel. Par ailleurs, il intègre une approche entièrement bayésienne, où tous les paramètres du modèle sont estimés avec les inconnus.

En conséquence, l'algorithme proposé ne nécessite l'intervention de l'utilisateur et le processus de restauration est dirigé de manière adaptative entre les images à longue et à courte durée d'exposition. Enfin, nous intégrons une analyse bayésienne variationnelle, qui fournit des estimations des distributions des inconnues. Ces distributions sont implicitement utilisées pour incorporer les incertitudes des estimations dans l'algorithme et pour compenser l'erreur. L'estimation est démontrée par les expériences sur des images synthétiques et réelles à la fois et la méthode proposée fournit des résultats de restauration haute qualité et se compare favorablement aux méthodes existantes.

## 2. Formulation du problème

Les dégradations de la paire d'images peuvent être modélisées à l'aide d'un modèle de dégradation linéaire et invariant spatiale.

Pour simplifier l'idée, on suppose que le flou est principalement causé par le tremblement de l'appareil photo pendant la durée d'exposition prolongée. Dans cette hypothèse, les processus d'observation peuvent être mathématiquement exprimés comme suit:

$$y_1 = Hx + n_1 \quad (4.1)$$

$$y_2 = x + n_2 \quad (4.2)$$

où  $y_1$  et  $y_2$  sont les images observées,  $x$  l'image d'origine inconnue,  $n_1$  et  $n_2$  les composantes de bruit. Au long de ce chapitre et le dernier chapitre, nous utilisons la notation matrice-vecteur, de sorte que les images  $y_1, y_2, x, n_1$  et  $n_2$  sont des vecteurs à dimension  $N \times 1$ , où  $N$  est le nombre de pixels de chaque image.  $H$  est le modèle matriciel de la fonction de flou (PSF)  $h$ , qui respecte la contrainte  $M \leq N$ . Notons que la construction explicite de la matrice  $H$  n'est pas nécessaire, mais il est utilisé à des fins de notation seulement.

Ces modèles d'observation seront utilisés dans le reste de rapport. Rappelons que le problème ou l'objectif de la déconvolution aveugle est de trouver  $x$  et  $h$  à l'aide  $y_1$  et  $y_2$  et de la connaissance a priori de  $x, h, n_1$  et  $n_2$ .

## 3. Modèle hiérarchique bayésien

Le modèle hiérarchique proposé se compose de deux étapes. Dans la première étape, on utilise des lois de distributions a priori sur l'image inconnue et sur le flou, et les distributions conditionnelles sont utilisés pour les observations  $y_1$  et  $y_2$ .

Dans la première étape les distributions dépendent de certains paramètres, appelés « hyperparamètres », qui sont modélisées par des distributions hyperprior dans la deuxième étape. Les formes explicites de ces distributions sont présentées dans les paragraphes suivants.

### 3.1. Première étape: Modèles d'observation

Le modèle d'observation utilise à la forme gaussienne suivant :

$$p(y_1, y_2 | x, h, \beta_1, \beta_2, \beta_{12}) = 2\pi^{-N} |\Sigma|^{-1/2} \exp \left[ -\frac{\beta_1}{2} \|y_1 - Hx\|^2 - \frac{\beta_2}{2} \|y_2 - x\|^2 - \frac{\beta_{12}}{2} \|y_1 - Hy_2\|^2 \right] \quad (4.3)$$

où  $\Sigma$  est la matrice de covariance de la distribution de  $y_1$  et  $y_2$  sachant  $x, h, \beta_1, \beta_2$  et  $\beta_{12}$  existent. Avant de calculer  $|\Sigma|$ , nous allons décrire les termes d'énergie dans cette distribution de probabilité.

1. Terme d'énergie,  $\beta_1 \|y_1 - Hx\|^2$  est l'énergie de la distribution de probabilité associée à l'observation en (4.1).
2. Seconde terme d'énergie,  $\beta_2 \|y_2 - x\|^2$  correspond à l'énergie de la distribution de probabilité associée à l'observation en (4.2).
3. Troisième terme d'énergie,  $\beta_{12} \|y_1 - Hy_2\|^2$ . On utilise une méthode de déconvolution aveugle multicanaux (voir, par exemple, [121]). En combinant (4.1) et (4.2) nous obtenons:

$$y_1 - Hy_2 | h, \beta_1, \beta_2 \sim \mathcal{N}(0, \beta_1^{-1}I + \beta_2^{-1}HH^T) \quad (4.4)$$

Notons que  $y_1 - Hy_2$  ne constitue pas une nouvelle observation et sa distribution conditionnelle est obtenue à partir de  $y_1$  et  $y_2$  déjà observé. Toutefois,  $y_1 - Hy_2$  fournit des informations complémentaires à celles fournies par  $y_1$  et  $y_2$ .

Le modèle conditionnel (4.3) ne correspond pas exactement aux modèles présentés dans (4.1) et (4.2), mais il est encore une distribution de probabilité conditionnelle et il peut donc être utilisé en combinaison avec les modèles précédents de l'image et le flou pour effectuer l'inférence bayésienne sur le problème à double exposition. Notons que le troisième terme d'énergie dans le modèle d'observation ci-dessus fournit des informations sur la fonction de flou.

Passons maintenant au calcul de la fonction de partition. Nous avons la matrice de covariance

$$\Sigma = \begin{pmatrix} (\beta_1 + \beta_{12})I & -\beta_{12}H \\ -\beta_{12}H^T & \beta_2 I + \beta_{12}H^T H \end{pmatrix}^{-1} \quad (4.5)$$

et par conséquent

$$|\Sigma|^{-1} = |(\beta_1 + \beta_{12})I| \left| \beta_2 I + \beta_{12}H^T H - \frac{\beta_{12}^2}{\beta_1 + \beta_{12}} H^T H \right| \quad (4.6)$$

Ensuite, puisque  $\log |\beta_2(\beta_1 + \beta_{12})I + \beta_{12}\beta_1H^TH|$  change lentement par rapport au changement de  $H^TH$ , on obtient:

$$|\Sigma|^{-1/2} \approx |(\beta_1\beta_2 + \beta_1\beta_{12} + \beta_2\beta_{12})I|^{1/2} \quad (4.7)$$

Enfin, pour un modèle d'observation produit par ces trois modèles indépendants et génère  $2N$  observations, chacun d'entre eux avec le même nombre d'observations,  $2N/3$  on peut écrire à partir de (4.7) :

$$|\Sigma|^{-1/2} \approx (\beta_1\beta_2\beta_{12})^{N/3} \quad (4.8)$$

Par conséquent, on peut réécrire (4.3):

$$p(y_1, y_2 | x, h, \beta_1, \beta_2, \beta_{12}) \propto \beta_1^{N/3} \beta_2^{N/3} \beta_{12}^{N/3} \exp\left[-\frac{\beta_1}{2} \|y_1 - Hx\|^2\right] \exp\left[-\frac{\beta_2}{2} \|y_2 - x\|^2\right] \exp\left[-\frac{\beta_{12}}{2} \|y_1 - Hy_2\|^2\right] \quad (4.9)$$

Il est important de noter que le modèle d'observation supplémentaire intègre la forte dépendance entre les observations et dans la procédure d'inférence (voir [122] pour une approche similaire).

### 3.2. Première étape: Modèle a priori de flou

Depuis le flou est principalement causé par le tremblement de l'appareil photo pendant le long temps d'exposition, il présente les caractéristiques du flou de mouvement non uniforme. Par conséquent, il devrait être très mince (sparse), c'est à dire, la plupart des coefficients de PSF étant nul ou très faible. Afin d'exploiter cette information, nous utilisons un mélange de prior D des distributions exponentielles sur chaque coefficient PSF,

$$p(h|\{\tau_{jd}\}) = \prod_{j=1}^M \left( \sum_{d=1}^D \tau_{jd} \text{Expon}(h_j|\sigma_{jd}) \right) \quad (4.10)$$

avec  $\tau_{jd}$  les coefficients de mélange (mixture coefficient) pour chaque pixel  $j$  et

$$\text{Expon}(h_j|\sigma_{jd}) = \begin{cases} \sigma_{jd} \exp(-\sigma_{jd}h_j), & \text{if } h_j \geq 0 \\ 0, & \text{if } h_j < 0 \end{cases} \quad (4.11)$$

Avec  $\sigma_{jd}$  les paramètres de chaque distribution exponentielle.

Notons que ce modèle a priori de flou appliqué renforce le degré de rareté (sparsity). Or d'après la notion de probabilité une augmentation de rareté conduit à l'augmentation de nombre de coefficients de mélange  $D$  [123].

En plus (4.11) impose également la positivité des coefficients de flou. Cette propriété donne un particulier important de ce modèle ; car contrairement à la plupart des travaux antérieurs la contrainte de positivité est imposée lors de la formulation. En plus le processus d'optimisation s'impose ultérieure, et non artificiellement après l'optimisation, qui peut déplacer les estimations loin de leurs valeurs optimales. Notez que ce mélange-de-exponentielles a été déjà utilisé dans [38], [123].

### 3.3. Première étape: Modèle a priori de l'image

L'image inconnue attendue peut être lisse, essentiellement aux emplacements des discontinuités (i.e. frontières ou bords). Par conséquent, dans le modèle a priori de l'image, nous utilisons la fonction de la TV, car elle préserve les bords de l'image sans pénaliser les discontinuités tout en imposant le lissage dans [56]. Plus précisément, nous utilisons l'approximation suivante de la TV dans [124]

$$p(x|\alpha_{im}) = cte \alpha_{im}^{\frac{N}{2}} \exp \left[ -\frac{1}{2} \alpha_{im} TV(x) \right] \quad (4.12)$$

Où cte est une constante et

$$TV(x) = \sum_{j=1}^N \sqrt{(\Delta_j^h(x))^2 + (\Delta_j^v(x))^2} \quad (4.13a)$$

Les opérateurs  $\Delta_j^h(x)$  et  $\Delta_j^v(x)$  correspondent, respectivement, à la dérivé horizontale et verticale de premier ordre, de pixel j, en d'autre terme  $\Delta_j^h(x) = x_j - x_{l(j)}$  et  $\Delta_j^v(x) = x_j - x_{a(j)}$  où l(j) et a(j) représentent les voisins les plus proches de j vers la gauche et en haut, respectivement.

On note que la forme en (4.13a) est appelée forme isotropique comme déjà défini en section 3.

En mathématique, TV est utilisée pour éviter la discontinuité de signal surtout quand on utilise l'opérateur gradient. Là au niveau de traitement d'image on appelle l'expression de TV pour éviter le lissage et bien sur une vision continue de l'image. Et comme le forme (4.13a) est un peu complexe en application numérique, nous proposons une forme anisotropique:

$$TV(x) = \sum_{j=1}^N |\Delta_j^h(x)| + |\Delta_j^v(x)| \quad (4.13b)$$

### 3.4. Deuxième étape: Hyperpriors en fonction des hyperparamètres

En deuxième étape du modèle hiérarchique, nous modélisons les hyperparamètres  $\alpha_{im}, \beta_1, \beta_2, \beta_{12}, \{\sigma_{jd}\}$  et  $\{\tau_{jd}\}$  par des distributions hyperpriors.

Dans les modèles bayésiens, les distributions hyperpriors sont généralement choisis pour être des distributions conjuguées, c'est à dire, ils ont la même forme fonctionnelle que le produit de la distribution conditionnelle et les priors. Ce choix de hyperpriors simplifie le calcul analytique de la procédure d'inférence. Sur la base de cela, nous employons des distributions Gamma conjugués pour les hyperpriors  $p(\alpha_{im}), p(\beta_1), p(\beta_2), p(\beta_{12}), p(\sigma_{jd})$  et les distributions de Dirichlet pour les coefficients de mélange  $p(\tau_{jd})$ , qui est présentée dans (4.14) - (4.19)

$$\begin{aligned} p(\alpha_{im}) &= \text{Gamma}(\alpha_{im} | a_{\alpha_{im}}^0, b_{\alpha_{im}}^0) \\ &= \frac{(b_{\alpha_{im}}^0)^{a_{\alpha_{im}}^0}}{\Gamma(a_{\alpha_{im}}^0)} \alpha_{im}^{a_{\alpha_{im}}^0 - 1} \exp[-\alpha_{im} b_{\alpha_{im}}^0] \end{aligned} \quad (4.14)$$

$$p(\beta_1) = \text{Gamma}(\beta_1 | a_{\beta_1}^0, b_{\beta_1}^0) \quad (4.15)$$

$$p(\beta_2) = \text{Gamma}(\beta_2 | a_{\beta_2}^0, b_{\beta_2}^0) \quad (4.16)$$

$$p(\beta_{12}) = \text{Gamma}(\beta_{12} | a_{\beta_{12}}^0, b_{\beta_{12}}^0) \quad (4.17)$$

$$p(\sigma_{jd}) = \text{Gamma}(\sigma_{jd} | a_{\sigma_{jd}}^0, b_{\sigma_{jd}}^0), \quad j = 1, \dots, M, d = 1, \dots, D \quad (4.18)$$

$$p(\{\tau_{jd}\}_{d=1}^D) = \text{Dirichlet}(\{\tau_{jd}\}_{d=1}^D | c_{\tau_{jd}}^0), \quad j = 1, \dots, M \quad (4.19)$$

avec les paramètres de forme (angl. shape parameter) sont  $a_{\alpha_{im}}^0, a_{\beta_1}^0, a_{\beta_2}^0, a_{\beta_{12}}^0$  et paramètres d'échelle inverse sont  $b_{\alpha_{im}}^0, b_{\beta_1}^0, b_{\beta_2}^0, b_{\beta_{12}}^0$ . Les paramètres de forme et d'échelle inverse des distributions Gamma sont fixés à une valeur petite commune par rapport à la gamme de valeurs d'intensité de l'image (par exemple, 0.001 dans la gamme d'image  $x \in [0, 255]$ ), et  $c_{\tau_{jd}}^0$  est mis à 1 pour obtenir hyperpriors vagues qui rendent le processus d'estimation.

Notons, cependant, que ces distributions hyperpriors sont très flexibles pour intégrer les informations supplémentaires fournies par l'utilisateur. Si une certaine connaissance de la valeur de certains des hyperparamètres (par exemple, sur les variances de bruit dans les images observées) est disponible, cette information peut facilement être intégré dans la procédure d'estimation en choisissant la forme et les paramètres d'échelle inverse des distributions correspondantes en conséquence (voir, par exemple, [124], [125] pour une telle incorporation des connaissances antérieures).

Par ailleurs, notez que l'utilisation des valeurs non nulles pour la forme et l'inverse d'échelle aide à éviter les solutions triviales telles que les estimations PSF delta. Notons également que le modèle d'observation supplémentaire (4.9) a un rôle important dans la prévention du flou d'être estimée comme une fonction delta à mois  $y_1 - y_2$  peut être considérée comme un bruit gaussien indépendant de  $y_1$  et  $y_2$ .

Enfin, la combinaison du premier et deuxième étage du modèle hiérarchique, donne:

$$p(x, h, y_1, y_2, \alpha_{im}, \beta_1, \beta_2, \beta_{12}, \{\sigma_{jd}\}, \{\tau_{jd}\}) = p(x|\alpha_{im})p(h|\{\tau_{jd}\})p(y_1, y_2|x, h, \beta_1, \beta_2, \beta_{12}) \\ \times p(\alpha_{im})p(\beta_1)p(\beta_2)p(\beta_{12}) \prod_{j=1}^M \left[ p(\{\tau_{jd}\}_{d=1}^D) \prod_{d=1}^D p(\sigma_{jd}) \right] \quad (4.20)$$

#### 4. Inférence bayésienne variationnelle

Dans les formulations de l'approche Bayésienne, l'inférence est basée sur la distribution a posteriori, ce qui dans notre cas est intraitable. Par conséquent, dans ce travail, nous utilisons des approximations variationnelles de distribution. Soit  $\theta$  l'ensemble des inconnues, c'est-à-dire  $\theta = \{x, h, \alpha_{im}, \beta_1, \beta_2, \beta_{12}, \{\sigma_{jd}\}, \{\tau_{jd}\}\}$ . L'objectif est d'approximer la distribution a posteriori  $p(\theta|y_1, y_2)$  par une autre distribution  $q(\theta)$  qui est une analyse traitable.

L'approximation de la distribution  $q(\theta)$  est obtenue en minimisant la distance de Kullback-Leibler (KL) divergence entre  $q(\theta)$  et  $p(\theta|y_1, y_2)$  qui est donnée par

$$C_{KL}(q(\theta)||p(\theta|y_1, y_2)) = \int q(\theta) \log \left( \frac{q(\theta)}{p(\theta|y_1, y_2)} \right) d\theta \quad (4.21)$$

$$= \int q(\theta) \log \left( \frac{q(\theta)}{p(\theta, y_1, y_2)} \right) d\theta + \text{constante} \quad (4.22)$$

En général, la seule hypothèse dans l'analyse bayésienne variationnelle est la factorisation. Dans ce travail, nous utilisons la factorisation suivante:

$$q(\theta) = q(x)q(h)q(\alpha_{im})q(\beta_1)q(\beta_2)q(\beta_{12}) \prod_{j=1}^M \left[ q(\{\tau_{jd}\}_{d=1}^D) \prod_{d=1}^D q(\sigma_{jd}) \right] \quad (4.23)$$

où

$$q(h) = \prod_{j=1}^M q(h_j) \quad (4.24)$$

Malheureusement, les résultats généraux de l'analyse bayésienne variationnelle ne peuvent pas être directement utilisées dans ce travail, étant que le TV et le mélange a priori dans notre modèle rend le calcul de la divergence KL dans (4.22) impossible. Les problèmes causés par le TV peuvent être évité en utilisant une approche majorization-minimisation (MM), dont les détails sont donnés dans [124].

Une procédure itérative utilise pour converge vers la meilleure approximation de la vraie distribution a posteriori  $p(\theta|y_1, y_2)$  par des distributions de la forme (4.23). Cette procédure itérative fournit des estimations  $q(\theta)$  de la distribution des inconnues. Dans ce travail, nous utilisons les moyens de ces distributions que les estimations ponctuelles des inconnues.

## 5. Calcul approximatif de la distribution a posteriori

Dans cette section, nous présentons les formes explicites de chaque distribution. Dans ce qui suit, les moyens des distributions seront désignés par  $\langle \cdot \rangle = E_{q(\theta)}[\cdot]$ .

A partir du résultat de la section précédent, la distribution  $q(x)$  est calculée comme une distribution gaussienne multivariée, sous forme:

$$q(x) = \mathcal{N}(x | \langle x \rangle, \Sigma_x) \quad (4.25)$$

où la moyenne et la covariance sont donnés par

$$\langle x \rangle = \Sigma_x (\langle \beta_1 \rangle \langle H \rangle^T y_1 + \langle \beta_2 \rangle y_2) \quad (4.26)$$

$$\Sigma_x^{-1} = \langle \alpha_{im} \rangle (\Delta^h)^T W (\Delta^h) + \langle \alpha_{im} \rangle (\Delta^v)^T W (\Delta^v) + \langle \beta_1 \rangle \langle H^T H \rangle + \langle \beta_2 \rangle I \quad (4.27)$$

avec

$$w_j = (\langle \Delta_j^h(x) \rangle)^2 + (\langle \Delta_j^v(x) \rangle)^2, j = 1, \dots, N \quad (4.28)$$

$$W = \text{diag} \left( \frac{1}{\sqrt{w_j}} \right), j = 1, \dots, N \quad (4.29)$$

Avec  $\Delta_j^h$  et  $\Delta_j^v$  sont deux operateurs linéaires correspond à la différence de premier ordre simultanément horizontal et vertical, sur pixel  $j$ , d'une façon que:  $\Delta_j^h(x) = x_j - x_{i_j}$  (avec  $i_j$  le premier pixel voisin à gauche de  $j$ ) et  $\Delta_j^v(x) = x_j - x_{k_j}$  (avec  $k_j$  le premier pixel voisin au-dessus de  $j$ ).

La moyenne de la distribution est utilisée comme estimation de l'image, qui est calculé par application d'une méthode de gradient conjugué (CG) de (4.26). On peut voir à partir de (4.27) en ce que la matrice (4.29) est la matrice spatiale adaptative qui commande la quantité de lissage à chaque emplacement de pixel en fonction de la variation d'intensité à cet élément d'image, tel qu'il est exprimé par le vecteur représentant la variation totale de l'image estimée. Il contrôle donc le compromis entre la fidélité et le lissage d'image. De plus, les paramètres  $\langle \beta_1 \rangle$ ,  $\langle \beta_2 \rangle$  et  $\langle \beta_{12} \rangle$ , et contrôler la fidélité de l'image pour l'estimation des images observées  $y_1$  et  $y_2$ . Comme ils sont également estimés simultanément avec l'image (voir ci-dessous), le processus d'estimation est automatiquement dirigé vers l'observation plus fiable. La fiabilité est exprimée par les contraintes imposées sur les estimations d'image et le flou en utilisant leurs correspondants distributions a priori. Par exemple, si le niveau de bruit dans l'image à court exposé  $y_2$  est faible, le processus d'estimation repose davantage par l'augmentation de  $\langle \beta_2 \rangle$ , cela donne PSF et l'image estime plus tranquille (lisse).

Nous trouvons ensuite les approximations de distribution  $q(h_j)$  de flou comme distribution gaussiennes, donnés par (voir l'annexe pour plus de détails dérivation)

$$q(h_j) = \mathcal{N}^R(h_j | \hat{h}_j, \tilde{h}_j) \quad (4.30)$$

Où

$$\hat{h}_j = (\tilde{h}_j)^{-1} \left[ - \sum_{d=1}^D \langle \sigma_{id} \rangle \mu_{id} + \sum_{n=1}^N \langle \beta_1 X_{nj} \left( (y_1)_n - \sum_{\substack{1 \leq m \leq M \\ m \neq j}} X_{nm} h_m \right) \rangle \right] \\ + (\tilde{h}_j)^{-1} \left[ \sum_{n=1}^N \langle \beta_{12} (Y_2)_{nj} \left( (y_1)_n - \sum_{\substack{1 \leq m \leq M \\ m \neq j}} (Y_2)_{nm} \langle h_m \rangle \right) \rangle \right] \quad (4.31)$$

$$\tilde{h}_j = \sum_{n=1}^N \langle \beta_1 \rangle \langle X_{nj}^2 \rangle + \langle \beta_{12} \rangle \sum_{n=1}^N (Y_2)_{nj}^2 \quad (4.32)$$

où  $(.)_{ij}$  désigne l'élément d'une matrice  $(i, j)^{th}$ , et  $X, Y_2$  sont des matrices construites à partir la convolution de  $x$  et  $y_2$ , respectivement. La moyenne  $\langle h_j \rangle$  et la variance  $var(h_j)$  des distributions  $q(h_j)$  sont donnés par [123]

$$\langle h_j \rangle = \hat{h}_j + \frac{\sqrt{2}}{\sqrt{\pi \tilde{h}_j}} \frac{1}{\operatorname{erfcx}\left(-\hat{h}_j \sqrt{\frac{\tilde{h}_j}{2}}\right)} \quad (4.33)$$

$$var(h_j) = \tilde{h}_j^{-1} + \frac{1}{\sqrt{\pi \tilde{h}_j}} \frac{\hat{h}_j}{\operatorname{erfcx}\left(-\hat{h}_j \sqrt{\frac{\tilde{h}_j}{2}}\right)} \quad (4.34)$$

où  $\operatorname{erfcx}(\cdot)$  est la fonction d'erreur.

Dans la prochaine étape, nous calculons les distributions des hyperparamètres à partir de:

$$q(\alpha_{im}) = \text{Gamma}(\alpha_{im} | \bar{a}_{\alpha_{im}}, \bar{b}_{\alpha_{im}}) \quad (4.35)$$

$$q(\beta_1) = \text{Gamma}(\beta_1 | \bar{a}_{\beta_1}, \bar{b}_{\beta_1}) \quad (4.36)$$

$$q(\beta_2) = \text{Gamma}(\beta_2 | \bar{a}_{\beta_2}, \bar{b}_{\beta_2}) \quad (4.37)$$

$$q(\beta_{12}) = \text{Gamma}(\beta_{12} | \bar{a}_{\beta_{12}}, \bar{b}_{\beta_{12}}) \quad (4.38)$$

$$q(\sigma_{jd}) = \text{Gamma}(\sigma_{jd} | \bar{a}_{\sigma_{jd}}, \bar{b}_{\sigma_{jd}}), \quad j = 1, \dots, M, d = 1, \dots, D \quad (4.39)$$

$$q(\{\tau_{jd}\}_{d=1}^D) = \text{Dirichlet}\left(\{\tau_{jd}\}_{d=1}^D \mid \{\bar{c}_{\tau_{jd}}\}_{d=1}^D\right), \quad j = 1, \dots, M \quad (4.40)$$

Notez que les approximations de distribution postérieures ont les mêmes formes des distributions a priori dues à l'utilisation de lois a priori conjuguées. Nous utilisons les moyennes de ces distributions comme leurs estimations, qui sont donnés par

$$\langle \alpha_{im} \rangle = \frac{\bar{a}_{\alpha_{im}}}{\bar{b}_{\alpha_{im}}} = \frac{a_{\alpha_{im}}^0 + \frac{N}{2}}{b_{\alpha_{im}}^0 + \sum_j \sqrt{w_j}} \quad (4.41)$$

$$\langle \beta_1 \rangle = \frac{\bar{a}_{\beta_1}}{\bar{b}_{\beta_1}} = \frac{a_{\beta_1}^0 + \frac{N}{3}}{b_{\beta_1}^0 + \frac{1}{2} \langle \|y_1 - Hx\|^2 \rangle} \quad (4.42)$$

$$\langle \beta_2 \rangle = \frac{\bar{a}_{\beta_2}}{\bar{b}_{\beta_2}} = \frac{a_{\beta_2}^0 + \frac{N}{3}}{b_{\beta_2}^0 + \frac{1}{2} \langle \|y_2 - x\|^2 \rangle} \quad (4.43)$$

$$\langle \beta_{12} \rangle = \frac{\bar{a}_{\beta_{12}}}{\bar{b}_{\beta_{12}}} = \frac{a_{\beta_{12}}^0 + \frac{N}{3}}{b_{\beta_{12}}^0 + \frac{1}{2} \langle \|y_1 - y_2\|^2 \rangle} \quad (4.44)$$

$$\langle \sigma_{jd} \rangle = \frac{\bar{a}_{\sigma_{jd}}}{\bar{b}_{\sigma_{jd}}} = \frac{a_{\sigma_{jd}}^0 + \mu_{jd}}{b_{\sigma_{jd}}^0 + \mu_{jd} h_j}, \quad j = 1, \dots, M, d = 1, \dots, D \quad (4.45)$$

$$\bar{c}_{\tau_{jd}} = c_{\tau_{jd}}^0 + \mu_{jd} \quad (4.46)$$

$$\langle \tau_{jd} \rangle = \frac{\bar{c}_{\tau_{jd}}}{\sum_{d=1}^D \bar{c}_{\tau_{jd}}} \quad (4.47)$$

Les variances de ces distributions peuvent être utilisées pour évaluer la certitude de l'algorithme sur les paramètres estimés. Enfin, les variables auxiliaires  $\mu_{jd}$  sont calculés en (4.48):

$$\mu_{jd} \propto \langle \tau_{jd} \rangle \text{Expon}(\langle h_j \rangle \mid \langle \sigma_{jd} \rangle), j = 1, \dots, M \quad (4.48)$$

avec la condition

$$\sum_{d=1}^D \mu_{jd} = 1, j = 1, \dots, M \quad (4.49)$$

Les formes explicites des expressions  $\langle H^T H \rangle$  dans (4.27),  $\langle (\Delta_j^h(x))^2 + (\Delta_j^v(x))^2 \rangle$  dans (4.28),  $\langle \|y_1 - Hx\|^2 \rangle$  dans (4.42),  $\langle \|y_2 - x\|^2 \rangle$  dans (4.43), et  $\langle \|y_1 - Hy_2\|^2 \rangle$  dans (4.44) sont données par (4.50) - (4.54).

$$\langle H^T H \rangle = \langle H \rangle^T \langle H \rangle + Y_h \quad (4.50)$$

$$\langle (\Delta_j^h(x))^2 + (\Delta_j^v(x))^2 \rangle = (\Delta_j^h(\langle x \rangle))^2 + (\Delta_j^v(\langle x \rangle))^2 + \text{trace}(\Sigma_x(\Delta_j^h)(\Delta_j^h)^T + (\Delta_j^v)(\Delta_j^v)^T) \quad (4.51)$$

$$\langle \|y_1 - Hx\|^2 \rangle = \|\langle y_1 \rangle - \langle H \rangle \langle x \rangle\|^2 + \text{trace}(\langle H \rangle^T \langle H \rangle \Sigma_x) + \text{trace}(Y_h \langle x \rangle \langle x \rangle^T + \text{trace}(\Sigma_x Y_h)) \quad (4.52)$$

$$\langle \|y_2 - x\|^2 \rangle = \|y_2 - \langle x \rangle\|^2 + \text{trace}(\Sigma_x) \quad (4.53)$$

$$\langle \|y_1 - Hy_2\|^2 \rangle = \|y_1 - \langle H \rangle y_2\|^2 + \text{trace}(Y_h Y_2 Y_2^T) \quad (4.54)$$

Notez que le calcul explicite de la matrice de covariance  $\Sigma_x$  n'est nécessaire que dans (4.51) - (4.53), ce qui est peu pratique, en raison de sa grande taille  $N \times N$ . Comme mentionné ci-dessus, l'estimation d'image en (4.26) est calculée en utilisant une méthode de gradient conjugué, qui ne nécessite pas d'être construit  $\Sigma_x$ . Afin d'éviter la grande complexité de calcul de  $\Sigma_x$  dans (4.51) - (4.53), dans ces équations nous supposons approximatives que  $\Sigma_x$  une matrice diagonale obtenue par les inverses des éléments diagonaux de  $\Sigma_x^{-1}$  dans (4.27). Nous avons mené des expériences approfondies avec de petites images qui permettent la construction explicite de vérifier la validité de cette approximation. Nous avons constaté empiriquement que ce rapprochement se traduit par des estimations très similaires et a un effet mineur dans le processus d'estimation. Notez que des approximations similaires ont également été utilisées dans d'autres méthodes de récupération bayésiennes [126], [127].

L'algorithme 1 proposé est résumé par ce schéma :

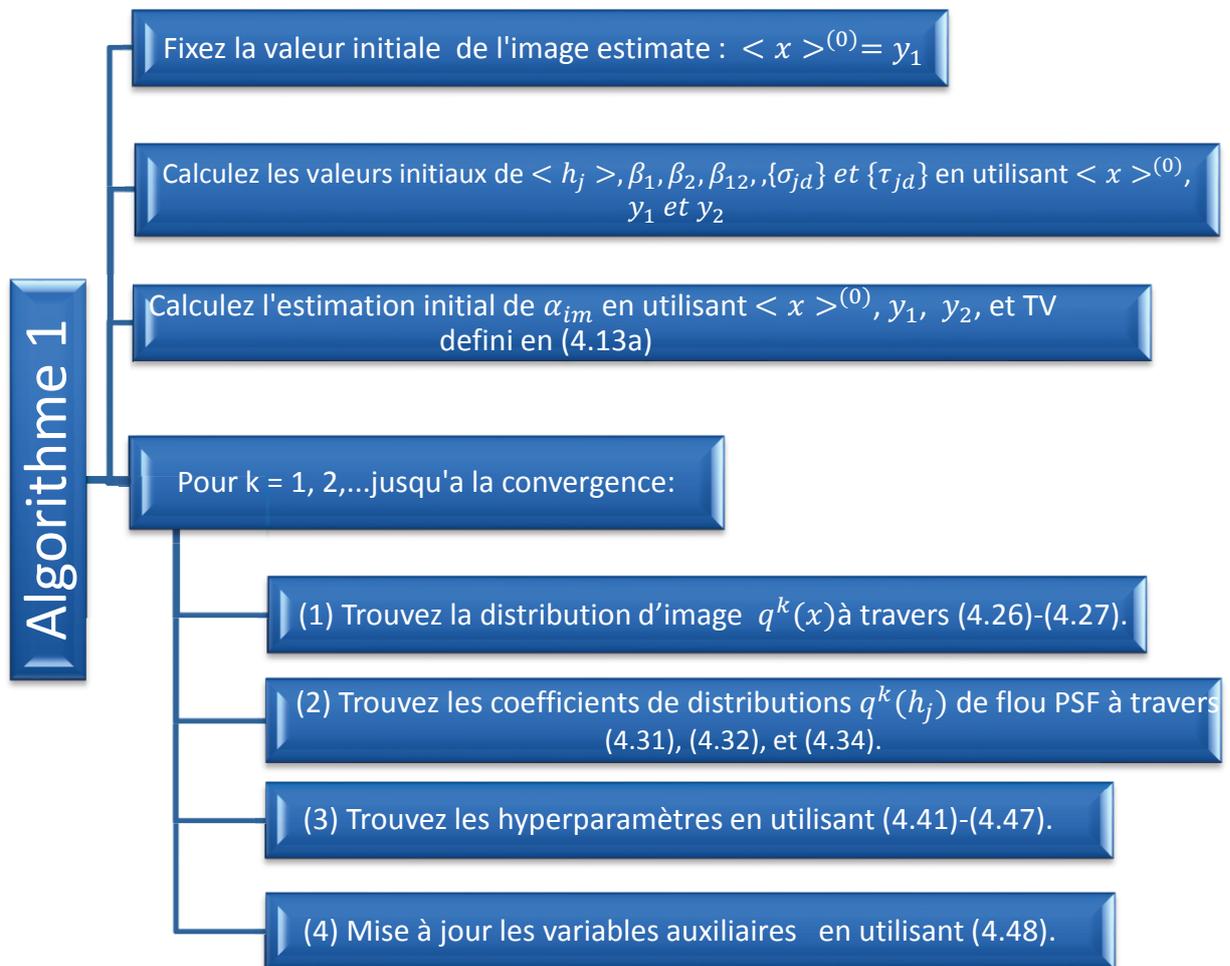


Figure IV.1. Algorithme 1: Deconvolution Aveugle Bayésienne d'un court et long temps exposition de duo images.

Le deuxième algorithme proposé dans ce travail diffère de ce qui précède, par l'expression de TV. On a une forme anisotropique défini dans (4.13b).

Alors L'algorithme2 proposé est :

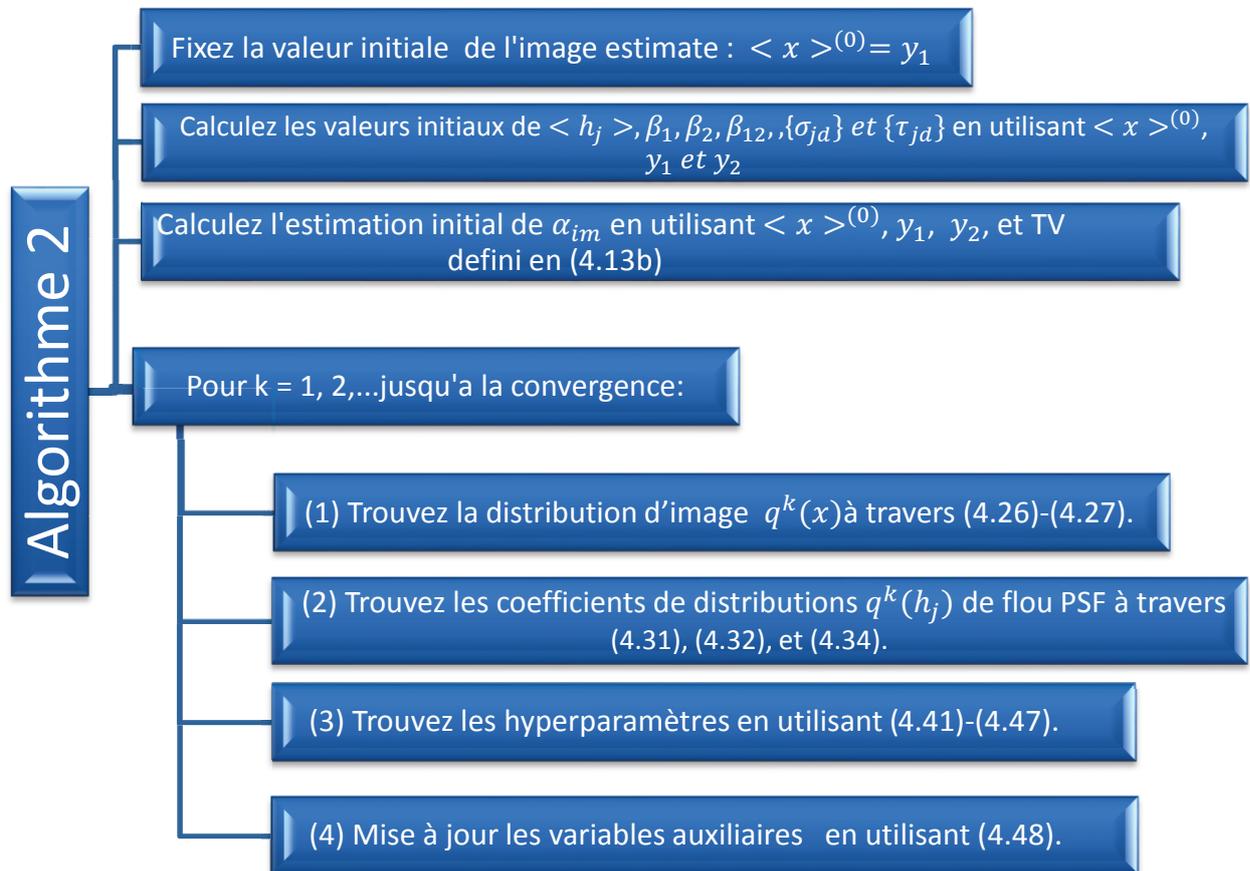


Figure IV.2. Algorithme2: Deconvolution Aveugle Bayésienne d'un court et long temps exposition de duo images.

## 6. Conclusion

Dans le cadre proposé, les distributions des variables latentes (cachées) sont estimées au lieu de leurs estimations ponctuelles, ce qui présente plusieurs avantages.

Tout d'abord, l'incertitude des estimations peuvent être calculées en examinant les variances des distributions estimées.

Deuxièmement, ces incertitudes sont incorporées dans la procédure d'estimation en utilisant les espérances indiquées dans (4.50) - (4.54), de sorte que lors de l'estimation d'un inconnu, l'algorithme met en compte les erreurs possible dans les estimations des autres variables.

Cette incorporation d'incertitudes améliore significativement les performances de l'algorithme et il est aussi très utile pour éviter le grand nombre de minima locaux résultant en faible qualité des estimations (voir [129]).

Notons que si nous souhaitons minimiser la complexité de calcul, on règle toutes les covariances dans la méthode proposée égal à zéro, et au lieu d'utiliser les moyens des distributions des variables inconnues, on peut appliquer un algorithme d'échantillonnage de tirer des valeurs différentes de la distribution. Bien que cette approche mène à une procédure beaucoup plus lente, elle peut être utilisée pour éviter les minima locaux dans les cas où les estimations initiales des inconnues sont loin d'être les solutions souhaitées.

Il convient également de noter que le cadre proposé peut être facilement étendu pour traiter un grand nombre d'images d'entrée avec éventuellement plus d'une image floue. Le cadre proposé constitue la principale base mathématique pour des cas plus généraux avec des images d'entrée multiples et des fichiers PSF de flou multiples, et les algorithmes qui en résultent sont très similaires à celui présenté dans ce travail.

# Chapitre V.

## Résultats expérimentaux

### 1. Introduction

Dans ce paragraphe, nous montrons la performance de la méthode proposée dans [114] par des expériences sur des images synthétiques et réelles.

Dans toutes nos expériences, on respecte les notations suivantes :

- a. PSF par convention est le masque de flou.
- b. Image observée  $y_1$  est utilisée comme une valeur initiale de l'image nette  $x$ .
- c. L'estimation initiale de PSF  $h$  est obtenue à partir de (4.31) avec la matrice de covariance  $\sum_x$  égale à zéro. L'algorithme est capable de fournir des résultats très précis, mais aussi très sensibles avec cette initialisation brute.
- d. La dimension  $M$  de PSF choisi est généralement beaucoup plus petite que la dimension d'image  $N$ , ce qui améliore la complexité calculatoire de la méthode. Notez que des méthodes similaires ont été utilisées par les méthodes de déconvolution, où la dimension de PSF est généralement choisie manuellement.
- e. Le nombre de distributions Gaussien mélangé (Mixture of Gaussian) est  $D = 3$ , sachant que d'autres choix ( $D = 2$  ou  $D = 4$ ) sont donnés des résultats similaires.
- f. les paramètres (4.41 jusqu'à 4.48) requis de l'algorithme sont initialisés automatiquement sauf la dimension et les valeurs initiale de PSF.
- g. Comme critère de convergence, nous utilisons  $\frac{\| \langle x \rangle^{(k)} - \langle x \rangle^{(k-1)} \|_2^2}{\| \langle x \rangle^{(k-1)} \|_2^2} < 10^{-5}$  (en terme Matlab, erreur =  $\text{norm}(x(k)-x(k-1),2)/\text{norm}(x(k-1),2)$ ), où  $\langle x \rangle^{(k)}$  et  $\langle x \rangle^{(k-1)}$  sont les images estimées aux itérations  $k$  et  $k-1$ , respectivement.
- h. La convergence est généralement obtenue en 20 itérations, où chaque itération prend environ 50 secondes à l'aide du code Matlab « Non-Optimize » sur un processeur Pentium Core i7 à 1,6 GHz.

Nous avons commencé les premiers simulations sur des images dégradés par un logiciel bien déterminé dans le paragraphe 2. Ceci, pour comparer les résultats obtenus avec ceux attendus. Et alors, nous présentons au premier pas les outils de base associés à la problématique de

dégradation d'images. En effet, nous commençons par donner des éléments PSF (masque responsable de flou), les éléments de bruit et le processus de formation des images floues. Ensuite, nous précisons les résultats obtenir dans [114]. Puis nous présentons les résultats de nos travaux.

Et pour faire des comparaisons avec d'autres approches, nous présentons le travail proposé en MATLAB version 7.12 (fonction blinddéconvolution.m).

Afin de pallier le caractère et le point fort de nos travaux, le dernier paragraphe est un synthé global de toutes les procédures et les algorithmes utilisés. C'est-à-dire, des informations utiles sur le processus de déconvolution, telles que l'amélioration du rapport signal sur bruit (SNR) comme une mesure objective de la qualité du processus.

## **2. Dégradation d'image principe et logiciel**

### **2.1. Introduction**

Dans le cadre de ce projet, nous avons développé un logiciel pour effectuer la synthèse des images bruitées et flouées.

L'objectif de cette section est d'expliquer facilement comment chacun module fonctionne. Les éléments de programme seront expliqués en détails, étape par étape à l'aide de nombreuses photos et schémas.

Tout d'abord, l'application de bruit et de flou sera expliquée avec une série du programme instantanée, et tous les menus et leurs fonctionnalités seront affichés. La dernière partie contient un exemple de l'usage qui pourrait aider l'utilisateur.

En résumé, ce logiciel dégrade une image donnée initiale, lui applique un flou et un bruit additif.

### **2.2. Flou et bruit appliqués**

Cette application effectue la fonctionnalité de dégrader une image originale avec flou et de bruit déterminée par l'utilisateur. Pour démarrer le module, il faut exécuter le "FlouAndNoise.m" fichier dans la console Matlab. Une autre option consiste à exécuter l'application Guide de Matlab, et exécutez le "FlouAndNoise.fig" fichier ensemble. Ne pas oublier de changer le répertoire courant de Matlab dans la console Matlab à celui dans lequel les fichiers sont stockés dans FlouAndNoise afin d'être en mesure de l'exécuter.

Les sections suivantes montrent comment utiliser le programme et décrire les fonctionnalités mises en œuvre à ce sujet.

### **2.3. Fenêtre principale**

Ici, nous pouvons voir le programme de flou et de bruit, et une explication brève est donnée

des boutons et des menus principaux:

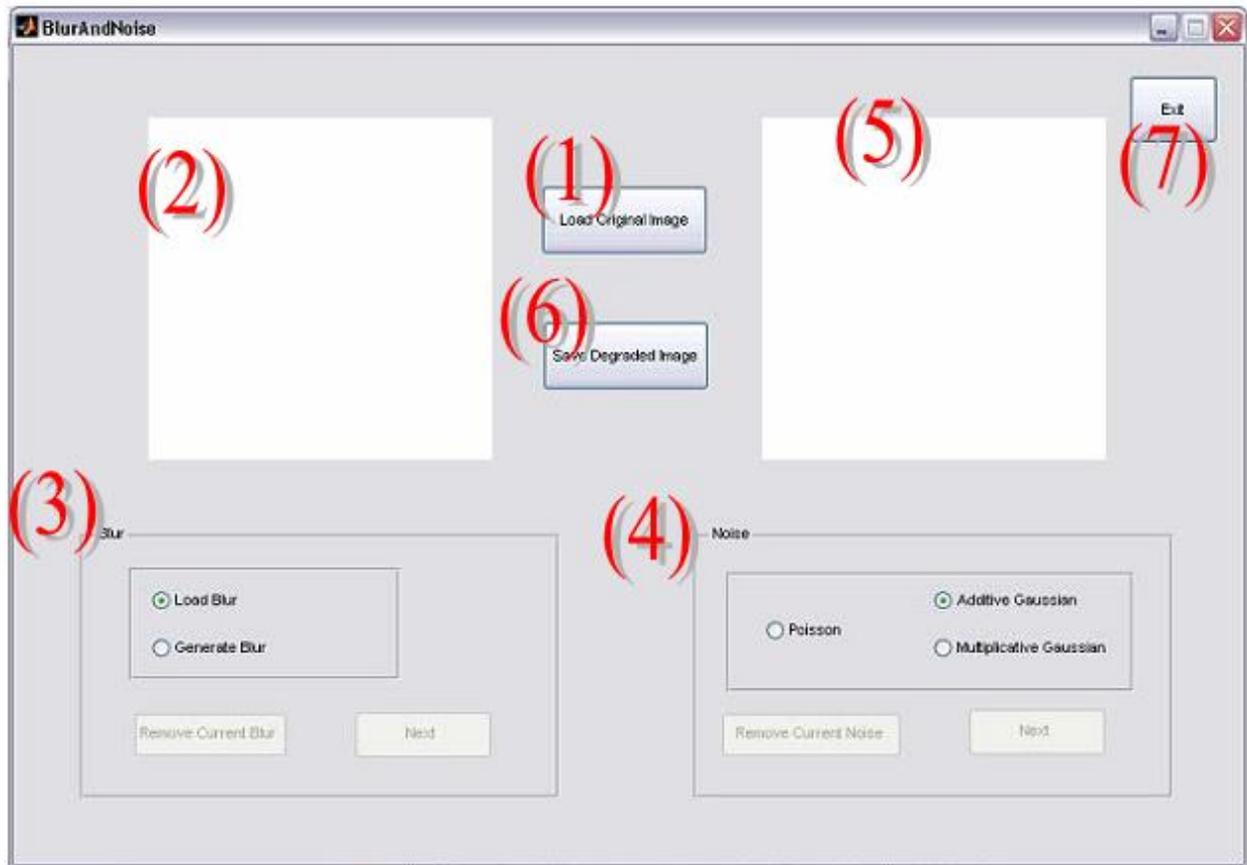
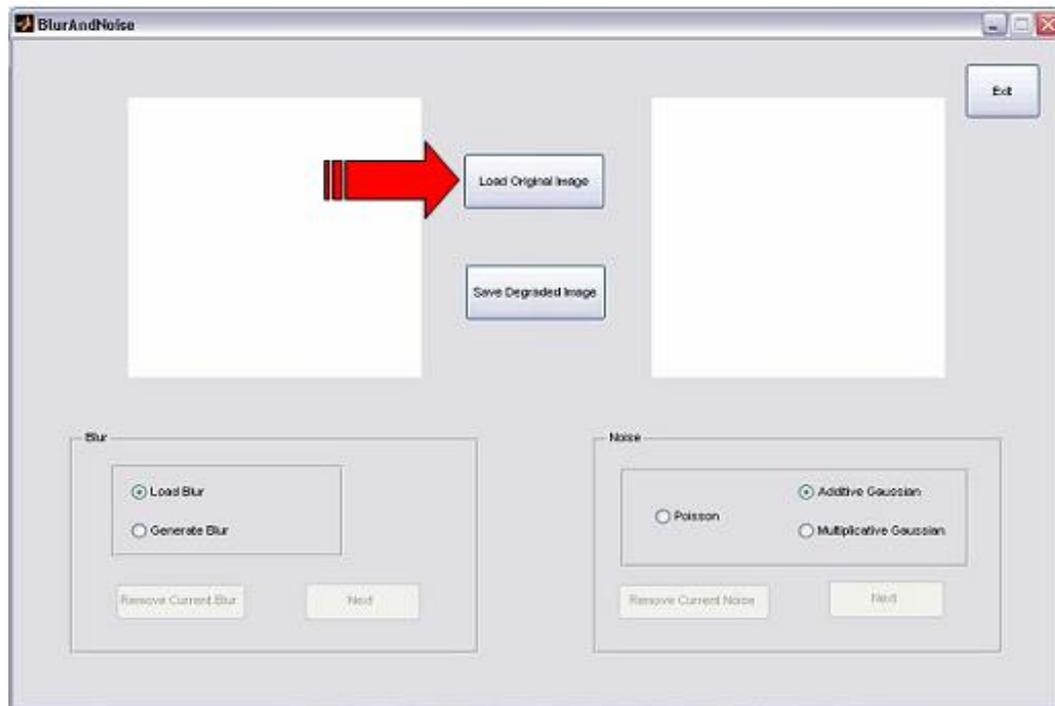


Figure V.1. Fenêtre principale de fonction Matlab " FlouAndNoise.m "

- (1) Le bouton Charger l'image originale nous permet de charger une image de l'intérieur du système.
- (2) Cet espace affiche l'image déjà chargée.
- (3) Le panneau de flou permet à l'utilisateur d'appliquer un masque flou (PSF) à l'image chargée.
- (4) Le panneau de bruit permet à l'utilisateur d'appliquer de bruit à l'image précédemment floue.
- (5) Cet espace affiche l'image observée, la première image floue et, plus tard, si tel est le cas, l'image floue et bruitée.
- (6) Le bouton Enregistrer l'image dégradée, nous permet d'enregistrer l'image observée dans le système.
- (7) Le bouton Quitter ferme l'application.

## 2.4. Charger une image

La première étape est charger une image; il suffit de cliquer sur le bouton « Charger » une image comme il est montré dans l'image:



*Figure V.2. Chargement de l'image originale.*

Ensuite, il faut choisir et ouvrir une image en utilisant le navigateur qui apparaît. Lorsque l'image est choisie, la prochaine étape est de cliquer sur le bouton « Ouvrir » dans le navigateur, comme il est montré dans l'image:

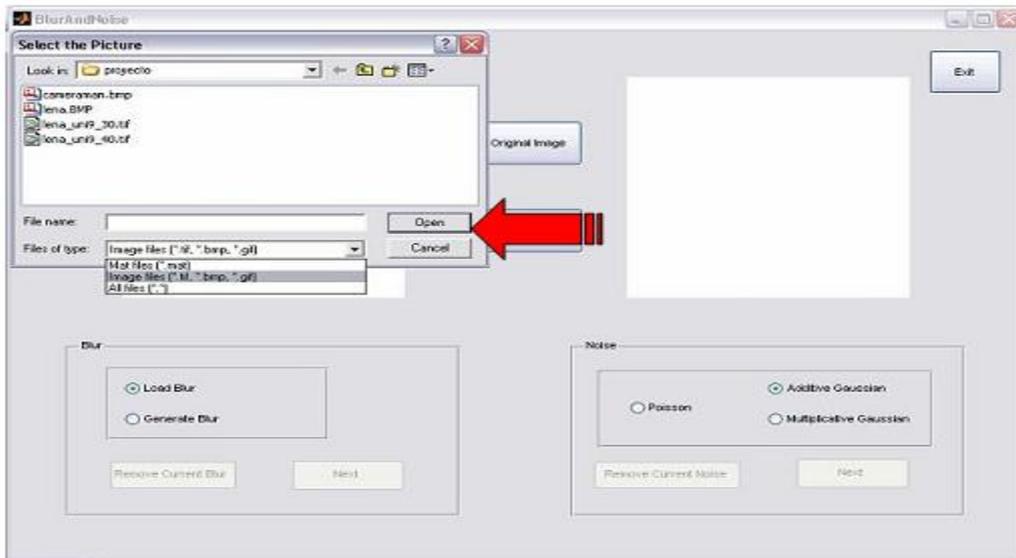


Figure V.3. Navigateur de sélection l'image original.

Lorsqu'elle est sélectionnée, l'image sera affichée dans le carré à gauche de l'interface, qui ressemble à ceci:

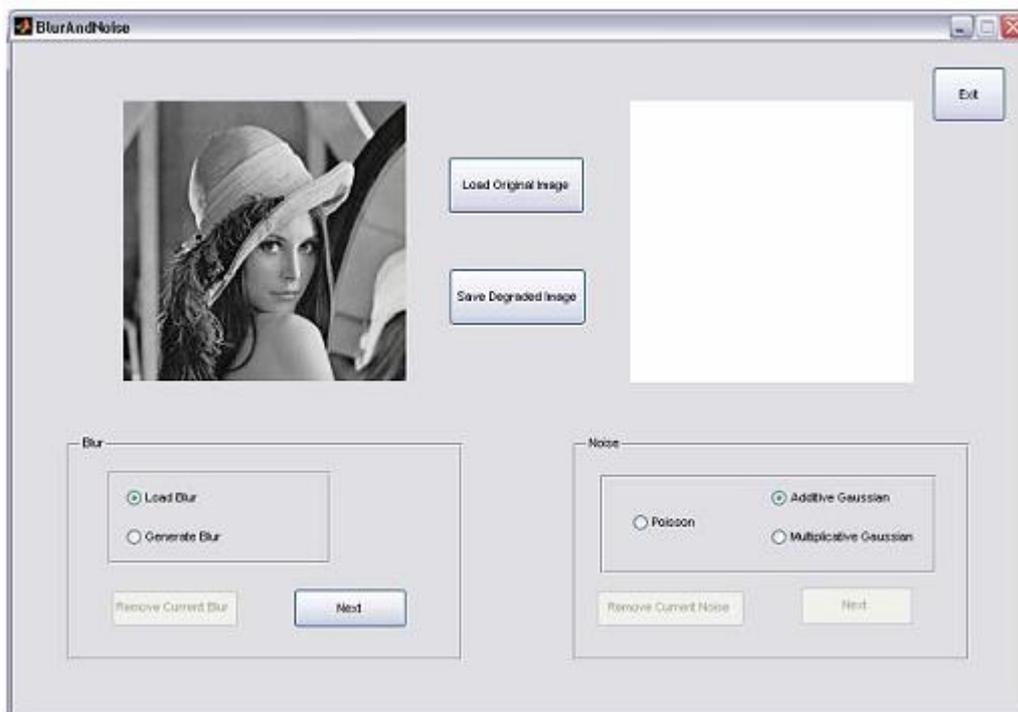


Figure V.4. Affiche l'image sélection dans la fenêtre principale.

Notez que le bouton « Suivant » est maintenant disponible dans le panneau de flou, nous

sommes maintenant en mesure de commencer à travailler sur l'image.

## 2.5. Flouage de l'image sélectionnée

Nous allons maintenant utiliser le panneau de flou à brouiller l'image chargée. Le panneau nous donne deux options:

- ❖ Charger le flou.
- ❖ Générer le flou.

### 2.5.1. Charger le flou.

Cette option permet de choisir le PSF stocké dans un fichier .mat pour l'appliquer à l'image chargée. Il doit être pris en compte du fait que le noyau flou doit être stocké dans la première variable stockée dans le fichier .mat. Si le fichier en question a été enregistré avec le flou et bruit, il n'y aura pas de problème de le charger. Nous devons cliquer sur le bouton Suivant alors que l'option flou de charge est sélectionnée, comme il est montré dans l'image suivante.

(On note que: Le centre de PSF est le centre du masque.)

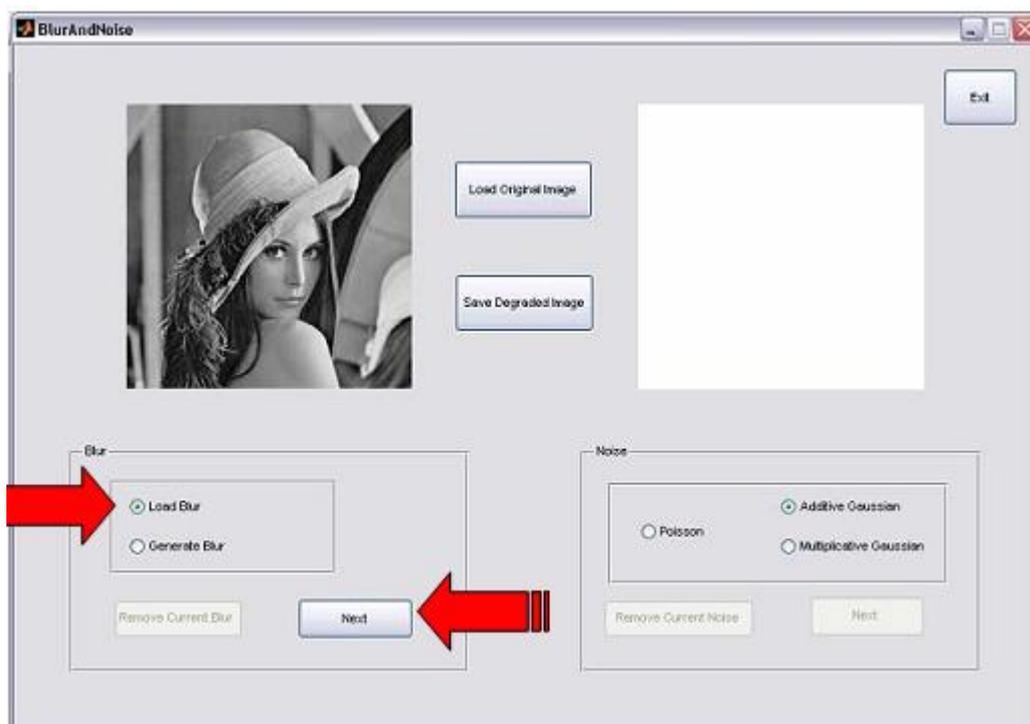


Figure V.5. Charge le flou.

La prochaine chose faire est de sélectionner le masque de flou désiré en utilisant le navigateur

qui apparaît, puis cliquez sur « Suivant » sur le panneau de Flou:

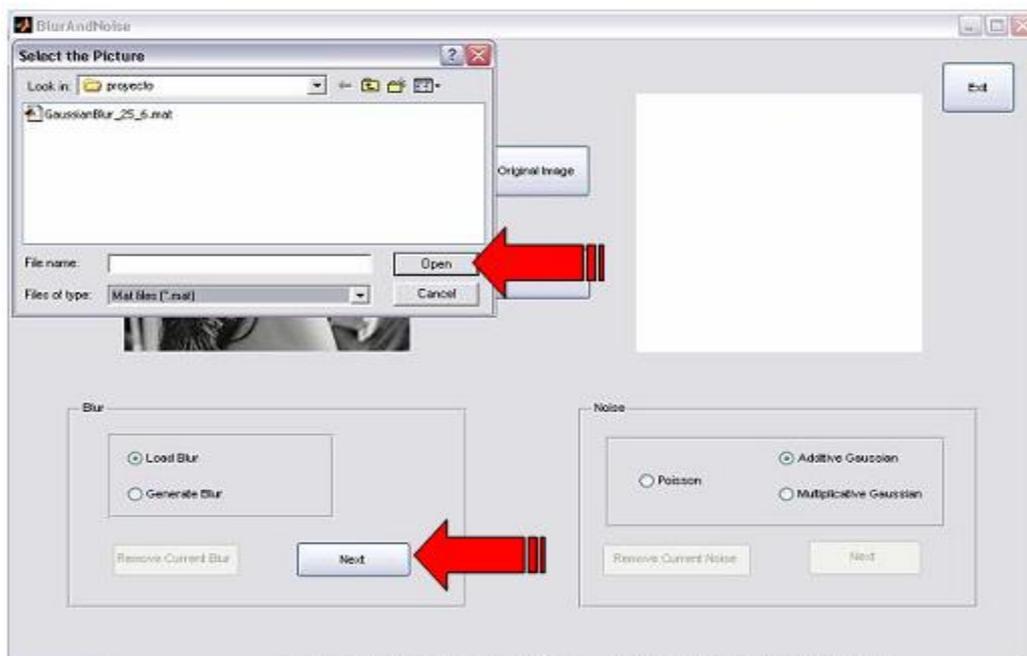


Figure V.6. Navigateur de sélection le flou désirée.

Après cela, le programme va nous donner la possibilité d'appliquer le masque sélectionné, ou revenir à la fenêtre principale et recommencer le processus de flou:

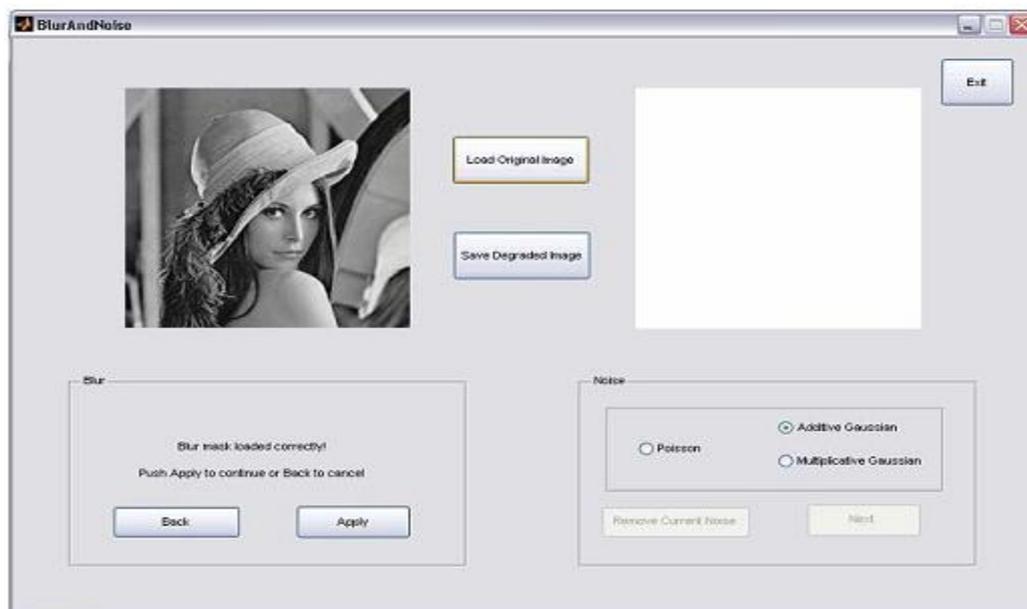


Figure V.7. Appliquer le masque sélectionné, ou revenir à la fenêtre principale pour recommencer le processus de flou.

Image floue sera affichée dans le carré de droit blanc, montrant que le flou a été appliqué avec succès.

## 2.5.2. Générer le flou

Dans le cas où nous voulons générer un masque flou au lieu de le charger à partir d'un fichier, il faut sélectionner l'option Générer de flou tel qu'il apparaît dans l'image:

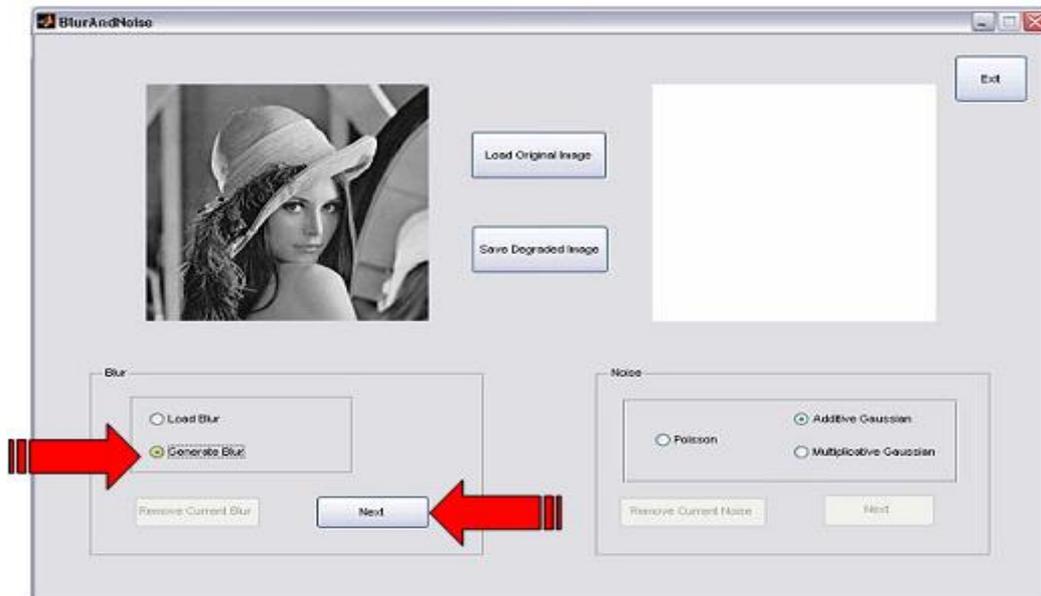


Figure V.8. Générer un masque flou (PSF).

L'interface affiche les quatre possibilités que nous avons pour le flou: gaussien, uniforme, personnalisé et de mouvement (angl. Motion Blur); l'utilisateur doit sélectionner un, puis cliquer sur « Suivant ». Comme indique la Fig. V.9.

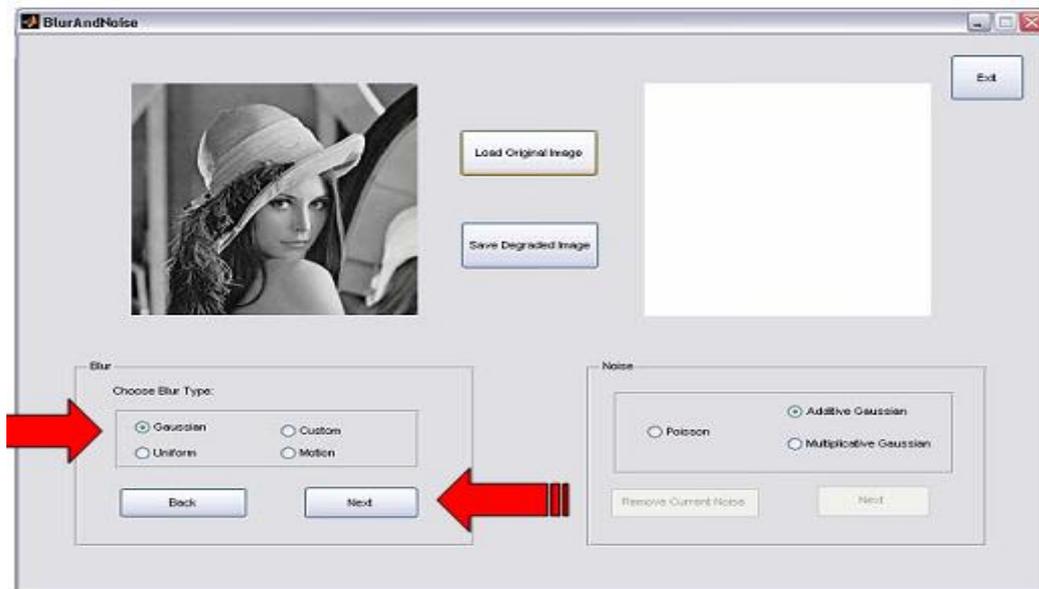


Figure V.9. Panneau de sélection un de quatre genre de flou.

### 2.5.2.1. Flou gaussien

Pour appliquer ce flou, il faut introduire deux paramètres qui définissent le flou: Taille et Sigma. Cela crée une dimension = [Taille x Taille] filtre à noyau gaussien avec variance égale Sigma.

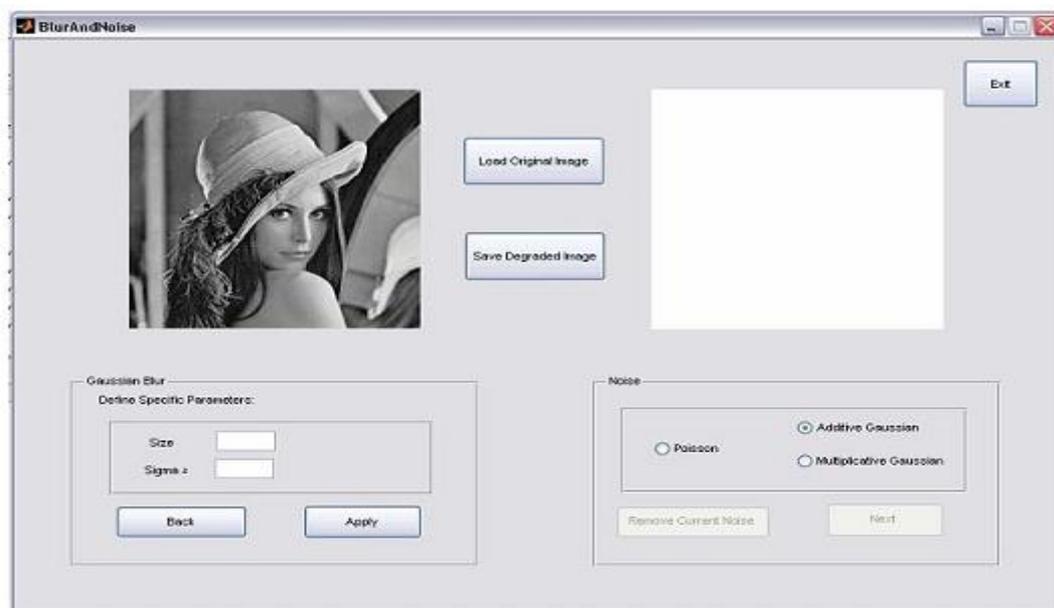


Figure V.10. Panneau de choisir de flou gaussienne.

### 2.5.2.2. Flou uniforme.

Ce flou nécessite un paramètre qui est de la taille du noyau flou, et donc l'interface nous demande:

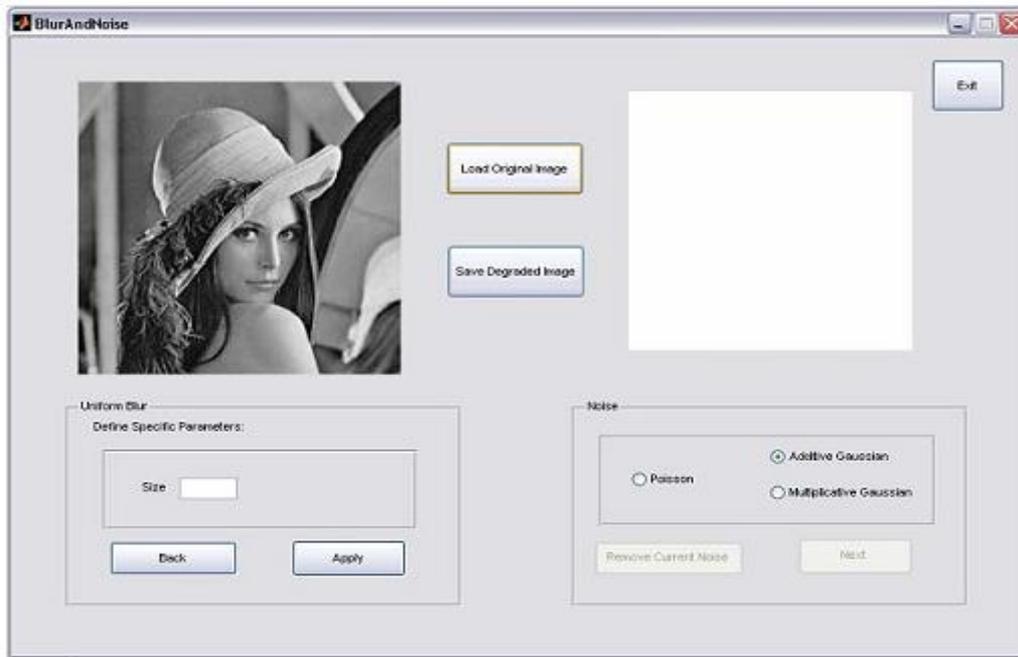


Figure V.11. Panneau de choisir de flou uniforme.

### 2.5.2.3. Flou personnalisé.

Cette option nous donne la possibilité de configurer un noyau flou selon nos propres préférences, nous donnant une matrice 7x7 pour le spécifier. Le noyau doit être centré dans le milieu de la place (4<sup>th</sup> ligne, 4<sup>th</sup> colonne).

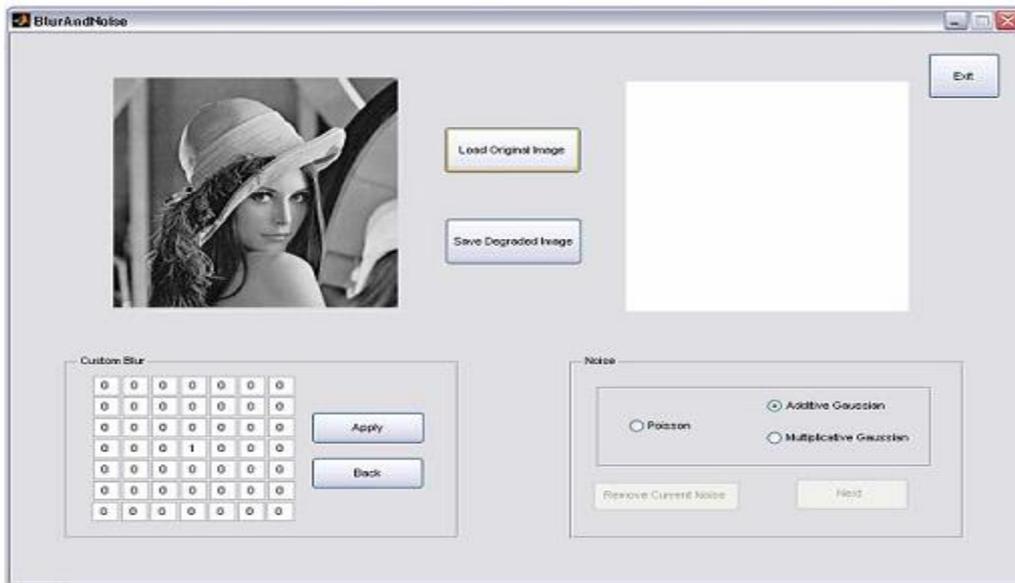


Figure V.12. Panneau de choix personnel de flou.

#### 2.5.2.4. Flou de mouvement (Motion Blur)

Pour appliquer ce genre de flou, il faut introduire deux paramètres qui définissent le flou: Taille et Theta. Le flou de mouvement se rapproche du mouvement linéaire de la caméra par Taille pixels, avec un angle de degrés  $\theta$  dans le sens inverse des aiguilles d'une montre. C'est pour cela que l'utilisateur doit introduire la taille en pixels et l'angle en degrés pour obtenir des résultats corrects:

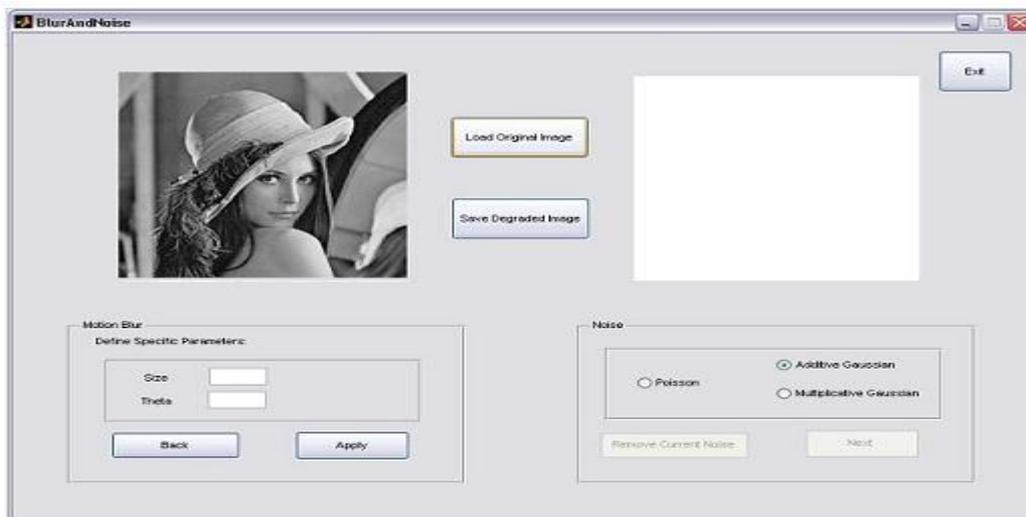


Figure V.13. Panneau de choisir de flou du mouvement.

### 2.5.3. Situation après l'application de flou

Après que le flou ait été appliqué, nous allons retourner au panneau principal avec quelques modifications. Dans l'image suivante, on donne des détails de la nouvelle situation:

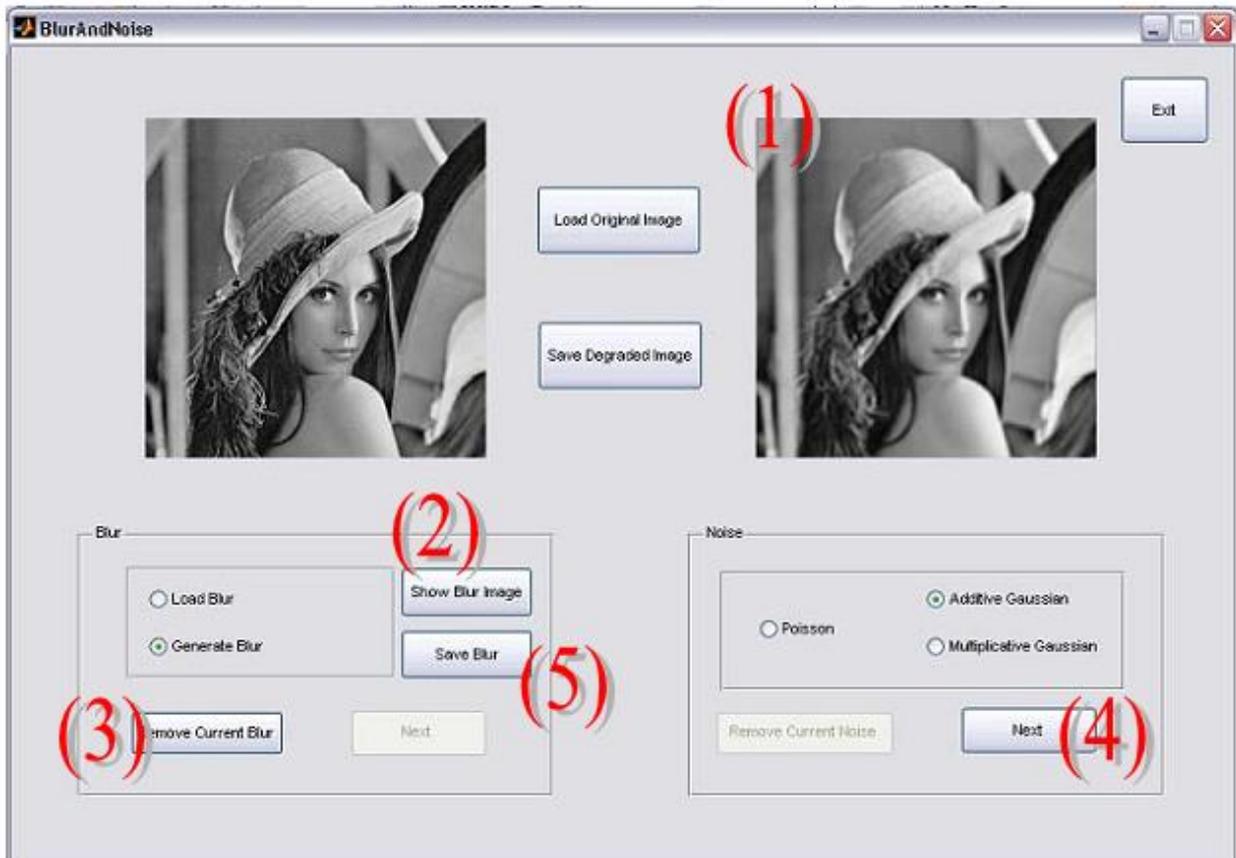


Figure V.14. Situation après l'application de flou.

- (1) Nous avons ici l'image floue.
- (2) Ce bouton permet de voir la matrice de flou comme une image.
- (3) Le bouton supprime le masque flou actuel et nous permet de recommencer depuis le début.
- (4) Le bouton Suivant dans le panneau de bruit est maintenant actif, afin que nous puissions appliquer le bruit à l'image floue.
- (5) Avec le bouton Enregistrer le flou, nous avons la possibilité de sauvegarder le flou appliqué dans un fichier .mat.

### 2.6. Application du bruit à une image

La deuxième transformation que nous pouvons faire à l'image est une application bruit. Nous montrons maintenant comment ajouter du bruit à l'image floue. L'interface nous donne

différentes possibilités: le bruit de Poisson, bruit Gaussien additif ou bruit Gaussien multiplicatif.

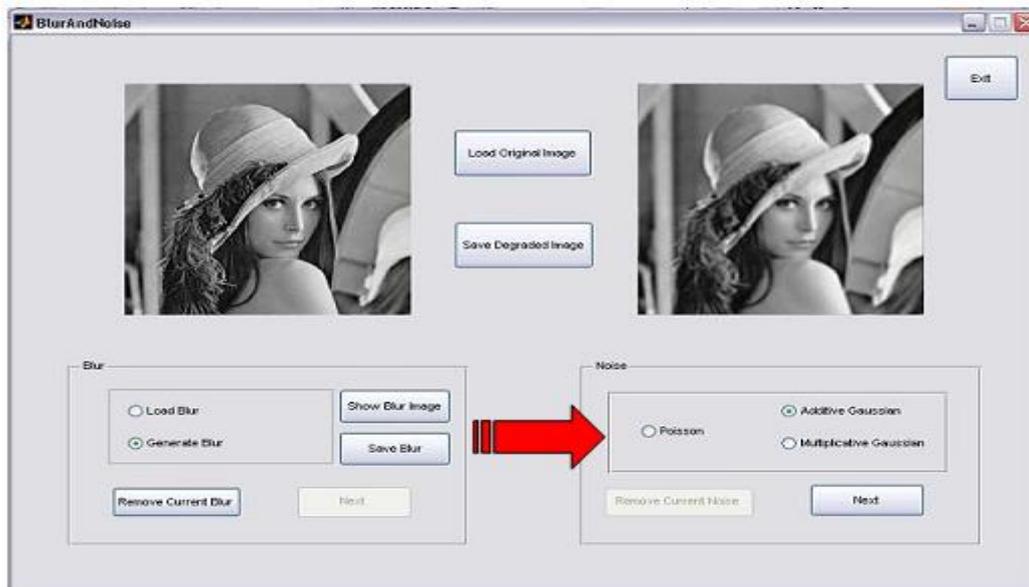


Figure V.15. Panneau de choisir entre les différents genres de bruit.

### 2.6.1. Bruit de Poisson

Ce type de bruit ne nécessite aucun type de paramètres à fournir, si l'interface directement nous demander une confirmation, en cliquant sur « Appliquer », comme il est montré dans l'image:

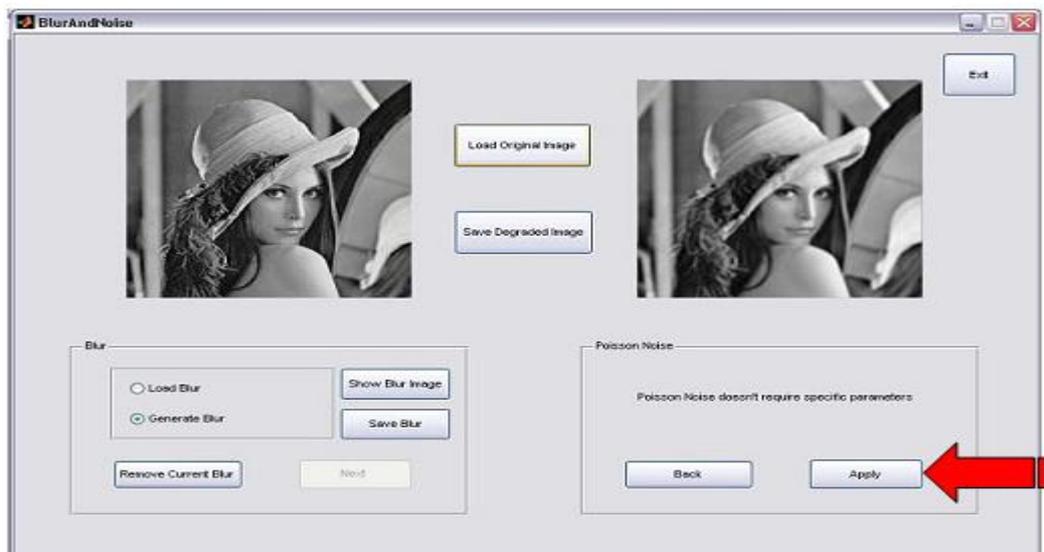


Figure V.16. Applique le bruit poisson sur l'image floue.

Après avoir appuyé sur « Appliquer », la fenêtre de l'image de droite sera mise à jour avec le bruit désiré ajouter à l'image floue.

## 2.6.2. Bruit Gaussien additif

Afin d'introduire ce type de bruit, il faut fournir la variance du bruit ou le rapport signal-bruit. Si on appuie sur la touche « Entrée » dans notre clavier après l'introduction de l'un des paramètres, l'autre sera automatiquement actualisé, donc nous allons voir si le bruit est celui de votre choix avant de l'appliquer sur l'image:

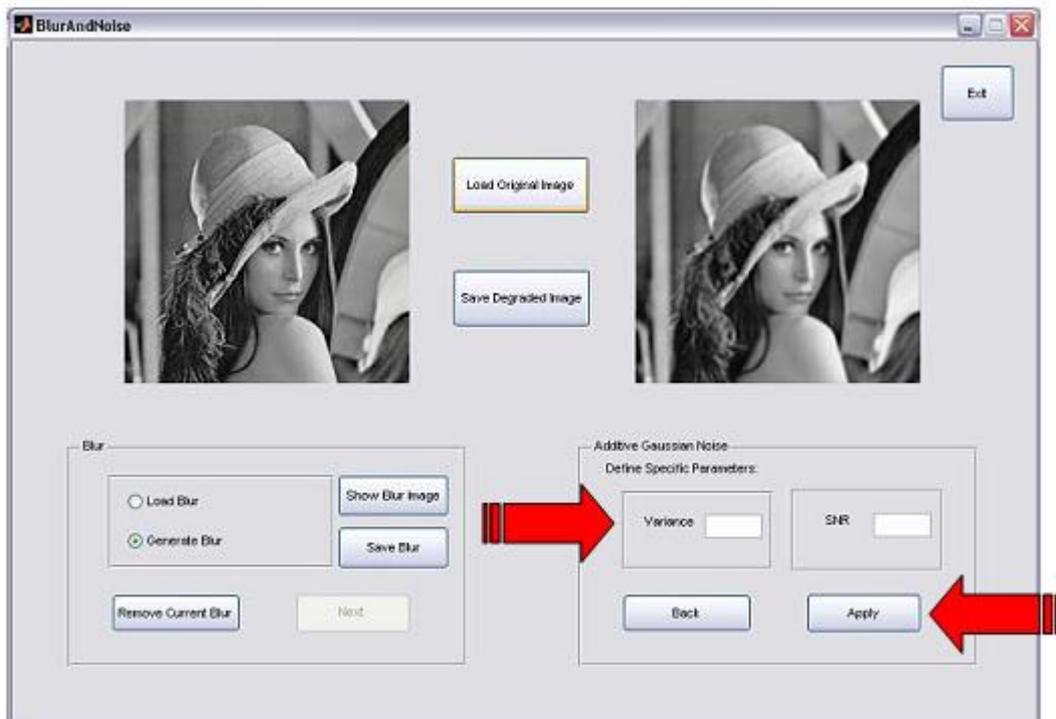


Figure V.17. Applique le bruit Gaussien additif sur l'image floue.

## 2.6.3. Bruit Gaussien multiplicatif

Ce type de bruit nécessite le paramètre multiplié par l'observation floue correspondent à la variance de l'observation bruitée, il sera demandé tel qu'il est indiqué ci-dessous:

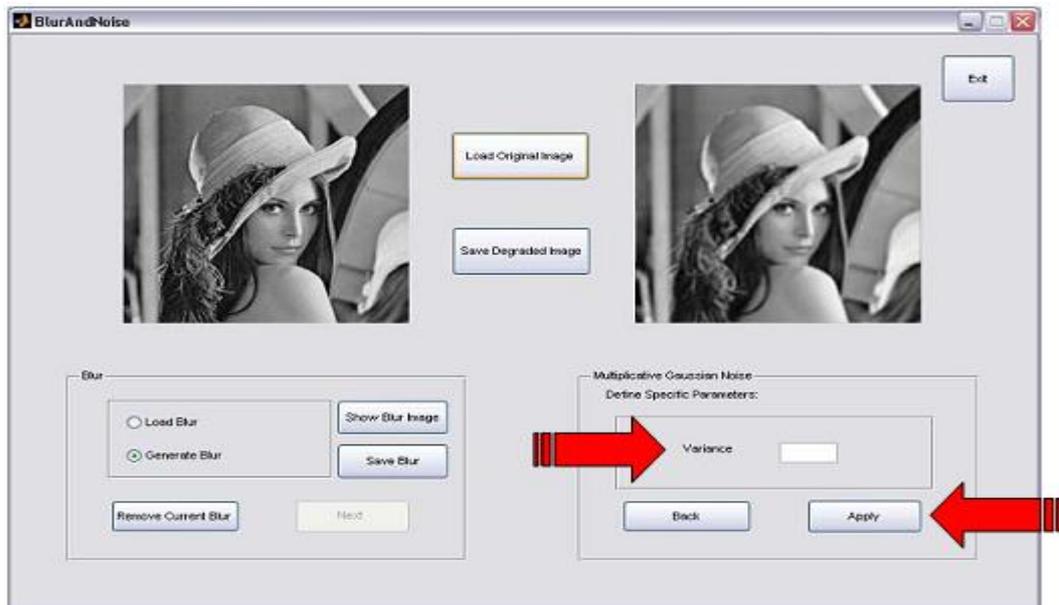


Figure V.18. Bruit Gaussien appliqué sur l'image floue.

#### 2.6.4. Situation après l'application du bruit

Après avoir appliqué le bruit sur l'image floue, l'interface ressemble à l'image ci-dessous. Notez que nous ne serons pas en mesure d'introduire un autre type de bruit dans l'image à moins que le « supprime le bruit actuelle » bouton est cliqué, dans ce cas, l'interface permet de supprimer le bruit en cours et nous serons en mesure de commencer dès le début du processus de gérer le bruit.

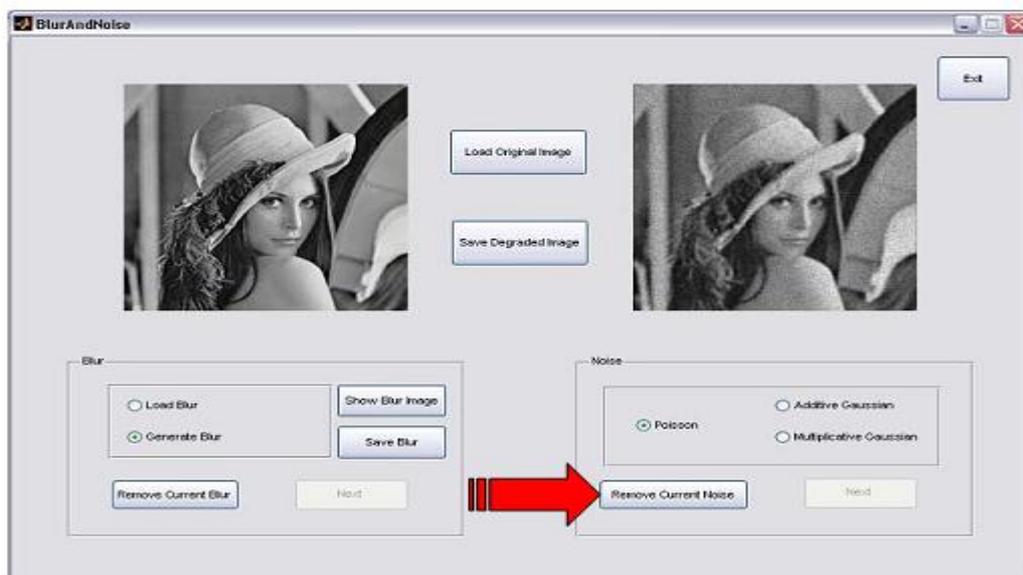


Figure V.19. Affichage de l'image floue et bruitée.

## 2.7. Résultats sauvegardé

Nous sommes en mesure d'enregistrer l'image dégradée à tout moment dans le processus de dégradation, seulement en appuyant sur le bouton « Enregistrer l'image dégradée », comme il est montré dans l'image. Gardez à l'esprit que l'image en cours d'enregistrement est celle montrée sur la place du droit de l'interface:

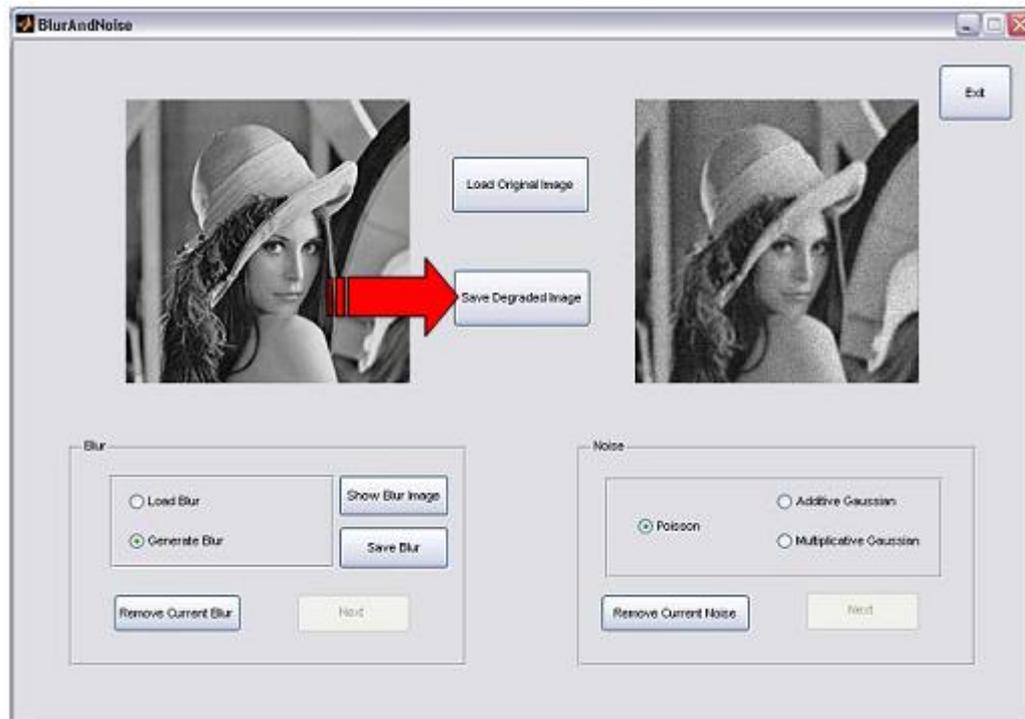


Figure V.20. Enregistrer l'image dégradée.

Une fenêtre d'enregistrement apparaît et qui nous donne en mesure de spécifier le nom et l'extension de l'image. L'extension .mat sera donnée par défaut, mais nous pouvons choisir n'importe quel autre format de fichier en l'indiquant dans la fenêtre à la fin du nom de fichier.

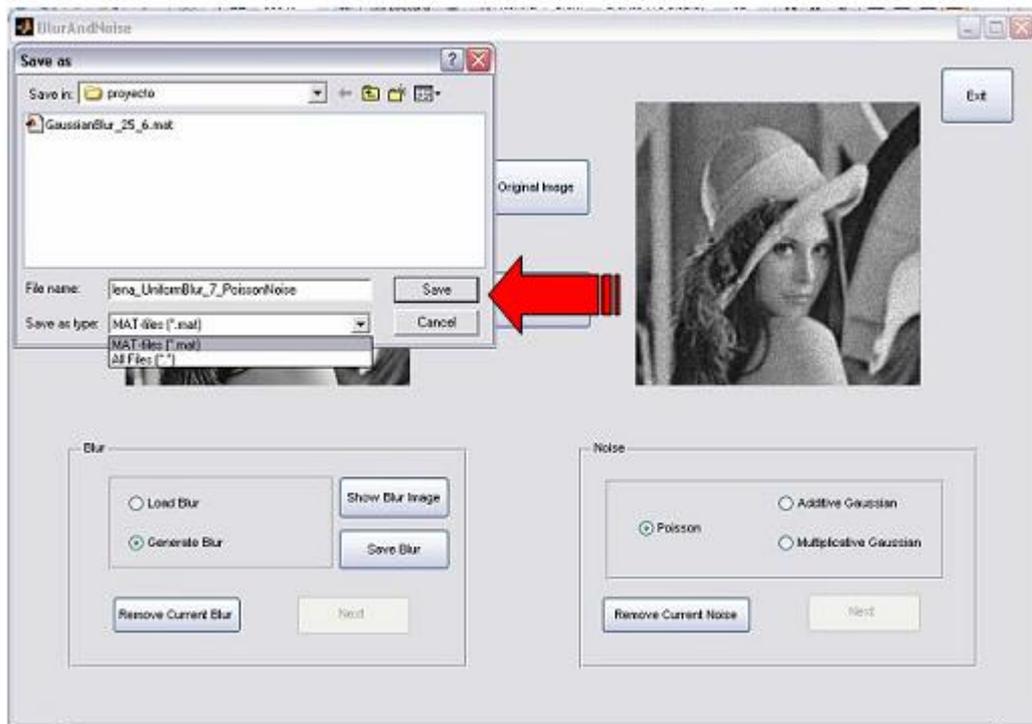


Figure V.21. Navigateur de précisé l'endroit, le nom, et le format d'image dégradée.

## 2.8. Fermeture de l'application

Pour fermer l'application et toutes ses fenêtres, il suffit d'appuyer sur le bouton "Exit", comme il est montré dans l'image:

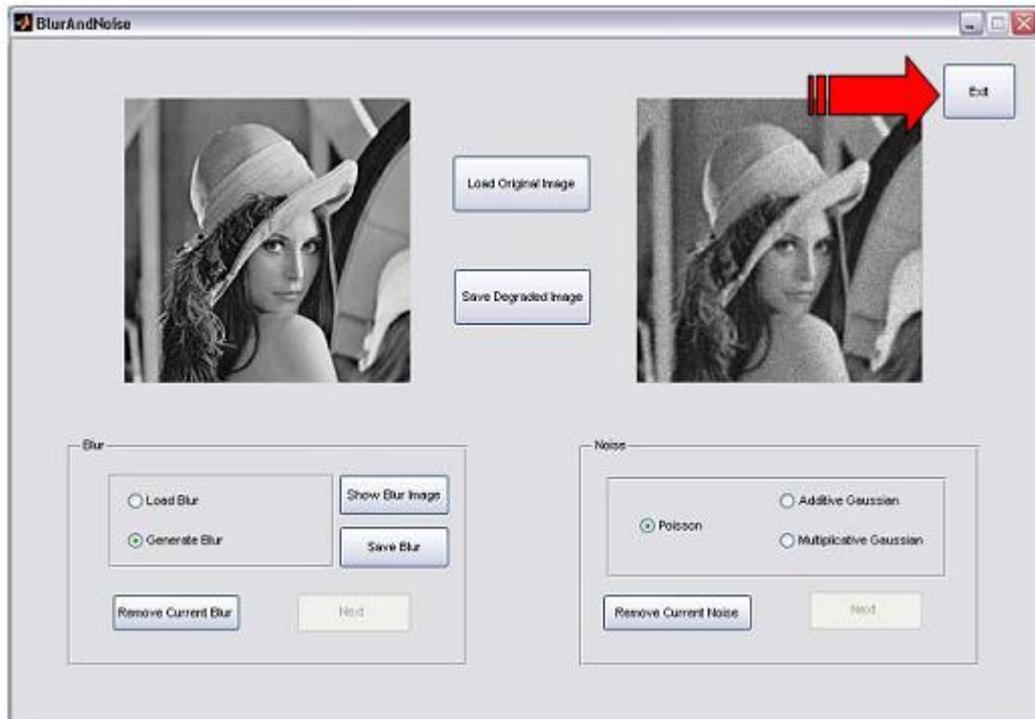


Figure V.22. Ferme l'application « BurAndNoise.fig ».

### 3. Résultat de restauration d'image publié dans [127]

Dans ce paragraphe, nous avons classé les résultats en deux grands parties : la restauration des images de synthèse et la restauration des images réelles.

#### 3.1. Restauration des images de synthèse

Pour les dégradations d'images de synthèse, l'image présentée sur la Fig. V.23(a) est utilisée pour créer l'image observée  $y_1$  et  $y_2$ .

L'image observée  $y_2$ , Fig. V.23(b), est obtenue en ajoutant un bruit blanc gaussien de variance 220 à l'image d'origine (SNR = 7 dB). Cette image souffre d'un très haut niveau de bruit: l'erreur quadratique moyenne (MSE) entre cette image et l'image originale est 196,20. On crée cinq observations  $y_1$  différentes en brouillant l'image d'origine par cinq PSF de flou différents représentés sur la ligne inférieure de la Fig.V.24, qui sont des exemples typiques



(a)



(b)

*Figure V.23. (a) Image originale Ephèse, (b) Image bruitée simuler d'une acquisition à court d'exposition.*

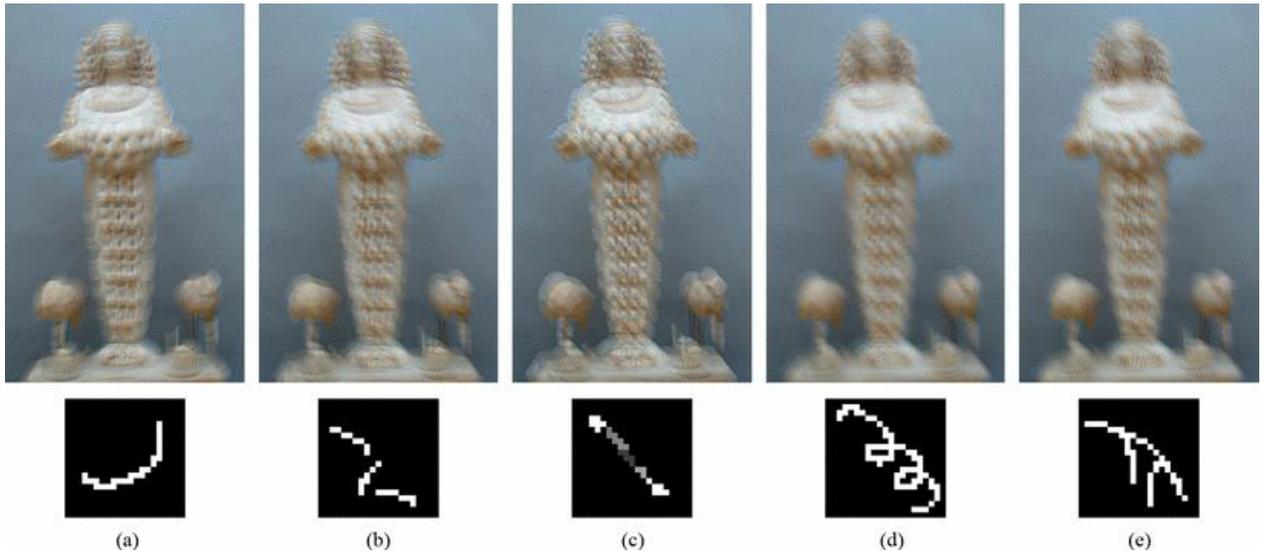


Figure V.24. Images floues simulées à longue exposition des caméras. Le PSF utilisé pour générer chaque image est affiché sous l'image correspondante. Tous les PSFs ont la même dimension  $21 \times 21$  et les images sont de taille  $430 \times 270$  pixels. Les valeurs des PSF sont linéairement associées à la plage  $[0:255]$  des fins de visualisation.

des PSF résultant du mouvement de la caméra pendant de longues expositions. Un bruit blanc gaussien avec une variance de 0.16 est ajouté aux images floues pour obtenir ces dernières images observées  $y_1$ , ce qui est indiqué sur la ligne supérieure de la Fig. V.24.

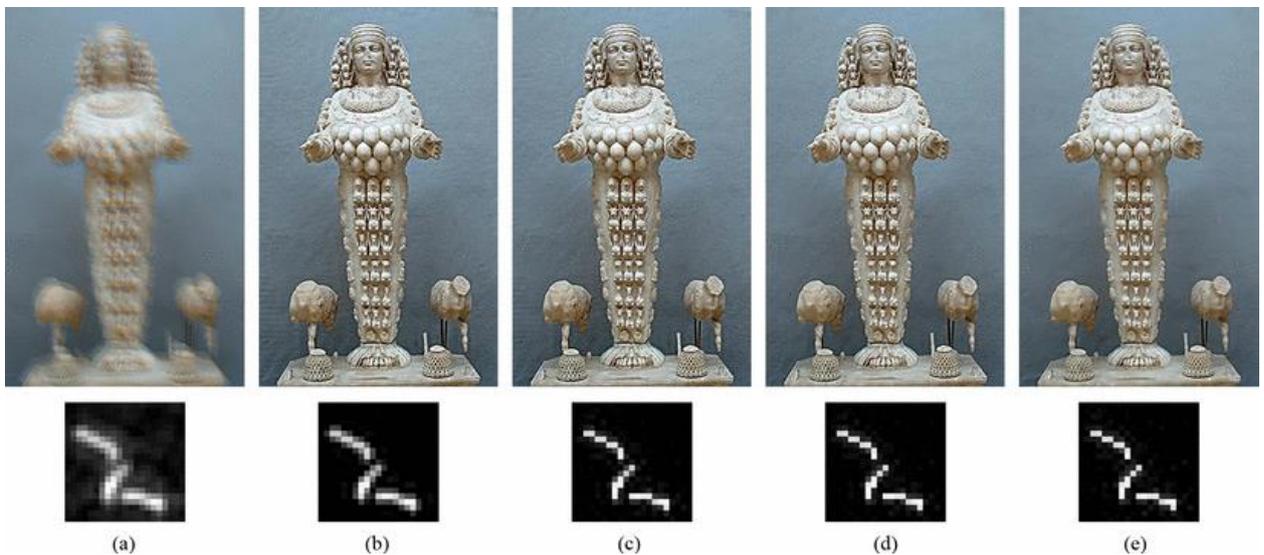


Figure V.25. Résultats de la restauration en utilisant l'algorithme proposé. Les images restaurées sont présentées dans la rangée du haut, et les PSF correspondantes récupérées sont présentés ci-dessous les images. Les valeurs des PSF sont linéairement associées à la plage  $[0:255]$  des fins de visualisation.

Les MSEs entre l'image originale et les images observées sont 485.20, 451.50, 508.52, 99 | Page

412.44 et 417.51, respectivement. Les supports des PSF utilisées dans cette expérience sont  $21 \times 21$ , et le taille de l'image originale est  $430 \times 270$ .

Les images restaurées obtenus par la méthode proposée correspondant à différents PSF sont présentés sur la ligne supérieure de la Fig. V.25, et le PSF récupéré pour chaque cas est représenté sur la ligne inférieure de la Fig. V.25. Par comparaison avec l'image originale sur la Fig. V.25(a), il est clair que l'algorithme proposé fournit des images restaurées avec une qualité visuelle très élevé dans tous les cas. La MSE entre l'image originale et les images restaurées sont 13.32, 24.9, 42.76, 17.68 et 23.14, respectivement, ce qui montre que les images restaurées sont très proches de l'image originale. Aussi, les MSEs entre les PSF d'origine et estimés sont  $9 \times 10^{-7}$ ,  $8 \times 10^{-6}$ ,  $5 \times 10^{-6}$ ,  $9 \times 10^{-7}$  et  $3 \times 10^{-6}$ . Les résultats quantitatifs MSE et l'inspection visuelle des PSF récupérés suggèrent que l'algorithme est très efficace pour estimer les PSF d'origine. Et qui est en fait homogène; c'est-à-dire le flou est le même sur toute l'image et par conséquence, réaliste.

### 3.2. Restauration des images réelles

Dans cette partie, nous appliquons la méthode proposée à une paire d'images réelles acquises par un appareil photo numérique compact. Les Fig. V.26(a)-(b) montrent une paire d'images acquises en plein air à ISO = 800 avec des temps d'exposition 1/100 et 1/3 secondes, respectivement.

En raison des conditions de faible luminosité le niveau de bruit dans l'image avec une exposition courte est assez élevé. En outre, il peut être observé que dans la Fig. V.26(a) le bruit dans l'image à courte exposition est coloré et donc ne suit pas le modèle d'observation supposé dans (4.2). En outre, certaines parties des images sont fortement saturés (par exemple, une partie de la fenêtre au-dessus de la porte), ce qui introduit une difficulté supplémentaire dans l'estimation de flou en raison de sa non-linéarité.

L'application de l'algorithme de débruitage dans [114] pour l'image à court exposition est représentée dans les Fig. V.26(c). Nous avons réglé manuellement les paramètres de l'algorithme de débruitage pour chaque canal de couleur; et le résultat est montre de haute qualité visuelle. Les estimations d'image et le flou fournis par l'algorithme proposé sont présentés dans les Fig. V.26(d)-(f), respectivement. Le PSF estimé a pour dimension  $51 \times 51$ .



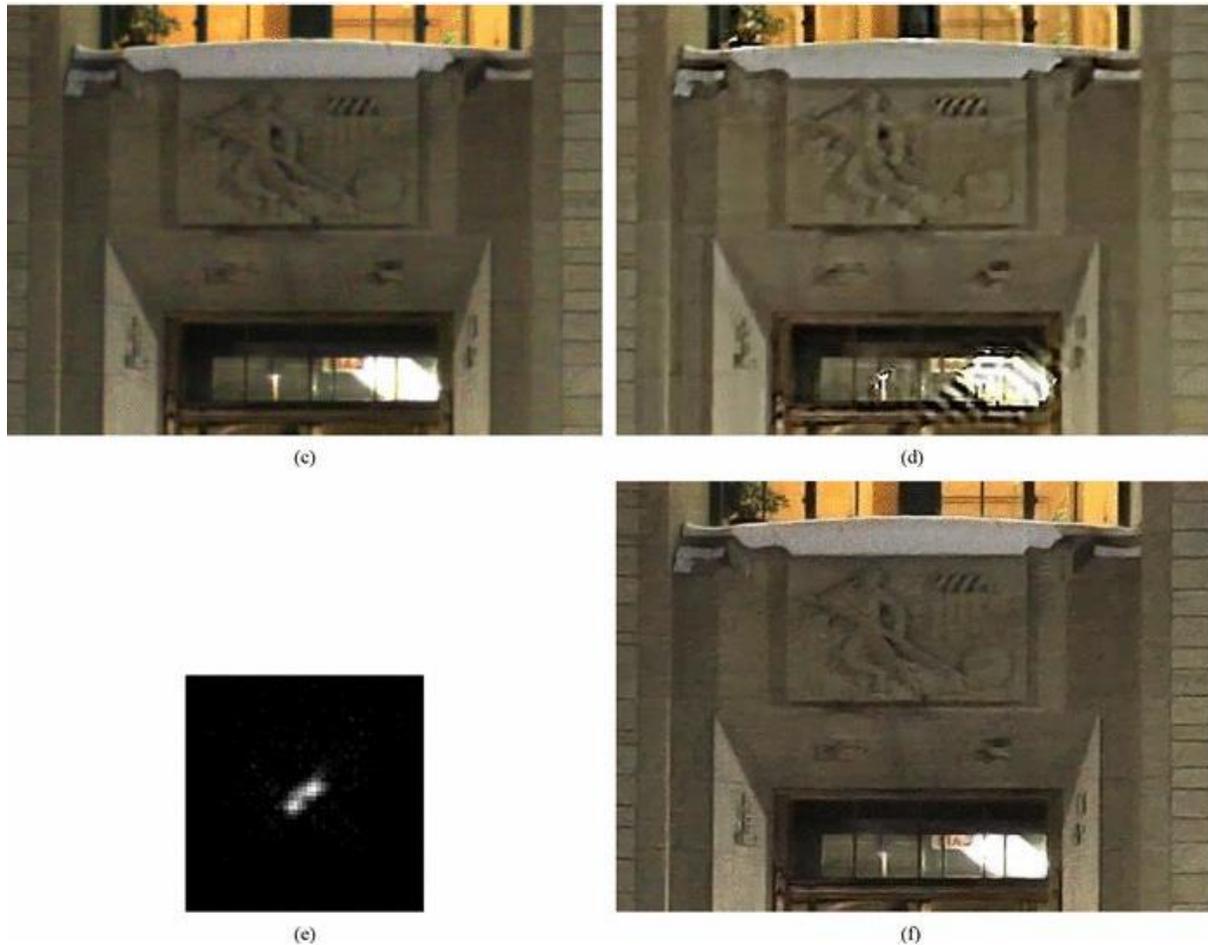
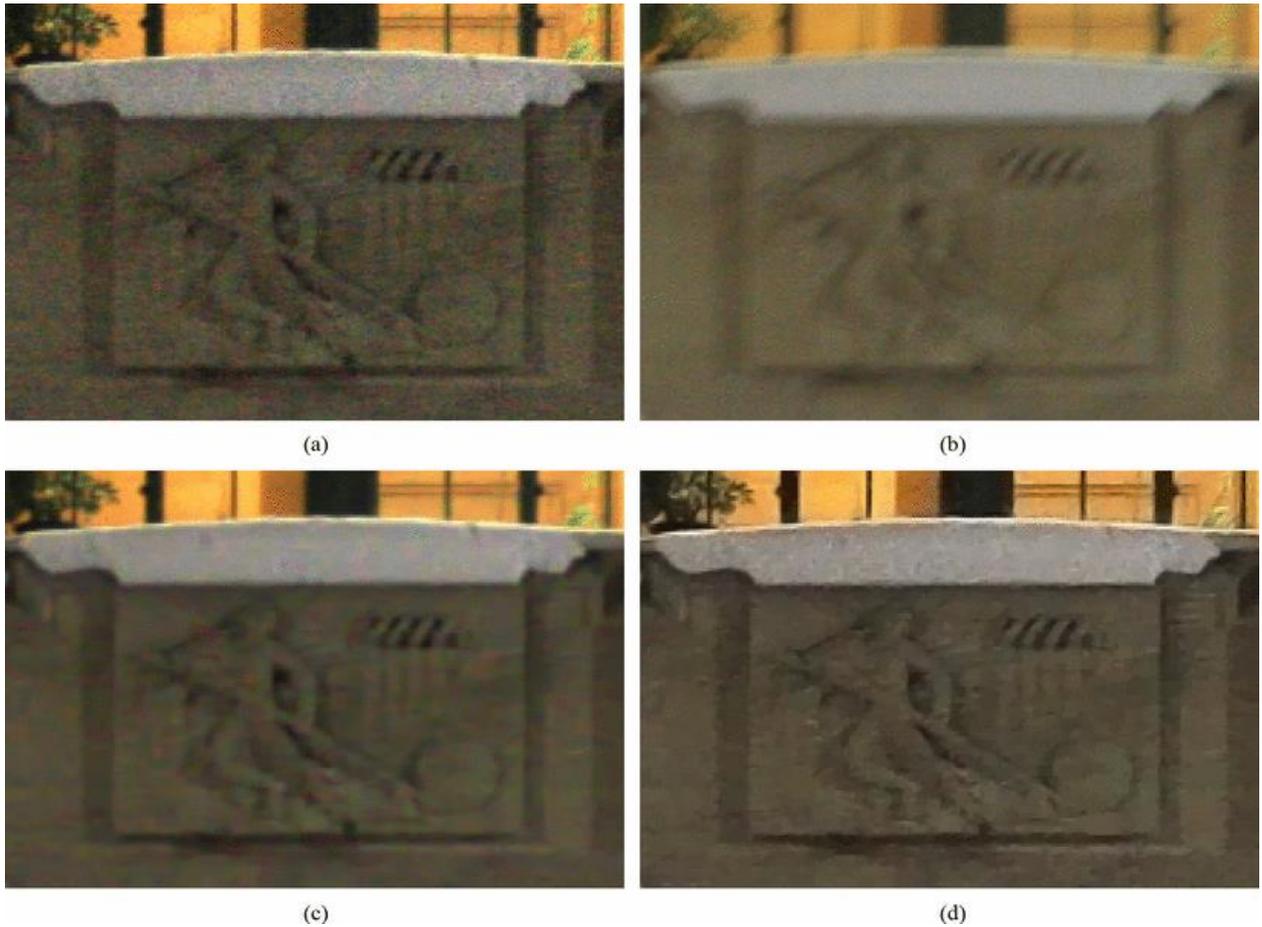


Figure V.26. Une paire d'images en plein air. (a) Image à courte exposition (niveau de luminosité corrigée), (b) image à longue exposition (c) image débruitée de courte durée exposition, (d) image restaurée en utilisant l'algorithme proposé, et (e) PSF estimé (support: 51x51).

Les régions centrales des images sont affichées dans leur taille d'origine dans la Fig. V.27 pour un examen plus approfondi. On peut observer que, malgré le caractère difficile des images d'entrée, l'algorithme fournit une amélioration significative tant en supprimant le flou et révélant des détails précis de l'image, ainsi que, dans la correction de la perte de la couleur apparente dans l'image de courte exposition. Bien que la méthode de débruitage a aussi réussi à éliminer le bruit d'acquisition, elle ne peut pas corriger la perte de couleur dans l'image à courte exposition, et son résultat est plus doux que celui fourni par la méthode proposée.



*Figure V.27. Régions centrales des images montrées à la Fig. V.26. (a) Image à courte durée exposition, (b) image avec une longue durée d'exposition (c) image débruitée à courte durée exposition, (d) le résultat de l'algorithme proposé.*

Enfin, nous présentons une comparaison de la méthode proposée avec la méthode de stabilisation d'image proposée dans [119] et l'algorithme de débruitage sur un ensemble d'images réelles.

Les images montrées à la Fig. V.28(a) et Fig. V.28(b) (publié dans [119]) sont prises avec des temps d'exposition 1/100 s et 1 s, respectivement. Le résultat de l'application de l'algorithme de débruitage pour l'image à courte durée d'exposition est montré sur la Fig. V.28 (c).

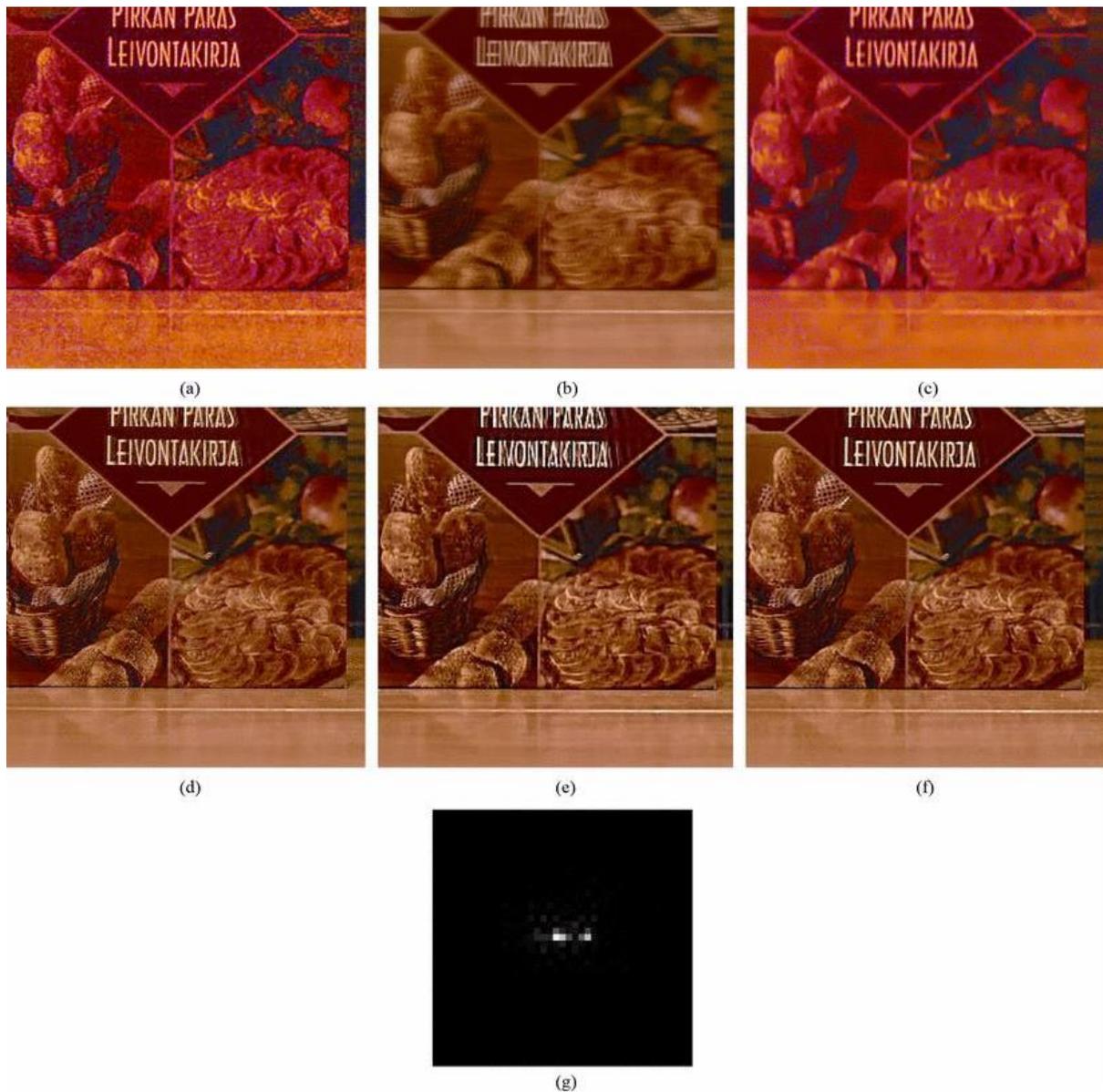


Figure V.28. Par exemple, à image réelle (avec la permission de [119]). (a) Image à courte durée exposition, (b) image avec une longue durée exposition (c) image débruitée à courte durée exposition, (d) image restaurée en utilisant [224-Molina], (e) PSF récupéré en utilisant l'algorithme proposé (Dimension: 41x41), et (f) Image restaurée en utilisant l'algorithme proposé.

Il est clair que, même si le niveau de bruit est réduit de façon significative, le contraste est très faible, et il y a une dominante de couleur rouge significative. L'image restaurée dans la Fig. V.28(d) est obtenue par l'algorithme de [119], ce qui nécessite la connaissance des variances du bruit dans les observations. Enfin, les estimations PSF et l'image fournis par l'algorithme proposé sont présentés sur la Fig. V.28(d) et (e), respectivement, lorsque la taille d'estimation de la PSF est 41x41.

Notons que bien que la méthode proposée est entièrement automatisé, l'image restituée est clairement plus forte que celle de [119] presque sans éléments redondants. Cela est particulièrement évident dans la zone autour des lettres. En outre, l'image restituée par la méthode proposée est plus nette que l'image débruitée avec un contraste plus élevée des couleurs correctement restaurés.

En résumé, les résultats expérimentaux sur les ensembles d'images synthétiques et réelles montrent que l'algorithme proposé est très efficace pour fournir des images de haute qualité restaurées, et avec sans aucune réglage des paramètres utilisent.

#### **4. Nos restaurations d'image: Logiciel et Résultats**

D'une façon similaire au paragraphe précédent, nous sommes présentons le logiciel et les résultats de nos simulation. Notons que le model publie dans [114] est à code privé.

Dans le cadre de donner à l'utilisateur une notion et manipulation complète, on a développé un logiciel « BayesianBlindDéconvolution.fig » qui est en mesure d'effectuer la restauration des images brutes et flouées.

Tout d'abord, l'application de déconvolution aveugle sera expliquée avec une série de programmes instantanés, et tous les menus et leurs fonctionnalités seront affichées. La dernière partie contient un exemple de l'usage pour aider l'utilisateur.

En résumé, ce logiciel restore deux images données initialement, on applique l'algorithme1 (page: 76) ou l'algorithme2 (page: 77) de celui-ci.

## 4.1. Fenêtre principale

Ici, nous pouvons voir le programme de base quand il est lancé, et une explication brève est donnée des boutons et des menus principaux:

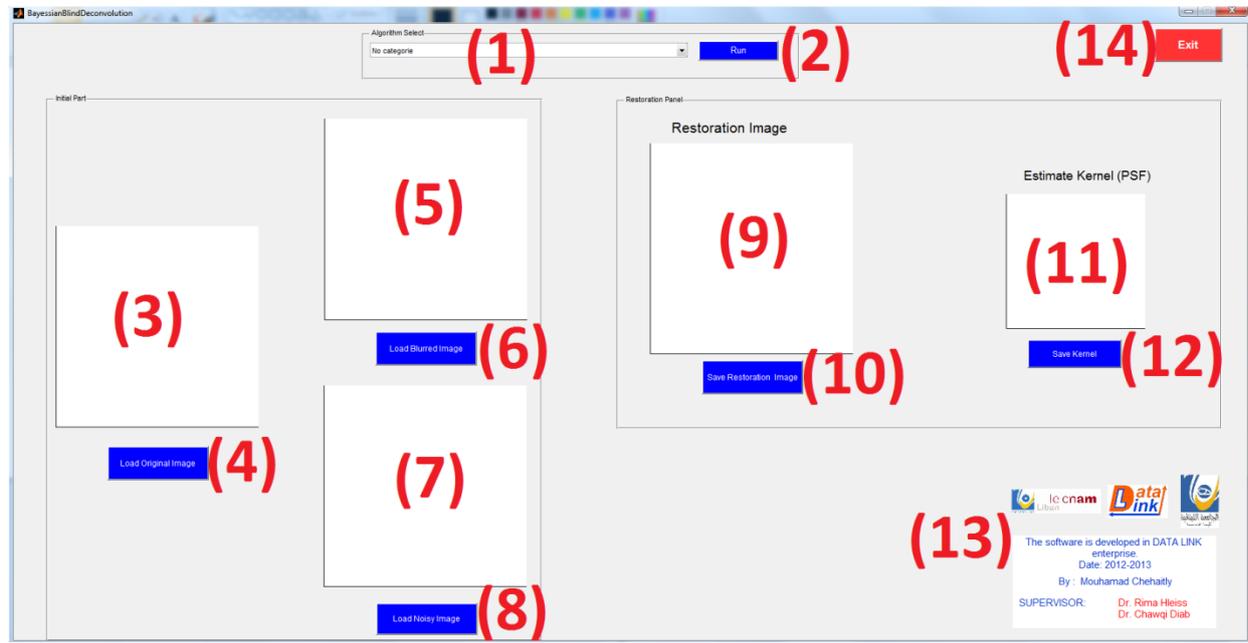


Figure V.29. Fenêtre principale de fonction Matlab " BayessianBlindDéconvolution.m

(1) Le bouton de sélection de l'algorithme à tourner. En fait on a trois choix :

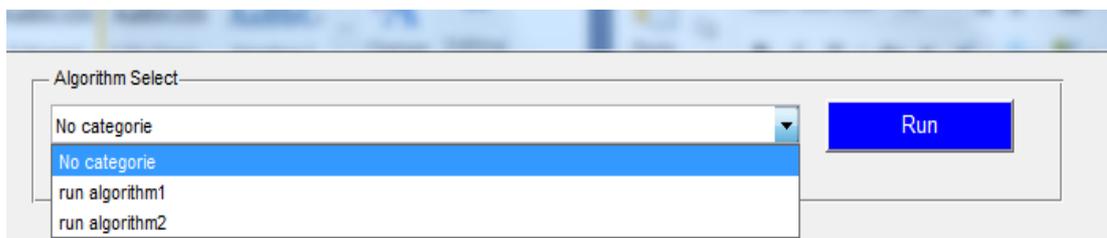


Figure V.30. Fenêtre principale pour sélectionner l'algorithme considéra.

1- « No categorie »: si l'utilisateur autre que l'algorithme1 ou l'algorithme2 et donc un message paraît pour alarmer l'utilisateur que le logiciel se ferme en 5 secondes.

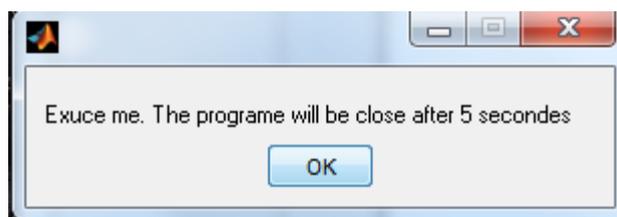


Figure V.31. Fenêtre alarme si l'utilisateur choisie « No categorie ».

2- « run algorithm1 »: l'utilisateur choisie l'agolrithme1 définir page : 76

3- « run algorithm2 »: l'utilisateur choisie l'agolrithme2 définir page : 77

(2) Bouton pour exécuter l'algorithme sélectionné.

(3) Cet espace affiche l'image chargée par pression sur Bouton 4.

(4) Le bouton Charger l'image originale nous permet de charger une image de l'intérieur du système.

(5) Cet espace affiche l'image chargée par pression sur Bouton 6.

(6) Le bouton Charger l'image flouée nous permet de charger une image floue de l'intérieur du système.

(7) Cet espace affiche l'image déjà chargée par pression sur Bouton 8.

(8) Le bouton Charger l'image bruitée nous permet de charger une image bruitée de l'intérieur du système.

(9) Cet espace affiche l'image restaurée obtenue après l'exécution de l'algorithme 1 ou l'algorithme 2.

(10) Le bouton Enregistrer l'image restaurée, permet d'enregistrer l'image observée (9) dans le système.

(11) Cet espace affiche le PSF (noyau de flou) estimé obtenu après l'exécution de l'algorithme 1 ou l'algorithme 2.

(12) Le bouton Enregistrer le masque restauré, permet d'enregistrer l'image observée (11) dans le système.

(13) Information sur moi, créateur de ce programme.

(14) Le bouton Quitter ferme l'application.

## 4.2. Charger une image

La première chose que nous devons commencer à travailler afin d'obtenir une image observée est de charger une image, si on clique sur un des boutons de charger une image comme il est montré dans l'image:

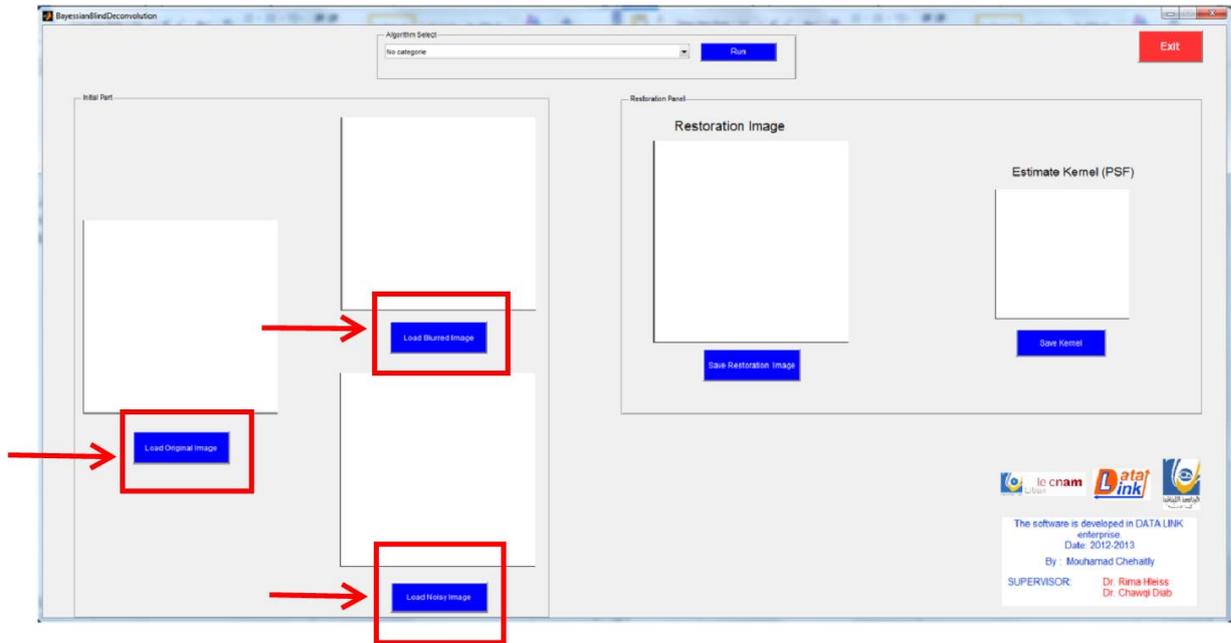


Figure V.32. Chargement de l'image originale, image floue et image bruitée.

Ensuite, nous aurons besoin de sélectionner et d'ouvrir une image en utilisant le navigateur qui apparaît. Nous sommes maintenant en système de la recherche d'une image à un format spécifique pour traiter. Lorsque l'image est trouvée, la prochaine étape est de cliquer sur le bouton "Ouvrir" dans le navigateur, comme il est montré dans l'image:

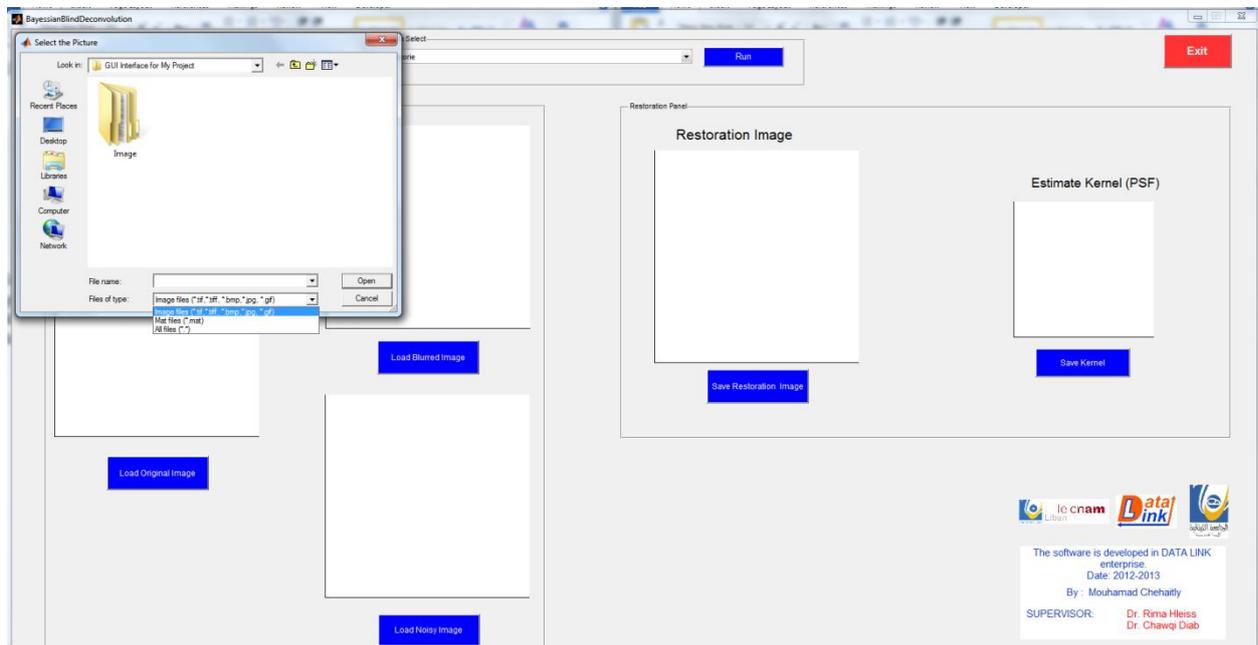


Figure V.33. Navigateur de sélection l'image.

Lorsqu'elle est sélectionnée, l'image sera affichée dans le carré de gauche (carré numéro (3) dans la figure V.29) de l'interface, qui ressemble à ceci:

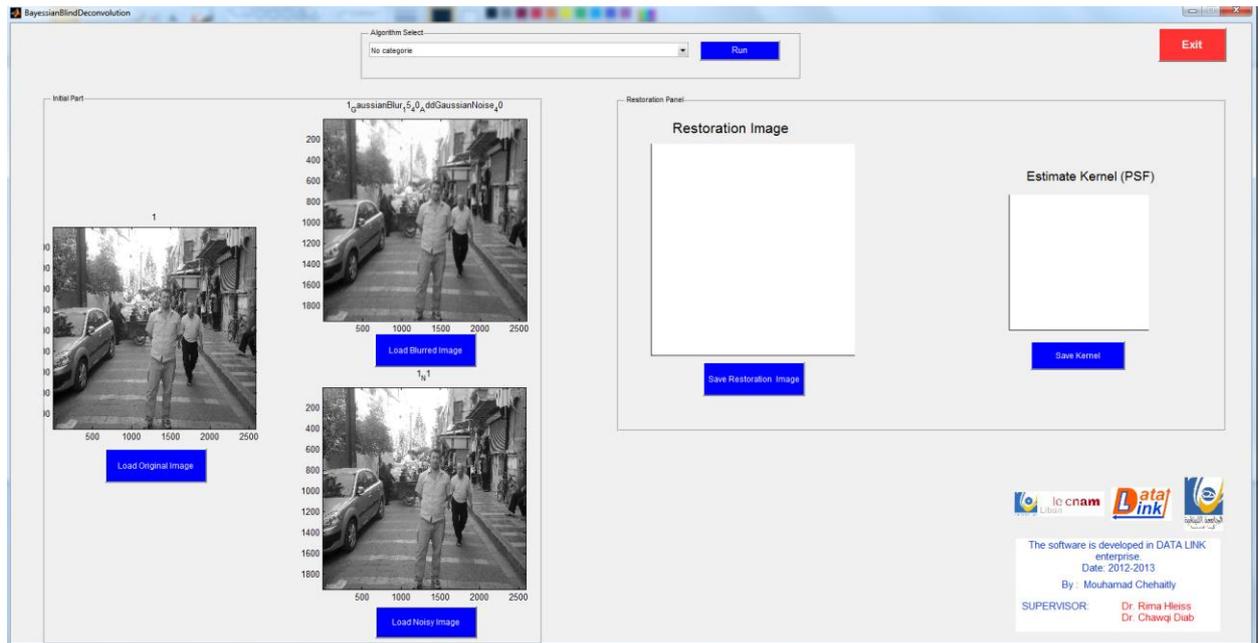


Figure V.34. Affiche l'image sélection dans la fenêtre principale.

Notez que le bouton "Exécuté ou Run" est maintenant disponible dans le panneau de sélection d'algorithme, nous sommes maintenant en mesure de commencer à travailler sur ces images.

Mais avant le mettre en marche l'algorithme choisi, il faut préciser la taille de masque flou (PSF) M

### 4.3. Précision la taille de PSF

Nous allons maintenant utiliser le panneau de dimension de PSF. Le panneau nous donne deux options:

#### 4.3.1. Charger la taille de PSF

Cette option permet d'écrire la dimension de PSF stocké dans l'interface « size ».

Quand, nous devons cliquer sur le bouton « Run » alors que le panneau de dimension de noyau flou est visible, comme il est montré dans l'image suivante.

(On note que: Le centre du flou est le centre du masque.)

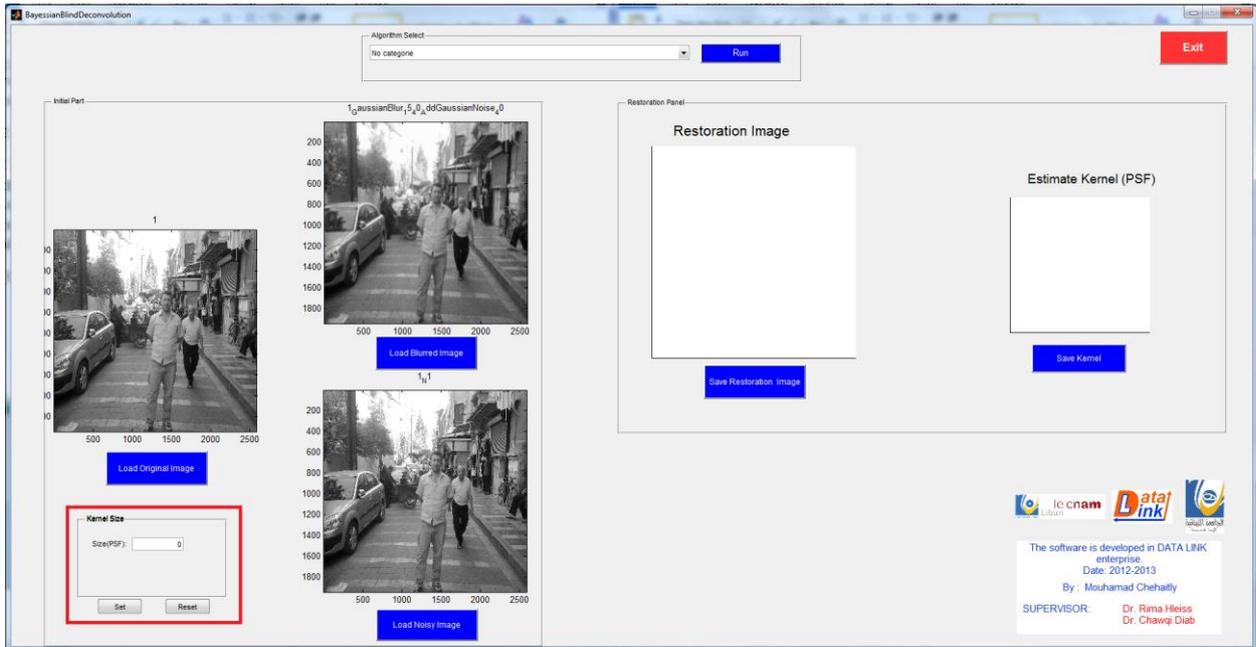


Figure V.35. Charge la taille de flou.

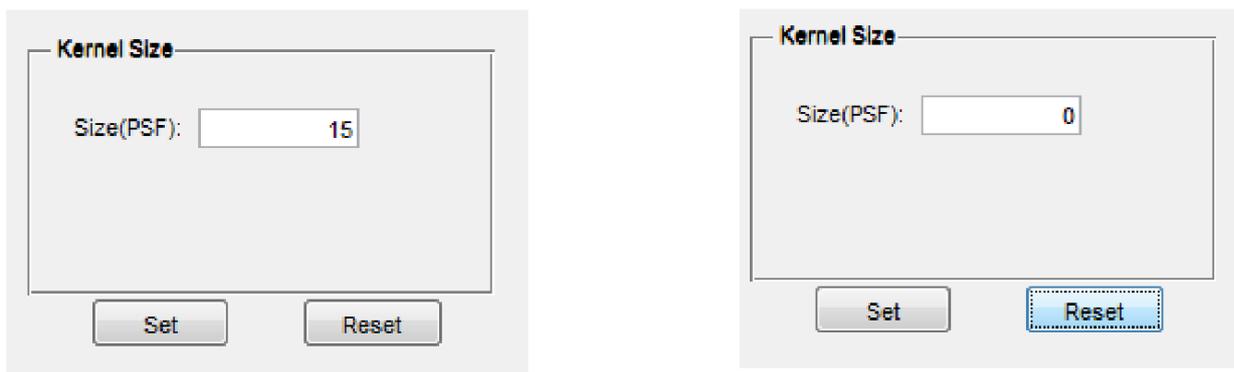
Lorsque l'utilisateur clique sur le bouton « Set » la dimension de noyau flou doit être stockée et le programme est mis en marche.

Mais avant de cliquer sur bouton « Set », l'utilisateur possède le choix de changer la taille de noyau comme indiqué au paragraphe suivant.

#### 4.3.2. Recharger la taille de PSF

Cette option permet de réécrire la dimension de PSF.

Quand, nous devons cliquer sur le bouton « Reset » alors que l'interface « Size » est réinitialisé à zéro, comme il est montré dans l'image suivante.



After push button Reset

Figure V.36. Recharge la taille de flou

## 4.4. Mettre en marche le programme

Si nous choisissons d'exécuter l'algorithme choisi, l'image restaurée et le noyau estime seront affichés dans les carrés blanc à droite (carrés numéros (9) et (11) de la figure V.29), montrant que le programme a été appliqué avec succès. Comme montre la figure suivante :

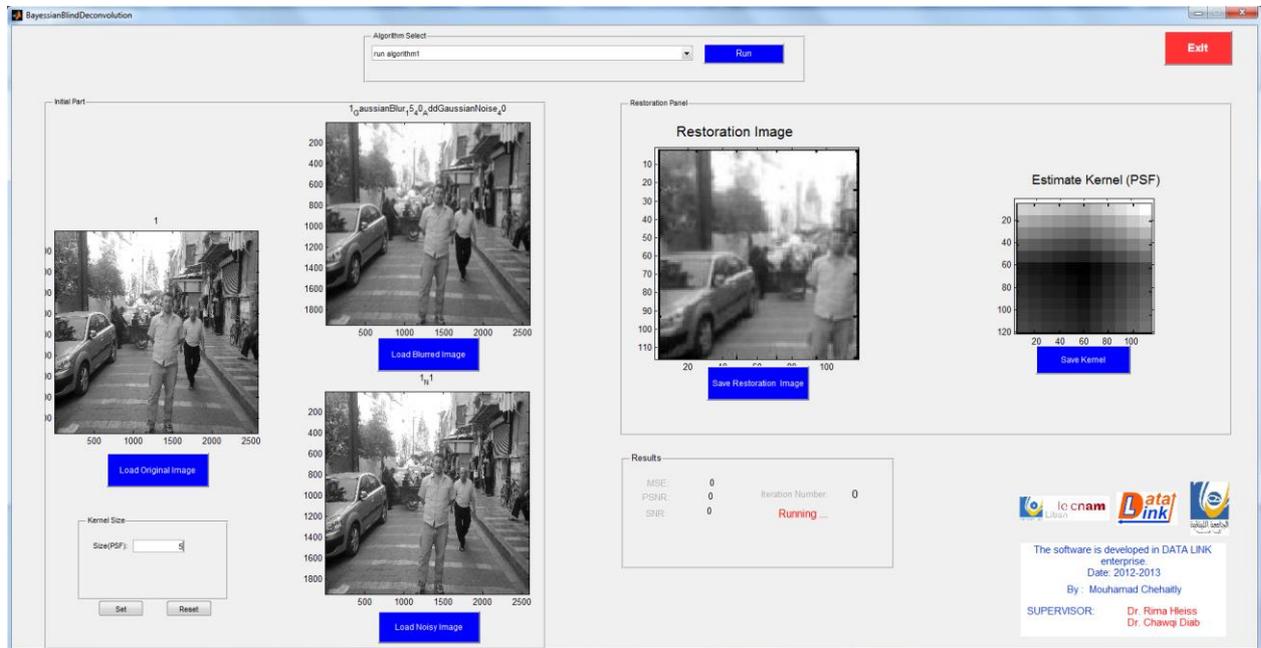


Figure V.37. Situation après l'exécution du programme.

On note qu'en fin d'itération le panneau de « Résultats » apparaît avec quatre titres importants :

- 1- MSE: l'erreur quadratique moyenne. Indice très important pour juger la qualité d'image restaurée.
- 2- SNR: Rapport signal sur bruit.
- 3- PSNR : en décibel avec  $PSNR = -10\log(1/MSE)$ .

## 4.5. Fermeture de l'application

Pour fermer l'application et toutes ses fenêtres, il suffit d'appuyer sur le bouton "Exit", comme il est montré le Fig. V.37:

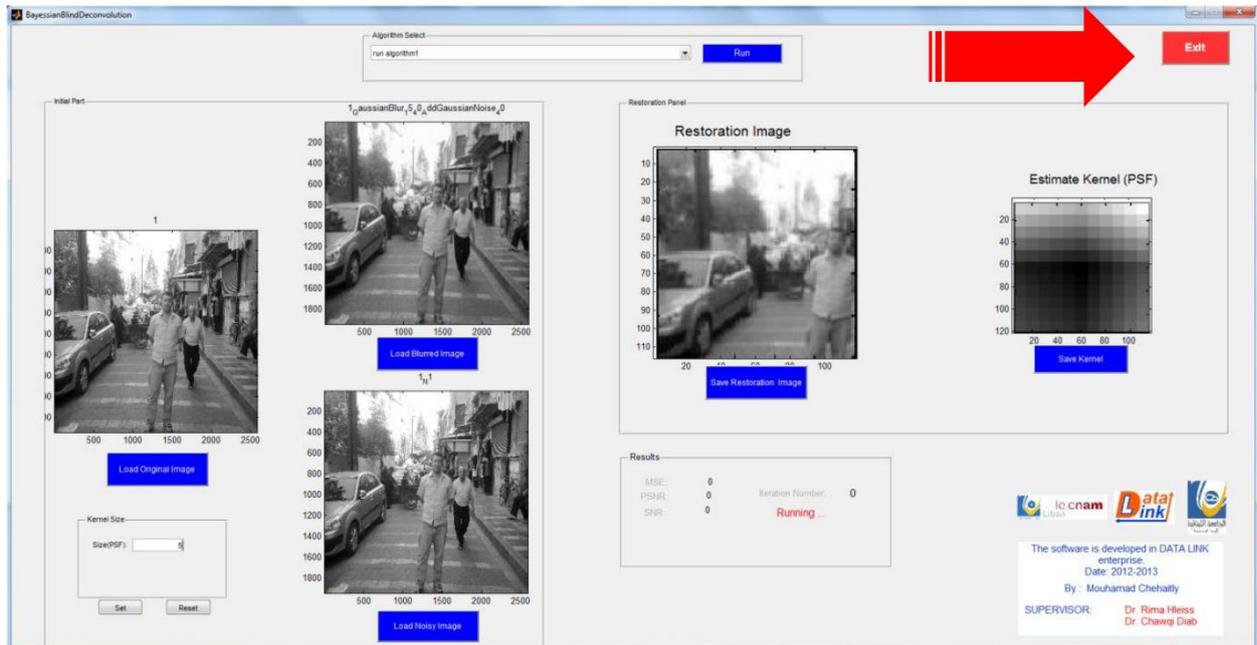


Figure V.37. Ferme l'application « BayesianBlindDéconvolution.fig ».

## 5. Restauration des images de synthèse en utilisant l'Algorithme1

Pour les dégradations d'images de synthèse, nous avons travaillé sur 8 modèles d'image différentes capturés sous différentes ISO. En plus, les images et les PSFs sont estimés à partir l'exécution de l'Algorithme1. C'est-à-dire avec la forme isotropie de TV.

### 5.1. Restauration des images de synthèse : Modele1

Modele1 représente l'image de l'artiste: Mademoiselle Scarlet Jonson prise avec ISO égale 875 dans une chambre.

L'image de Scarlet est utilisée pour créer l'image observée  $y_1$  et  $y_2$  de ce modèle proposé aux équations (4.1 et 4.2). L'image observée  $y_2$ , Fig. V.38 (b), est obtenue en ajoutant un bruit blanc gaussien de variance 35 à l'image d'origine (SNR = 27.1412). Cette image souffre d'un haut niveau de bruit: la MSE entre cette image et l'image originale est 18.5676.

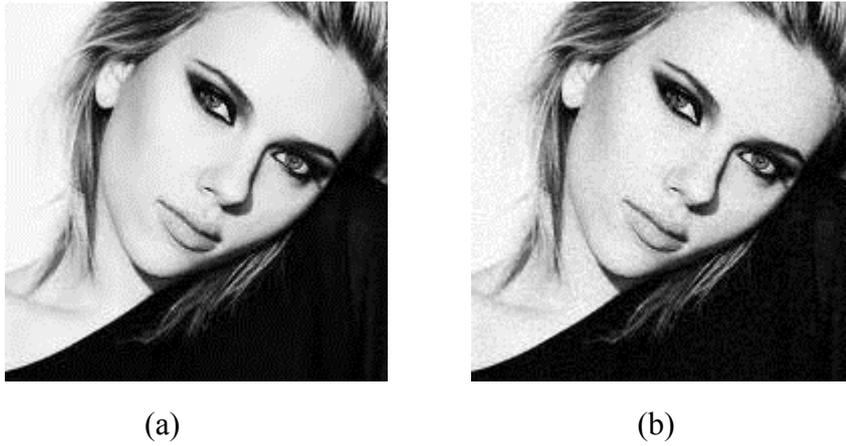


Figure V.38. (a) Image originale de Scarlet, (b) Image bruitée simuler d'une acquisition à court d'exposition.

Nous avons généré dix observations  $y_1$  différentes; en brouillant l'image d'origine par dix PSFs différentes représentées sur la ligne inférieure de la Fig. V.39, qui sont des exemples typiques des PSF résultant du mouvement de la caméra pendant de longues expositions. Où les valeurs du masque de flou (PSF) sont de standard publiés dans [130].

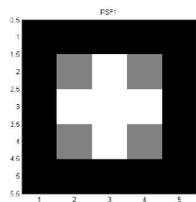
Les différentes PSFs sont:

```

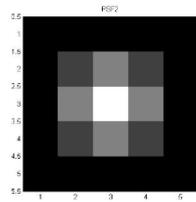
PSF1 = (1/14) * [0 0 0 0 0 ; 0 1 2 1 0 ; 0 2 2 2 0 ; 0 1 2 1 0 ; 0 0 0 0 0];
PSF2 = (1/16) * [0 0 0 0 0 ; 0 1 2 1 0 ; 0 2 4 2 0 ; 0 1 2 1 0 ; 0 0 0 0 0];
PSF3 = (1/18) * [0 0 0 0 0 ; 0 1 2 1 0 ; 1 2 2 2 1 ; 0 1 2 1 0 ; 0 0 0 0 0];
PSF4 = (1/18) * [0 0 0 0 0 ; 0 2 2 2 0 ; 0 2 2 2 0 ; 0 2 2 2 0 ; 0 0 0 0 0];
PSF5 = (1/19) * [0 0 1 0 0 ; 0 1 2 1 0 ; 1 2 3 2 1 ; 0 1 2 1 0 ; 0 0 1 0 0];
PSF6 = (1/25) * [1 1 1 1 1 ; 1 1 1 1 1 ; 1 1 1 1 1 ; 1 1 1 1 1 ; 1 1 1 1 1];
PSF7 = (1/26) * [0 1 1 1 0 ; 1 1 2 1 1 ; 1 2 2 2 1 ; 1 1 2 1 1 ; 0 1 1 1 0];
PSF8 = (1/28) * [0 1 1 1 0 ; 1 1 2 1 1 ; 1 2 4 2 1 ; 1 1 2 1 1 ; 0 1 1 1 0];
PSF9 = fspecial('gaussian',5,15);
PSF10 = fspecial('motion',5,45);

```

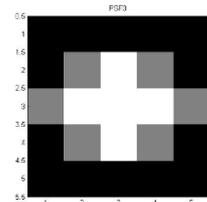
Maintenant, nous présentons l'effet de chaque noyau de flou sur l'image originale.



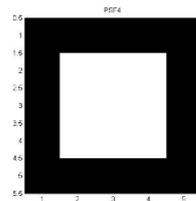
(a)



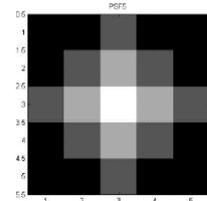
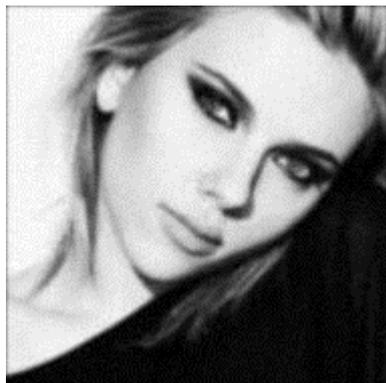
(b)



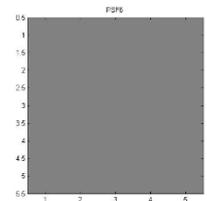
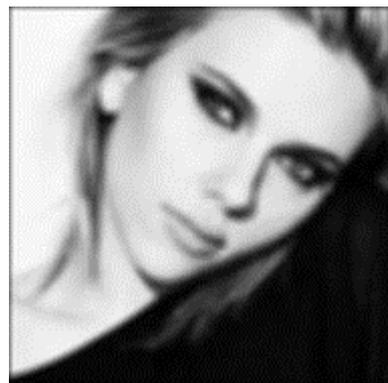
(c)



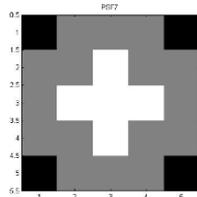
(d)



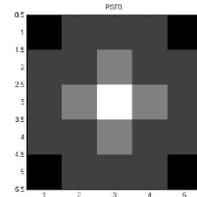
(e)



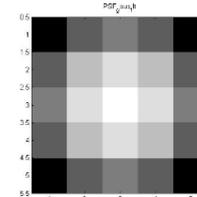
(f)



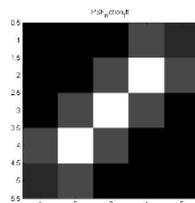
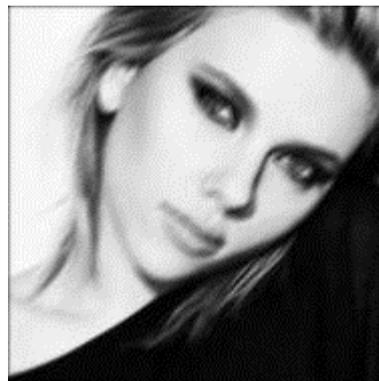
(g)



(h)



(i)



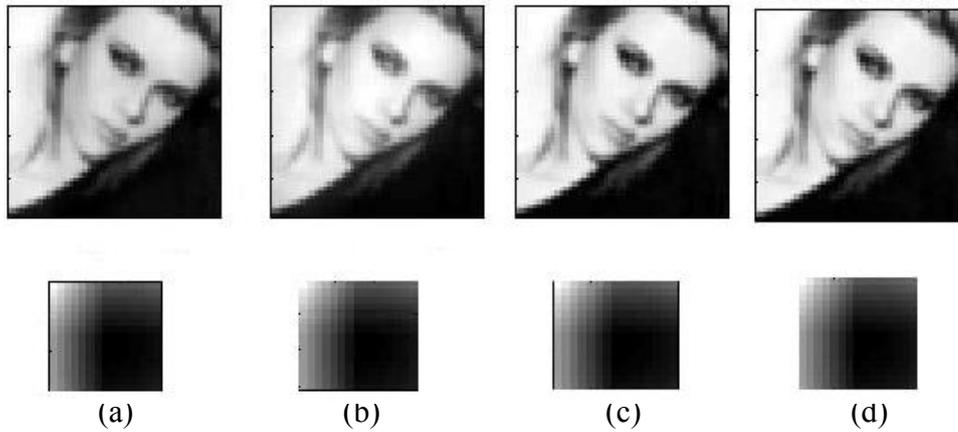
(j)

Figure V.39. Images floues prises similaire à longue exposition des caméras. Le masque de flou (PSF) utilisé pour générer chaque image est affiché sous l'image correspondante. Tous les PSF's ont la même dimension 5X5 et les images sont de taille 90X90 pixels. Les valeurs des PSF sont linéairement associées à la plage [0:255] des fins de visualisation.

Puis sur chaque image de Fig. V.39 un bruit gaussien est ajouté. Plus spécifiquement, un bruit blanc gaussien avec une variance de 35 est ajouté aux images floues pour obtenir ces

dernières images observées  $y_1$  avec  $\text{SNR} = 40$  dB.

Les images restaurées obtenues par la méthode proposée en algorithme1 (page: 76) correspondant à différents PSF sont présentés sur la ligne supérieure de la Fig. V.40, et le PSF récupéré pour chaque cas est représenté sur la ligne inférieure de la Fig. V.40.



*Figure V.40. Résultats de la restauration en utilisant l'algorithme proposé. Les images restaurées sont présentées dans la rangée du haut, et les PSFs correspondantes récupérées sont présentés ci-dessous les images. Les valeurs des PSF sont linéairement associées à la plage  $[0:255]$  des fins de visualisation.*

La Fig. V.40 représente l'image et le noyau restaurés après chaque itération, et comme nous avons vu il y a de changement important de qualité et de vision d'image. Donc il est important de voir la progression d'erreur après chaque itération.

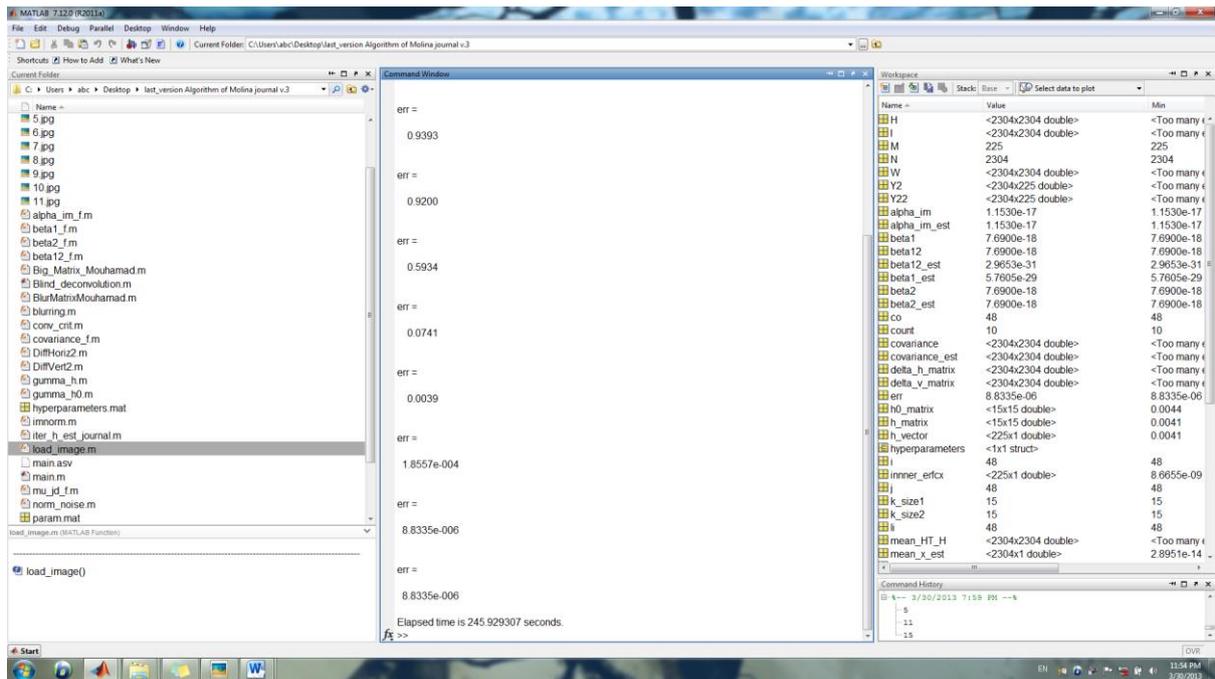


Figure V.40.a. Cette figure représente la fenêtre Matlab principale. L'erreur finale est égale  $8.8335 \times 10^{-6}$  entre l'image originale et l'image restaurée. Le temps consommé est 246 secondes. La taille de PSF est  $15 \times 15$ . Dimension de l'image estimée  $45 \times 45$ .

Par comparaison avec l'image originale sur la Fig. V.38 (a), il est clair que l'algorithme proposé fournit des images restaurées avec une qualité visuelle élevée.

La moyenne d'erreur quadratique (MSE) entre l'image originale et les images restaurées sont 13,32, 24,9, 42,76, 17,68 et 23.14, respectivement, ce qui montre que les images restaurées sont très proches de l'image originale. Aussi, les MSEs entre les PSF d'origine et estimées sont  $9 \times 10^{-7}$ ,  $8 \times 10^{-6}$ ,  $5 \times 10^{-6}$ ,  $9 \times 10^{-7}$  et  $3 \times 10^{-6}$ . Les résultats quantitatifs MSE et l'inspection visuelle des PSF récupérés suggèrent que l'algorithme est très efficace pour estimer les PSF d'origine.

## 5.2. Restauration des images de synthèse : Modele2

Modele2 représente mon image dans la vieille marche de Beyrouth: Cette image prise en plein d'air sous condition d'éclairage ISO égale 950.

L'image de la vieille marche est utilisée pour créer l'image observée  $y_1$  et  $y_2$  du modèle proposé en équations (4.1 et 4.2). L'image observée  $y_2$ , Fig. V.41 (b), est obtenue en ajoutant un bruit blanc gaussien de variance 45 à l'image d'origine (SNR = 10.2446 et PSNR = -13.4474 dB). Cette image souffre d'un haut niveau de bruit: la MSE entre cette image et l'image originale est 22.6849.



(a)

(b)

*Figure V.41. (a) Image originale de la veille marche, (b) Image bruitée simuler d'une acquisition à court d'exposition.*

Nous avons généré l'observation  $y_1$  en brouillant l'image d'origine par un noyau flou Gaussien de dimension  $15 \times 15$  et par variance égale 45 représenté dans la Fig. V.42 (a), qui est en principe similaire au mouvement de la caméra pendant de longues expositions.

Puis sur l'image floue de Fig. V.42(a) un bruit gaussien est ajouté. Plus spécifiquement, un bruit blanc gaussien avec une variance de 40 est ajouté aux images floues pour obtenir ces dernières images observées  $y_1$  Fig. V.42 (b) avec SNR égale 5.2922, PSNR= 40 dB et avec MSE=41.1987.



(a)

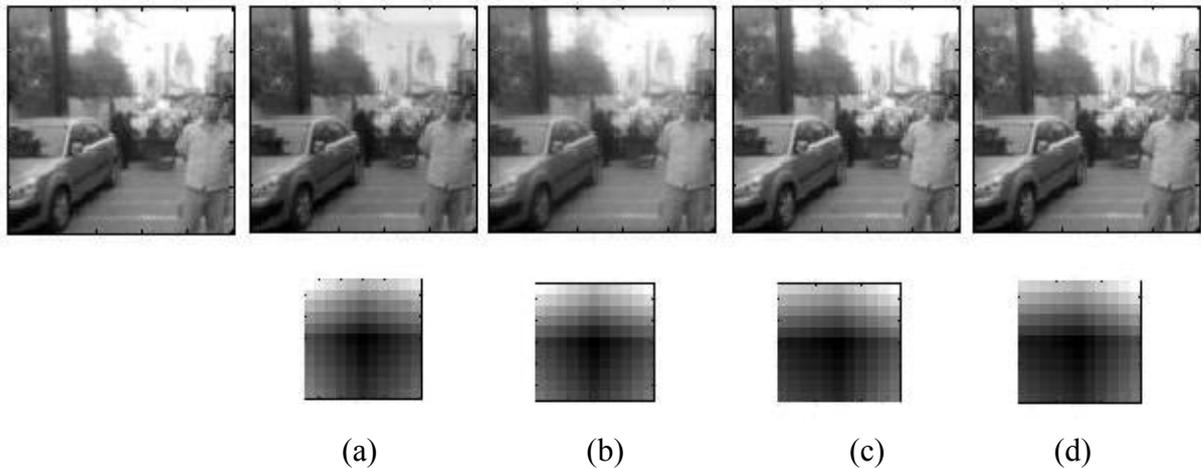
(b)

*Figure V.42. Images floues simulées à longue exposition des photographies.*

Quand j'ai un problème avec la grande dimension, je travaille avec ces images mais avec réduction du nombre de pixels.

L'image restaurée obtenus par la méthode proposée en algorithmel (page: 76) et soumise à ces conditions: dimension de l'image 100x100, dimension d'un PSF restaurée 11x11.

Les images sont présentées sur la ligne supérieure de la Fig. V.43, et les PSFs sont représentés sur la ligne inférieure de la même figure.



*Figure V.43. Résultats de la restauration en utilisant l'algorithme proposé. Les images restaurées sont présentées dans la rangée du haut, et les PSF correspondantes récupérées sont présentés ci-dessous les images. Les valeurs des PSF sont linéairement associées à la plage [0:255] des fins de visualisation.*

La Fig. V.43 représente l'image et le noyau restaurés après chaque itération, et comme nous avons vu il y a de changement important de qualité et de vision d'image. Donc il est important vue la progression d'erreur après chaque itération. L'erreur finale égale  $7.6872e-006$  entre l'image originale et l'image restaurée. Le temps consommé est 78909.647378 secondes. La taille de PSF estimé 11x11. Dimension de l'image estimée 100x100.

Par comparaison avec l'image originale sur la Fig. V.43, il est clair que l'algorithme proposé fournit des images restaurées avec une qualité visuelle élevée.

La moyenne d'erreur quadratique (MSE) entre l'image originale et l'image restaurée est  $2.2922e+004$ ,  $SNR=1.3783e-011$ . Les résultats quantitatifs MSE et l'inspection visuelle des PSF récupérés suggèrent que l'algorithme est très efficace pour estimer les PSF d'origine.

### **5.3. Restauration des images de synthèse : Modele3**

Modele3 représente mon image dans l'ABC Mall: Cette image prise en plein d'air sous condition d'éclairage faible ISO égale 650.

L'image de l'ABC est utilisée pour créer l'image observée  $y_1$  et  $y_2$  du modèle proposé aux équations (4.1 et 4.2). L'image observée  $y_2$ , Fig. V.44 (b), est obtenue en ajoutant un bruit

blanc gaussien de variance 45 à l'image d'origine (SNR = 12.2690 et PSNR= -11.8575 dB).

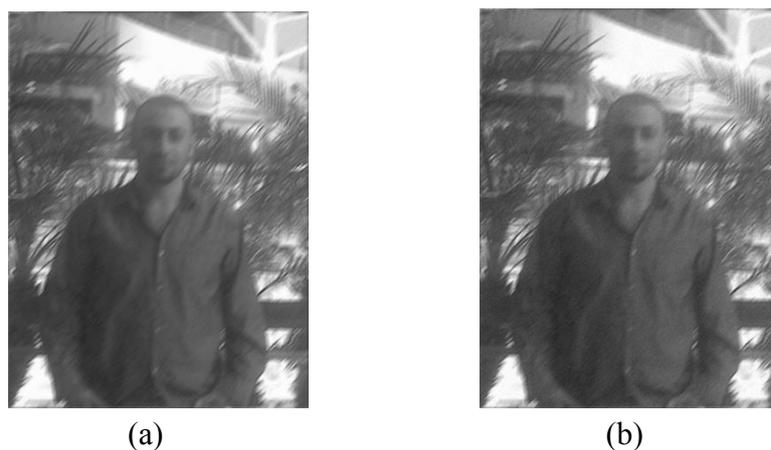
Cette image souffre d'un haut niveau de bruit: la moyenne l'erreur quadratique (MSE) entre cette image et l'image originale est 15.3375.



*Figure V.44. (a) Image originale de l'ABC, (b) Image bruitée simulée avec acquisition à courte exposition.*

Pour générer l'observation  $y_1$ , on applique sur l'image d'origine un noyau flou mouvement (Motion blur) de dimension  $15 \times 15$  et angle d'orientation égal 45 degré représenté par la Fig. V.45 (a), qui est en principe similaire au mouvement de la caméra pendant de longues expositions.

Puis sur l'image floue de Fig. V.45(a) un bruit gaussien est ajouté. Plus spécifiquement, un bruit blanc gaussien avec une variance de 40 est ajouté aux images floues pour obtenir ces dernières images observées  $y_1$  Fig. V.45 (b) avec SNR égale 4.3244, PSNR= -16.4374 dB et avec MSE=44.0291.

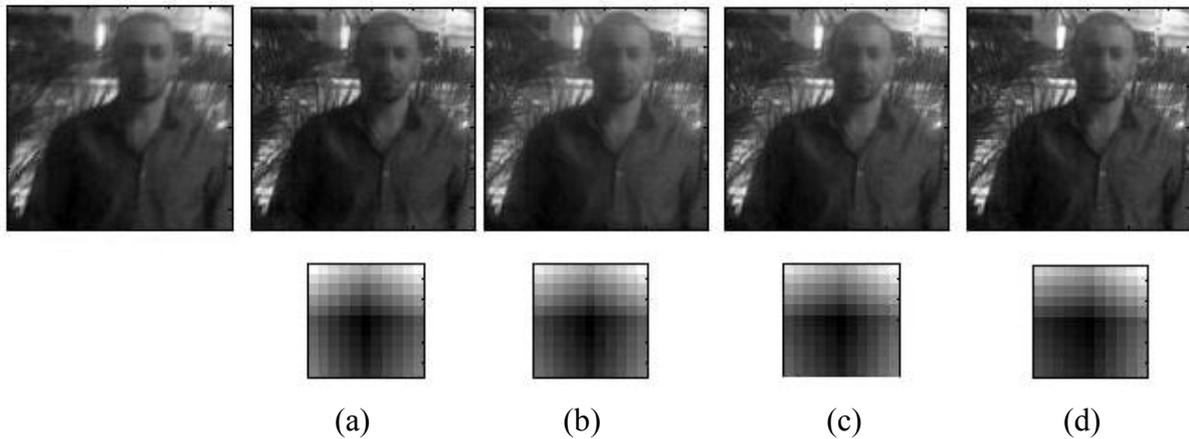


*Figure V.45. Images floues simulant à longue exposition des photographies.*

Le problème de grande dimension est résolu en réduisant le nombre de pixels.

L'image restaurée obtenues par la méthode proposée en algorithme1 (page: 76) est soumise à: dimension de l'image 110x110, dimension du PSF restaurée 11x11.

Les images sont présentées sur la ligne supérieure de la Fig. V.46, et les PSFs sont représentés sur la ligne inférieure de la même figure.



*Figure V.46. Résultats de la restauration en utilisant l'algorithme proposé. Les images restaurées sont présentées dans la rangée du haut, et les PSFs correspondantes récupérées sont présentés ci-dessous les images. Les valeurs des PSF sont linéairement associées à la plage [0:255] des fins de visualisation.*

La Fig. V.46 représente l'image et le noyau restaurée après chaque itération, et comme nous avons vu il y a de changement important de qualité et de vision d'image. Donc il est important de suivre la progression d'erreur aux itérations. L'erreur finale est égale 1.481 e-001 entre l'image originale et l'image restaurée. Le temps consomme est 26178.236322 secondes. La taille de PSF estime 11x11. Dimension de l'image estime 110x110.

Par comparaison avec l'image originale sur la Fig. V.46, il est clair que l'algorithme proposé fournit des images restaurées avec une qualité visuelle élevée.

La moyenne d'erreur quadratique (MSE) entre l'image originale et l'image restaurée est 1.1477e+004, SNR=2.9351e-010. Les résultats quantitatifs MSE et l'inspection visuelle des PSF récupérés suggèrent que l'algorithme est très efficace pour estimer les PSF d'origine.

#### **5.4. Restauration des images de synthèse : Modele4**

L'image modele4 est une image texture: « CHEHAITLY ».

L'image CHEHAITLY est utilisée pour créer l'image observée  $y_1$  et  $y_2$  du modèle propose en

équations (4.1 et 4.2). L'image observée  $y_2$ , Fig. V.44 (b), est obtenue en ajoutant un bruit blanc gaussien de variance 45 à l'image d'origine (SNR = 23.4865 et PSNR= 0.3111 dB). Cette image souffre d'un haut niveau de bruit: la MSE entre cette image et l'image originale est 0.9309.



**CHEHAITLY  
CHEHAITLY**

(a)



(b)

*Figure V.47. (a) Image originale CHEHAITLY, (b) Image bruitée simuler d'une acquisition à court d'exposition.*

Pour générer l'observation  $y_1$ , on applique sur l'image d'origine un noyau flou Gaussien de dimension 15x15 et par variance égale 40 représenté dans la Fig. V.47 (a), qui est en principe similaire au mouvement de la caméra pendant de longues expositions.

Puis sur l'image floue de Fig. V.47(a) un bruit de poisson est ajouté. Pour obtenir l'image observées  $y_1$  Fig. V.47 (b) avec SNR égale 11.3753, PSNR= -11.8265 dB et avec erreur moyen quadratique MSE=15.2283.



**CHEHAITLY  
CHEHAITLY**

(a)

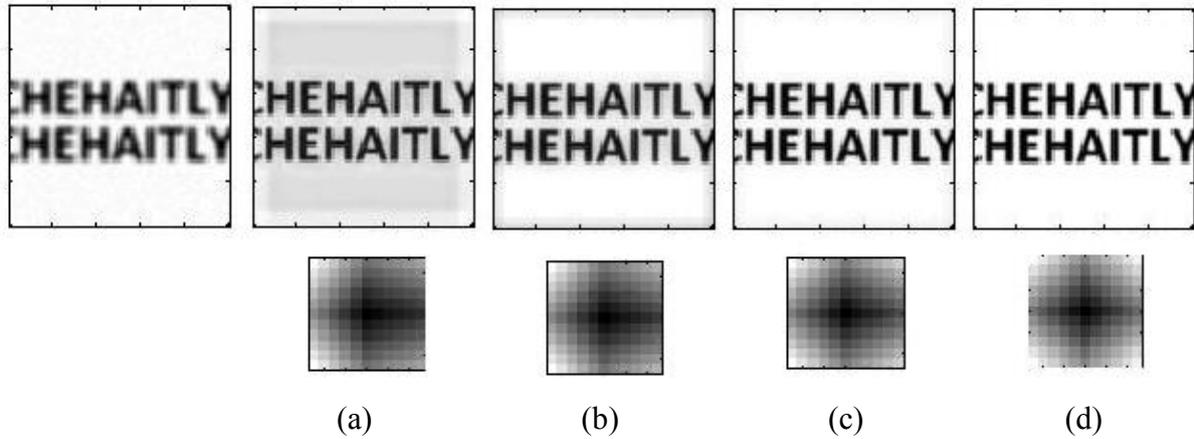


(b)

*Figure V.47. Images floues simulant à longue exposition des photographies.*

L'image restaurée obtenues par la méthode proposée en algorithmel (page: 76) est telle que: dimension de l'image 110x110, dimension d'un PSF restaurée 11x11.

Les images sont présentées sur la ligne supérieure de la Fig. V.48, et les PSF's sont représentés sur la ligne inférieure de la même figure.



*Figure V.48. Résultats de la restauration en utilisant l'algorithme proposé. Les images restaurées sont présentées dans la rangée du haut, et les PSFs correspondantes récupérées sont présentés ci-dessous les images. Les valeurs des PSF sont linéairement associées à la plage [0:255] des fins de visualisation.*

La Fig. V.47 représente l'image et le noyau restaurée après chaque itération, et comme nous avons vu il y a de changement important de qualité et de vision d'image. Donc il est important de suivre la progression d'erreur aux itérations. L'erreur finale est égale  $7.3582e-006$  entre l'image originale et l'image restaurée. Le temps consommé est 82352.813251 secondes. La taille de PSF estime 11x11. Dimension de l'image estime 110x110.

Par comparaison avec l'image originale sur la Fig. V.47, il est clair que l'algorithme proposé fournit des images restaurées avec une qualité visuelle élevée.

La moyenne d'erreur quadratique (MSE) entre l'image originale et l'image restaurée est  $5.4391e+004$ , rapport signal sur bruit SNR=  $3.0463e-011$ . Les résultats quantitatifs MSE et l'inspection visuelle des PSF récupérés suggèrent que l'algorithme est très efficace pour estimer les PSF d'origine.

## 6. Restauration des images de synthèse en utilisant l'Algorithme2

Pour les dégradations prises d'images de synthèse, nous avons travaillé sur 2 modèles d'images différentes et sous différentes ISO. En plus, les images et les PSFs sont estimés à partir l'exécution de l'agorithme2 définir page: 77. C'est-à-dire avec la forme anisotropie de TV.

### 6.1. Restauration des images de synthèse : Modele1

Ce modele1 est similaire à la modèle numéros 4 dans le paragraphe précédent.

De même, l'image CHEHAITLY est utilisée pour créer l'image observée  $y_1$  et  $y_2$  du modèle propose en équations (4.1 et 4.2). L'image observée  $y_2$ , Fig. V.44 (b), est obtenue en ajoutant un bruit blanc gaussien de variance 45 à l'image d'origine (SNR = 23.4865 et PSNR= 0.3111 dB). Cette image souffre d'un haut niveau de bruit: la moyenne l'erreur quadratique (MSE) entre cette image et l'image originale est 0.9309.

Pour générer l'observation  $y_1$ , on applique sur l'image d'origine un noyau flou Gaussien de dimension 15x15 et de variance égale 40 représenté dans la Fig. V.47 (a), qui est en principe similaire au mouvement de la caméra pendant de longues expositions. Puis sur l'image floue de Fig. V.47(a) un bruit de poisson est ajouté. Pour obtenir l'image observée  $y_1$  Fig. V.47 (b) avec rapport signal-bruit (SNR) égale 11.3753, PSNR= -11.8265 dB et avec erreur moyen quadratique MSE=15.2283.

L'image restaurée obtenues par la méthode proposée en algorithme2 (page: 77) est telle que: dimension de l'image 100x100, dimension d'un PSF restaurée 11x11.

Les images sont présentées sur la ligne supérieure de la Fig. V.49, et les PSFs sont représentés sur la ligne inférieure de la même figure.

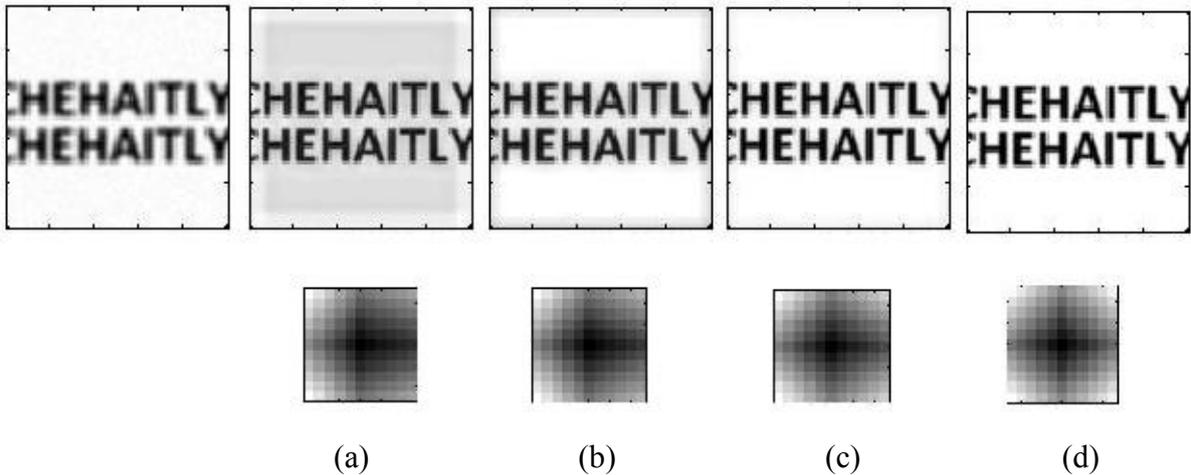


Figure V.49. Résultats de la restauration en utilisant l'algorithme proposé. Les images restaurées sont présentées dans la rangée du haut, et les PSF correspondantes récupérées sont présentés ci-dessous les images. Les valeurs des PSF sont linéairement associées à la plage [0:255] des fins de visualisation.

Par comparaison avec l'image originale sur la Fig. V.49, il est clair que l'algorithme proposé fournit des images restaurées avec une qualité visuelle élevée.

En plus, la moyenne d'erreur quadratique (MSE) entre l'image originale et l'image restaurée est  $5.4391e+004$ , rapport signal sur bruit  $9.4508e-011$ . Les résultats quantitatifs MSE et l'inspection visuelle des PSF récupérés montrent que l'algorithme est très efficace pour estimer les PSF d'origine.

## 6.2. Restauration des images de synthèse : Modele2

Ce modèle est similaire au Modele2 i.e. paragraphe 4.6.2 qui représente mon image dans la vieille marche de Beyrouth.

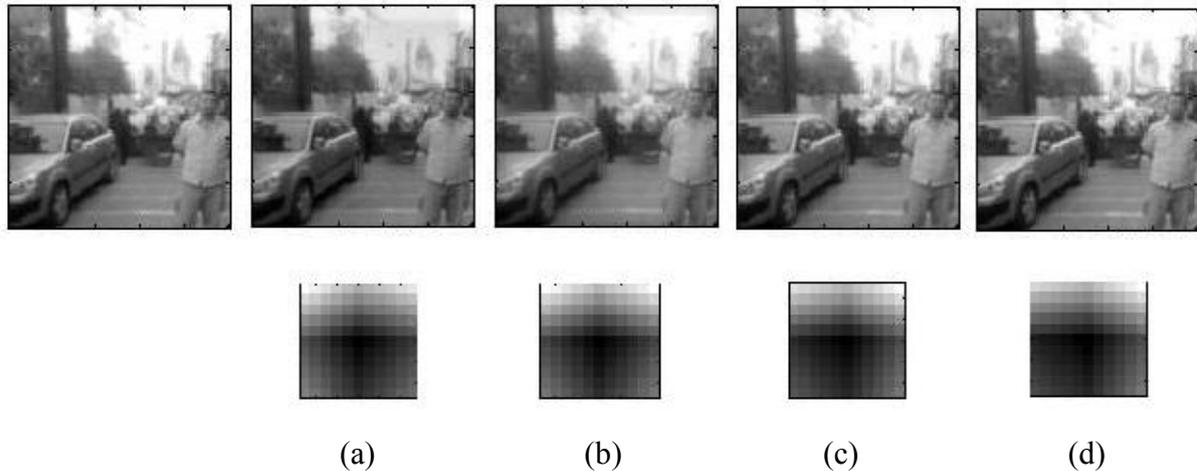
L'image de la vieille marche est utilisée pour créer l'image observée  $y_1$  et  $y_2$  du modèle proposé aux équations (4.1 et 4.2). L'image observée  $y_2$ , Fig. V.41 (b), est obtenue en ajoutant un bruit blanc gaussien de variance 45 à l'image d'origine (SNR = 10.2446 et PSNR = -13.4474 dB). Cette image souffre d'un haut niveau de bruit: la moyenne l'erreur quadratique (MSE) entre cette image et l'image originale est 22.6849.

Nous avons généré l'observation de  $y_1$  par brouillant l'image d'origine par un noyau flou Gaussien de dimension  $15 \times 15$  et variance égale 45 représenté dans la Fig. V.42 (a), qui est en principe similaire au mouvement de la caméra pendant de longues expositions.

Puis sur l'image floue de Fig. V.42(a) un bruit gaussien est ajouté. Plus spécifiquement, un bruit blanc gaussien avec une variance de 40 est ajouté aux images floues pour obtenir ces dernières images observées  $y_1$  Fig. V.42 (b) avec rapport signal-bruit (SNR) égale 5.2922, PSNR = 40 dB et erreur moyenne quadratique MSE = 41.1987.

L'image restaurée obtenues par la méthode proposée en algorithm2 (page: 77) est telle que: dimension de l'image 100x100, dimension d'un PSF restaure est 11x11

Les images sont présentées sur la ligne supérieure de la Fig. V.43, et les PSFs sont représentés sur la ligne inférieure de la même figure.



*Figure V.50. Résultats de la restauration en utilisant l'algorithme proposé. Les images restaurées sont présentées dans la rangée du haut, et les PSFs correspondantes récupérées sont présentés en dessous des images. Les valeurs des PSF sont linéairement associées à la plage [0:255] des fins de visualisation.*

La Fig. V.50 représente l'image et le noyau restauré après chaque itération, et on remarque un changement important de qualité d'image. Donc il est important de suivre la progression d'erreur aux itérations. L'erreur finale égale 0.1911 (nombre d'itération égale 15) entre l'image originale et l'image restaurée. Le temps consommé est 24719.368765 secondes. La taille de PSF estime 11x11. Dimension de l'image estimée 100x100.

Par comparaison avec l'image originale sur la Fig. V.43, il est clair que l'algorithme proposé fournit des images restaurées avec une qualité visuelle important.

La moyenne d'erreur quadratique (MSE) entre l'image originale et l'image restaurée est 2.2922e+004, rapport signal sur bruit SNR=2.8280e-011. Les résultats quantitatifs MSE et l'inspection visuelle des PSF récupérés suggèrent que l'algorithme est très efficace pour estimer les PSF d'origine.

## 7. Restauration avec fonction déconvolution Aveugle défini en Matlab

En Matlab, il y a une fonction spécifique pour effectuer l'opération de déconvolution aveugle qui s'appelle `blinddeconvolution.m`. On note que cette fonction respecte l'algorithme proposé par Richardson et Lucy.

Pour faire une comparaison entre cette fonction et celle proposée dans ce travail est faite: j'écris la fonction « `Richardson_Lucy_algorithm.m` » voir annexe A. Puis j'exécute cette méthode sur la même figure.

Les résultats sont enregistrés dans les paragraphes suivants.

### 7.1. Déconvolution Aveugle en Matlab: Modele1

Modele1 représente l'image de l'artiste: Mademoiselle Scarlet Jonson.

L'image de Scarlet est utilisée pour créer l'image observée  $y_1$  et  $y_2$  du modèle proposé aux équations (4.1 et 4.2). L'image observée  $y_2$ , Fig. V.38 (b), est obtenue en ajoutant un bruit blanc gaussien de variance 35 à l'image d'origine (SNR = 27.1412 dB). Cette image souffre d'un haut niveau de bruit: la moyenne l'erreur quadratique (MSE) entre cette image et l'image originale est 18.5676.

Pour obtenir l'image observé  $y_1$ , nous utilisons l'image de Fig. V.39 (b) puis on ajoute un bruit blanc gaussien avec une variance de 35 avec rapport signal-bruit de 40 dB.

Les images restaurées obtenues par la méthode proposée « `Richardson_Lucy_algorithm.m` » sont présentées ci-dessous.

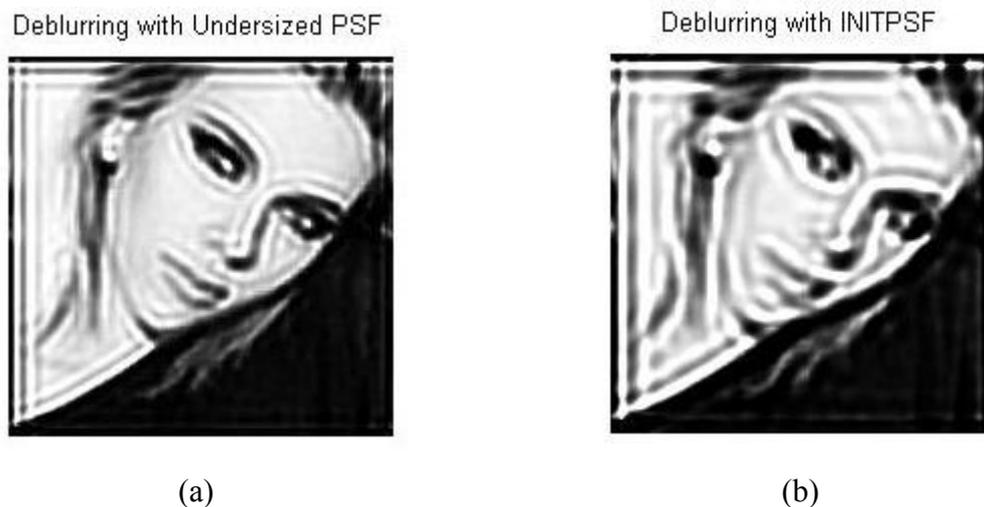


Figure V.51. (a).Première étape: restauration avec minimisation de la taille de PSF ou (with Undersized PSF). (b).Deuxième étape: restauration avec la même taille

initiale (with true PSF).

Deblurring with Oversized PSF

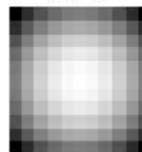


Figure V.52. Troisième étape: restauration avec augmentation de la taille initiale, avec utilisation le masque de Sobel pour reconstruire le contour.

Deblurred Image



True PSF



Reconstructed PSF

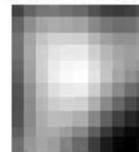


Figure V.53. Restauration de l'image avec le PSF correspondant en utilisant la routine deconvblind dans MATLAB.

La moyenne d'erreur quadratique (MSE) entre l'image originale et l'image restaurée est 76.7694, ce qui montre que l'image restaurée est différente de l'image originale. Aussi, le rapport signal sur bruit est  $SNR = 4.7333$ .

## 7.2. Déconvolution Aveugle en Matlab.: Modele2

Image texture: « CHEHAITLY ».

L'image CHEHAITLY est utilisée pour créer l'image observée  $y_1$  et  $y_2$  du modèle proposé aux équations (4.1 et 4.2). L'image observée  $y_2$ , Fig. V.44 (b), est obtenue en ajoutant un bruit blanc gaussien de variance 45 à l'image d'origine ( $SNR = 23.4865$  et  $PSNR = 0.3111$  dB). Cette image souffre d'un haut niveau de bruit: la moyenne l'erreur quadratique (MSE)

entre cette image et l'image originale est 0.9309.

Pour générer l'observation de  $y_1$ , on applique sur l'image d'origine un noyau flou Gaussien de dimension 15x15 et variance égale 40 représenté dans la Fig. V.47 (a), qui est en principe similaire au mouvement de la caméra pendant de longues expositions.

Puis sur l'image floue de Fig. V.47(a) un bruit de poisson est ajouté. Pour obtenir l'image observés  $y_1$  Fig. V.47 (b) avec rapport signal-bruit (SNR) égale 11.3753, PSNR= -11.8265 dB et avec erreur moyen quadratique MSE=15.2283.

Les images restaurées obtenues par la méthode proposée « Richardson\_Lucy\_algorithm.m » sont présentées ci-dessous.



Figure V.54. (a).Première étape: restauration avec minimisation de la taille de PSF ou (with Undersized PSF). (b).Deuxième étape: restauration avec la même taille initiale (with true PSF)

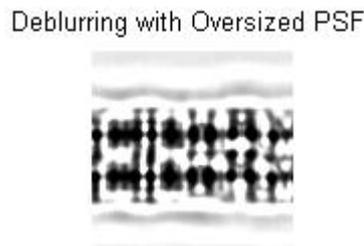


Figure V.55. Troisième étape: restauration avec augmentation de la taille initiale, avec utilisation le masque de Sobel pour construire le contour.

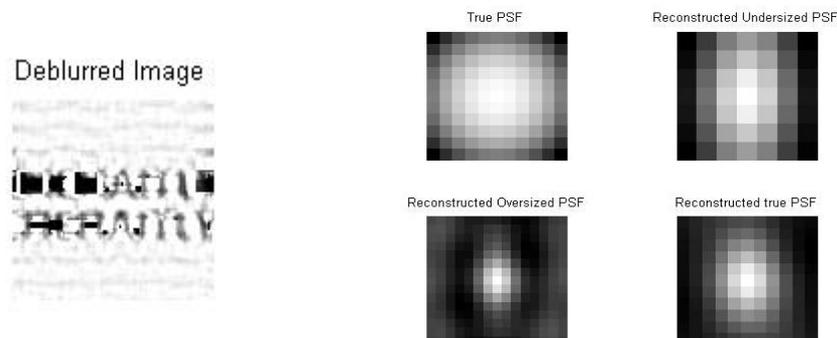


Figure V.55. Restauration de l'image avec le PSF correspondant en utilisant la

*routine deconvblind dans MATLAB.*

La moyenne d'erreur quadratique (MSE) entre l'image originale et l'image restaurée est 43.5591, ce qui montre que l'image restaurée est très loin de l'image originale. Aussi, le rapport signal sur bruit est  $SNR = 7.1465$ .

Il est clair que la qualité des images qui en résultent restaurés, présentés sur la Fig. V.55 et V.53 est beaucoup plus faible que celle de la méthode proposée.

## **8. Conclusions**

Dans ce chapitre, nous avons présenté une nouvelle formulation bayésienne pour la déconvolution aveugle de paires d'images acquises à long et à court durées d'exposition. Nous avons présenté cette formulation en deux algorithmes (1 et 2).

Les paramètres de l'image inconnue, flou et tous les modèles, y compris les variances de bruit, sont estimés uniquement à partir des observations sans connaissance préalable ou intervention de l'utilisateur. D'autre part, le cadre proposé est très souple de sorte que lorsque certaines connaissances a priori sur les inconnues sont disponibles, on peut facilement être les intégrer dans l'algorithme.

Les algorithmes développés permettent d'estimer simultanément les distributions des inconnues. Les algorithmes ne dépendent pas le seuillage de flou ou de débruitage comme de méthodes ad hoc utilisées dans la proposition Matlab. En outre, bien que la méthode proposée ne nécessite aucune intervention de l'utilisateur, elle fournit une estimation entièrement automatisée des paramètres algorithmiques; En plus les résultats expérimentaux démontrent des images à haute qualité de restauration, même pour des dégradations élevées dans le cas d'images synthétiques; et ceci permet favorablement une comparaison avec des autres méthodes existantes.

## Conclusion générale

Cette mémoire se situe dans le cadre d'une contribution au problème du déflouage d'images. Nous nous sommes placés dans le cadre délicat de la restauration dite déconvolution aveugle, pour laquelle l'opérateur de flou affectant la scène considérée est inconnu mais supposé le même sur la totalité de l'image c'est-à-dire flou homogène sur l'image.

La motivation de ce travail était double : la première pourrait être qualifiée académique, dans la mesure où le nombre de travaux traitant cette problématique aveugle est assez restreint [11] et [125]. La deuxième était la prise en compte d'une application pratique sous-jacente, qui consiste à examiner la faisabilité de l'emploi de telles techniques à la restauration d'images obtenues par camera stabilisé sur un drone d'hélicoptère, à des fins d'auscultation d'ouvrages de génie civil particuliers.

Nous avons dans une première phase dégagée les principaux éléments théoriques nécessaires à la compréhension du problème fondamental de restauration d'images floues, qui se modélise principalement sous la forme d'une déconvolution. Celui-ci rentre dans la classe des problèmes inverses mal posés, c'est-à-dire instables. Nous avons mis en évidence les liens existants entre la restauration classique, c'est-à-dire à noyau de flou connu, et la restauration aveugle.

Nous nous sommes alors mêmés à étudier état de l'art de différentes méthodes pouvant exister en déconvolution d'images (monocanal). Celui-ci nous a permis de bénéficier d'un aperçu global des techniques disponibles, de leurs performances, et de pouvoir retirer celles qui apparaissaient les plus performantes, les plus robustes, et les plus adaptées à notre contexte. Ces méthodes reposent sur une formulation variationnelle, et leur solution consiste donc en la recherche d'un extremum d'une fonctionnelle donnée, dont la minimisation modélise le problème de déconvolution.

Nous avons dans un premier temps envisagé un modèle de déconvolution basé sur l'utilisation d'une régularisation à variation totale (TV) isotropie, comme proposé par Chan et Wong [94], puis nous avons dans un deuxième temps envisagé un modèle basé sur l'utilisation d'une régularisation à variation totale anisotropie, ceci pour le traitement d'images monochromes.

Nous avons montré expérimentalement l'efficacité de notre approche, et les résultats comparatifs menés vis-à-vis d'une technique de l'état de l'art.

Nous avons tout d'abord mis en évidence la difficulté à appliquer le principe de déconvolution aveugle avec concept Bayésien à de tels cas, en raison de problèmes de convergence des estimations, et d'une incongruité majeure (devoir régler un paramètre expérimentalement pour en estimer un autre). Nous avons alors employé une des techniques de restauration variationnelle, à la fois avec un terme de fidélité quadratique, afin de déterminer si ce dernier pouvait apporter un bénéfice, au vu de l'erreur commise sur les paramètres caractérisant le flou considéré. Nous avons cependant constaté qu'il n'en était rien,

et que - surtout en présence de bruit - il valait mieux utiliser un opérateur quadratique, de type moindres carrés.

Il reste dans ce sujet de nombreuses voies à explorer, et les perspectives de développement sont très riches. Parmi celles qui nous semblent présenter un intérêt majeur, on pourra citer:

- Une meilleure définition des contraintes physiques à imposer aux noyaux de flou. On a vu par exemple que des mouvements non-rectilignes étaient beaucoup plus délicats à traiter que des mouvements linéaires. On pourrait alors envisager d'instaurer une contrainte visant à forcer la reconstruction à former un noyau ayant la forme d'une ligne courbe, continue. Il pourrait aussi être judicieux, dans certains cas, d'imposer à chaque pixel de ce noyau reconstruit une valeur identique, de sorte que son énergie ne soit pas concentrée systématiquement au centre de celui-ci, mais bien « étalée » sur toute la longueur du support.
- Bien entendu, l'ajout de telles contraintes ne serait pas sans poser de sérieux problèmes dans la définition de schémas de minimisation. Nous pensons qu'une manière plus pertinente de traiter ces contraintes serait alors d'envisager des algorithmes d'optimisation sous contrainte, afin d'éviter les méthodes usuelles procédant avec des algorithmes non-contraints suivis de diverses normalisation, seuillage, et autres opérations fortement non-linéaires. Ces dernières années ont vu l'émergence de ces méthodes en restauration variationnelle à noyau connu, mais leur utilisation dans un contexte aveugle reste encore à développer [82].
- Dans le contexte des images en couleur. Pour ce qui est de la déconvolution avec noyaux paramétriques, notre approche pourrait sans doute voir sa précision et/ou sa robustesse améliorée par l'utilisation de filtres de choc adaptés aux images couleurs, qui viseraient à bien aligner les canaux de l'image. Cela n'est pas à proprement parlé une perspective de développement de restauration d'image, mais peut présenter un intérêt certain en traitement de l'image de manière plus générale.
- Dans le même ordre d'idées, insistons sur le fait que, pour qu'un essai de déconvolution aveugle fournisse un résultat acceptable, il est primordial de disposer du maximum d'informations sur la scène que l'on observe et/ou sur la dégradation en jeu. A ce titre, il nous paraît fondamental que soit bien définis dans l'application les éléments que l'on observe, et que l'on souhaite voir ou détecter, avec leurs caractéristiques précises. Ceci afin d'arriver à des modèles d'images plus précis que ceux qui ont pu être mis en œuvre dans ce travail. On a par exemple pu observer des modèles prenant en compte la distribution des valeurs de gradient ou de laplacien dans les images, et ceci avec des résultats intéressants.
- Enfin, comme la simulation est effectuée en Matlab (problème au niveau de temps consommé et taille de mémoire) et pour améliorer le traitement d'image en générale: il faut essayer de travailler dans un environnement d'exécution parallèle par exemple:

avec logiciel Cuda développé par nVIDIA. Ou avec autre système d'opération comme Linux par exemple: Fedora.

Ces perspectives et recommandations ne sont bien sûr pas exhaustives, et le sujet de la déconvolution aveugle reste pour sûr un domaine de recherche encore empli de potentialités, potentialités qui se développeront parallèlement à l'augmentation des capacités de calcul des ordinateurs, permettant d'exécuter en un temps acceptable des algorithmes de plus en plus complexes.

## Annexe A

### Richardson\_Lucy\_algorithm.m

Ce fonction restaure l'image et le PSF simultanément d'une façon itérative. A chaque itération, il utilise l'algorithme d'accélération et l'algorithme déprime de Lucy- Richardson.

En effet, l'algorithme de Lucy- Richardson est utilisé lorsque le PSF (noyau flou) est connue, tandis que aucune information sur le bruit nécessaire. Les images floue et bruite sont restaures d'une façon itérative.

En plus, nous avons additionné le caractéristique d'un système optique (d'un camera) pour calculer les valeurs initiales de paramètres et bien sûr pour augmenter la qualité d'image restaure.

Corps de fonction: Richardson\_Lucy\_algorithm.m

```
% Step 1: Read Image
% Step 2: Simulate a Blur
% Step 3: Restore the Blurred Image Using PSFs of Various Sizes
% Step 4: Analyzing the Restored PSF
% Step 5: Improving the Restoration
% Step 6: Using Additional Constraints on the PSF Restoration

close all
clear all
clc

% Step 1: Read Image

% The example reads in an intensity image. The deconvblind function can handle arrays of %any dimension.
% Load originale image
y1_orig=imread('C:\Users\abc\Desktop\GUI Interface for My Project\Image\12\12_double.jpg');
y1_orig=rgb2gray(y1_orig);
y1_orig = impyramid(y1_orig, 'reduce');% reduce the size of image of half in evry operation
y1_orig = impyramid(y1_orig, 'reduce');
y1_orig=y1_orig(11:110,39:138);
I=y1_orig;
figure;imshow(uint8(I));title('Original Image');

% Step 2: Simulate a Blur

% Simulate a real-life image that could be blurred (e.g., due to camera motion or lack of focus).
% The example simulates the blur by convolving a Gaussian filter with the true image (using imfilter).
% The Gaussian filter then represents a point-spread function, PSF.
PSF = fspecial('gaussian',11,40);
y1_matrix=imread('C:\Users\abc\Desktop\GUI Interface for My Project\Image\12\
12_double_GaussianBlur_15_40_PoissonNoise.jpg');
y1_matrix=rgb2gray(y1_matrix);
y1_matrix = impyramid(y1_matrix, 'reduce');% reduce the size of image of half in evry operation
y1_matrix = impyramid(y1_matrix, 'reduce');
```

```
Blurred=y1_matrix(11:110,39:138);
figure;imshow(Blurred);title('Blurred Image');
```

```
% Step 3: Restore the Blurred Image Using PSFs of Various Sizes
```

```
% To illustrate the importance of knowing the size of the true PSF, this example performs three restorations.
% Each time the PSF reconstruction starts from a uniform array--an array of ones.
% The first restoration, J1 and P1, uses an undersized array, UNDERPSF, for an initial guess of the PSF.
% The size of the UNDERPSF array is 4 pixels shorter in each dimension than the true PSF. |
```

```
UNDERPSF = ones(size(PSF)-4);
[J1 P1] = deconvblind(Blurred,UNDERPSF);
figure;imshow(J1);title('Deblurring with Undersized PSF');
```

```
%The second restoration, J2 and P2, uses an array of ones, OVERPSF, for an initial PSF that is 4 pixels longer
%in each dimension than the true PSF.
```

```
OVERPSF = padarray(UNDERPSF,[4 4],'replicate','both');
[J2 P2] = deconvblind(Blurred,OVERPSF);
figure;imshow(J2);title('Deblurring with Oversized PSF');
```

```
%The third restoration, J3 and P3, uses an array of ones, INITPSF, for an initial PSF that is exactly of the same
size as the true PSF.
```

```
INITPSF = padarray(UNDERPSF,[2 2],'replicate','both');
[J3 P3] = deconvblind(Blurred,INITPSF);
figure;imshow(J3);title('Deblurring with INITPSF');
```

```
% |Step 4: Analyzing the Restored PSF
```

```
% All three restorations also produce a PSF. The following pictures show how the analysis of the reconstructed
%PSF might help in guessing the right size for the initial PSF. In the true PSF, a Gaussian filter, the maximum
%values are at the center (white) and diminish at the borders (black). |
```

```
figure;subplot(221);imshow(PSF,[],'InitialMagnification','fit');
title('True PSF');
subplot(222);imshow(P1,[],'InitialMagnification','fit');
title('Reconstructed Undersized PSF');
subplot(223);imshow(P2,[],'InitialMagnification','fit');
title('Reconstructed Oversized PSF');
subplot(224);imshow(P3,[],'InitialMagnification','fit');
title('Reconstructed true PSF');
```

```
% |Step 5: Improving the Restoration
```

```
%The ringing in the restored image, J3, occurs along the areas of sharp intensity contrast in the image and along
%the image borders. This example shows how to reduce the ringing effect by specifying a weighting function.
%The algorithm weights each pixel according to the WEIGHT array while restoring the image and the PSF. In
%our example, we start by finding the "sharp" pixels using the edge function. By trial and error, we determine
%that a desirable threshold level is 0.3.
```

```
WEIGHT = edge(I,'sobel',.3);
```

```
%To widen the area, we use imdilate and pass in a structuring element, se.
```

```

se = strel('disk',2);
WEIGHT = 1-double(imdilate(WEIGHT,se));

%The pixels close to the borders are also assigned the value 0.

WEIGHT([1:3 end-[0:2]],:) = 0;
WEIGHT(:,[1:3 end-[0:2]]) = 0;
figure;imshow(WEIGHT);title('Weight array');

%The image is restored by calling deconvblind with the WEIGHT array and an increased number of iterations
%(30).
%Almost all the ringing is suppressed.

[J P] = deconvblind(Blurred,INITPSF,30,[],WEIGHT);
figure;imshow(J);title('Deblurred Image');

% |Step 6: Using Additional Constraints on the PSF Restoration

%The example shows how you can specify additional constraints on the PSF. The function, FUN, below returns
%a modified PSF array which deconvblind uses for the next iteration. In this example, FUN modifies the PSF
%by cropping it by P1 and P2 number of pixels in each dimension, and then padding the array back to its
%original size with zeros. This operation does not change the values in the center of the PSF,
% but effectively reduces the PSF size by 2*P1 and 2*P2 pixels.

P1 = 2;
P2 = 2;
FUN = @(PSF) padarray(PSF(P1+1:end-P1,P2+1:end-P2),[P1 P2]);

% The anonymous function, FUN, is passed into deconvblind last. See the section Parameterizing Functions, in
%the MATLAB Mathematics documentation, for information about providing additional parameters to the
%function FUN. In this example, the size of the initial PSF, OVERPSF, is 4 pixels larger than the true PSF.
%Setting P1 = 2 and P2 = 2 as parameters in FUN effectively makes the valuable space in OVERPSF the same
%size as the true PSF. Therefore, the outcome, JF and PF, is similar to the result of deconvolution with the right
% sized PSF and no FUN call, J and P, from step 4.

[JF PF] = deconvblind(Blurred,OVERPSF,30,[],WEIGHT,FUN);
figure;imshow(JF);title('Deblurred Image');

```

## Bibliographie

- [1] T.F. Chan et J. Shen. Image processing and analysis: variational, PDE, wavelet, and stochastic methods. SIAM, 2005.
- [2] R. A. Adams et J. J. F. Fournier. Sobolev spaces. Academic Press, 2003.
- [3] M. Sorel, F. Sroubek et J. Flusser: Multichannel deblurring of digital images. *Kybernetika*, 47(3), 2011.
- [4] S. Stryhanyn-Chardon. Contribution au problème de la restauration myope des images numériques : analyse et synthèse. *Dissertatio doctoralis*, Université de RennesI, 1997.
- [5] J. Hadamard. *Lectures on Cauchy's Problem in Linear Partial Differential Equations*. Yale University Press, New Haven, 1923.
- [6] G. Demoment. Image reconstruction and restoration: overview of common estimation structures and problems. *IEEE Trans. on ASSP*, 37(12): 2024-2036, Dec. 1989.
- [7] A.K. Jain. *Fundamentals of digital image processing*. information and system sciences series. Prentice Hall, 1989.
- [8] A.K. Katsaggelos. *Digital image restoration*, volume 23 of Springer Series in Information Sciences. Springer-Verlag, 1991.
- [9] M. Bertero et P. Boccacci. Introduction to inverse problems in imaging. IOP Publishing, 1998. code. *IEEE Trans. on Comm.*, 31: 532-540, 1983.
- [10] A.S. Carasso. Linear and nonlinear image deblurring: a documented study. *SIAM J. Numer. Anal.*, 36(6): 1659-1689, 1999
- [11] R. Molina, J. Nuñez et F.J. Cortijo. Image restoration in Astronomy: a bayesian perspective. *IEEE Signal Processing Magazine*, pages 11-29, March 2001.
- [12] R.E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME. Ser. D. J. Basic Engineering*, 82: 35-45, 1960.
- [13] L. Landweber. An iteration formula for Fredholm integral equations of the 1rst kind. *Amer. J. Math.*, 73: 615-624, 1951.
- [14] L.A. Shepp et Y. Vardi. Maximum likelihood reconstruction for emission tomography. *IEEE Trans. on MI*, 1: 113-122, 1982.
- [15] L.B. Lucy. An iterative technique for the rectification of observed images. *The Astronomical Journal*, 79(6): 745-754, June 1974.
- [16] A.N. Tikhonov. Regularization of incorrectly posed problems. *Sov. Math. Dokl.*, 4: 1624-1627, 1963.
- [17] A. Levin, Y. Weiss, F. Durand, W.T. Freeman. *Efficient Marginal Likelihood Optimization in Blind Déconvolution*. Weizmann Institute of Science, MIT CSAIL, 2011.
- [18] R.L. Lagendijk, J. Biemond et D.E. Boeke. Identification and restoration of noisy blurred images using the expectation-maximization algorithm. *IEEE Trans. on ASSP*, 38(7), July 1990.
- [19] M. Sigelle. Simultaneous image restoration and hyperparameter estimation for incomplete data by a cumulant analysis. *INRIA Research Report*, 3249, Sept. 1997.
- [20] S.D. Babacan, J.Wang, R. Molina, A.K. Katsaggelos. Bayesian blind déconvolution from differently exposed image pairs. EECS Department Northwestern University, Evanston, IL USA. Departamento de Ciencias de la Computacione I.A.Universidad de Granada, Granada, Spain. 2011

- [21] S. Mallat. *A Wavelet tour of signal processing*. Academic Press, 1998.
- [22] P. Campisi et K. Egiazarian. *Blind image déconvolution, theory and applications*. CRC Press, 2007.
- [23] M. Cannon. Blind déconvolution of spatially invariant image blurs with phase. *IEEE Transactions on Acoustic, Speech, and Signal Processing*, 24(1), 1976.
- [24] T.G. Stockham, T.M. Cannon et R.B. Ingebretsen: Blind déconvolution through digital signal processing. *Proceedings IEEE*, 63(4), 1975.
- [25] N. Nguyen et P. Milanfar: Efficient generalized cross-validation with applications to parametric image image restoration and resolution enhancement. *IEEE Transactions on Image Processing*, 10(9), 2001.
- [26] S.J. Reeves et R.M. Mersereau. Blur identification by the method of generalized cross-validation. *IEEE Transactions on Image Processing*, 1(3), 1992.
- [27] Y. Yitzhaky et N.S. Kopeika. Identification of blur parameters from motion blurred images. *Graphical Models and Image Processing*, 59, 1997.
- [28] A. Carasso: Direct blind déconvolution. *SIAM Journal of Applied Mathematics*, 61(6), 2001.
- [29] A. Carasso. The APEX method in image sharpening and the use of low exponent Lévy stable laws. *SIAM Journal of Applied Mathematics*, 63(2), 2002.
- [30] Y. Yitzhaky, I. Mor, A. Lantzman et N.S. Kopeika. Direct method for restoration of motion-blurred images. *Journal of the Optical Society of America A*, 15(6), 1998.
- [31] Y. Yitzhaky, R. Milberg, S. Yohaev et N.S. Kopeika. Comparison of direct blind déconvolution methods for motion-blurred images. *Applied Optics*, 38(20), 1999.
- [32] D.P.K. Lun, T.C.L. Chan, T.C. Hsung, D.D. Feng et Y.H. Chan. Efficient blind image restoration using discrete periodic Radon transform. *IEEE Transactions on Image Processing*, 13(2), 2004.
- [33] M.E. Moghaddam et M. Jamzad. Motion blur identification in noisy images using mathematical models and statistical measures. *Pattern Recognition*, 40(7), 2007.
- [34] J.P. Oliveira, M.A.T. Figueiredo et J.M. Bioucas-Dias. Blind estimation of motion blur parameters for image déconvolution. In *Iberian Conference on Pattern Recognition and Image Analysis*, p. 604-611, 2007.
- [35] J.N. Caron, N.M. Namazi, R.L. Lucke, C.J. Rollins et P.R. Lynn. Blind data restoration with an extracted filter function. *Optics Letters*, 26(15), 2001.
- [36] J.N. Caron, N.M. Namazi et C.J. Rollins. Noninteractive blind data restoration by use of an extracted filter function. *Applied Optics*, 41(32), 2002.
- [37] A. Levin. Blind motion deblurring using image statistics. In *Advances in Neural Information Processing Systems (NIPS)*, 2006.
- [38] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis et W. T. Freeman. Removing camera shake from a single photograph. In *SIGGRAPH*, 2006.
- [39] S. Ladjal. Estimation du flou dans les images naturelles. In *Reconnaissance des Formes et Intelligence Artificielle*, 2006.
- [40] J. Jia. Single image motion deblurring using transparency. In *CVPR*, 2007.
- [41] J. Money et S. Kang. Total variation minimizing blind déconvolution with shock filter reference. *Image and Vision Computing*, 26(2), 2008.
- [42] S. Osher et L. Rudin. Feature-oriented image enhancement using shock filters. *SIAM Journal of numerical analysis*, 27(4), 1990.
- [43] D. Strong et T.F. Chan. Relation of regularization parameter and scale in total variation

- based image restoration. Rapport technique, Université de Californie-Los Angeles, 1996.
- [44] J. Yuan et Z. Hu. Image blind déconvolution based on kurtosis extrema. In Fourth International Conference on Photonics and Imaging in Biology and Medicine, 2006
- [45] D. Li, S. Simske et R.M. Mersereau. Blind image déconvolution using constrained variance maximization. In Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems, and Computers, 2004.
- [46] J.W. Miskin et D.J.C. MacKay. Ensemble learning for blind image separation and déconvolution. In Advances in independent component analysis, 2000.
- [47] P.H. van Cittert: Zum Einfluß der Spaltbreite auf die Intensitätsverteilung in Spektrallinien. II. Zeitschrift für Physik, 65(5-6), 1931.
- [48] L. Landweber: An iteration formula for Fredholm integral equations of the first kind. American Journal of Mathematics, 73(3), 1951.
- [49] C.R. Vogel. Computational Methods for Inverse Problems. SIAM, 2002.
- [50] M. Donatelli: A multigrid for image deblurring with Tikhonov regularization. Numerical linear algebra with applications, 12(8), 2005.
- [51] J.M. Bioucas-Dias, M.A.T. Figueiredo et J.P. Oliveira. Adaptive total variation-based image déconvolution: a majorization-minimization approach. In International Conference on Acoustics, Speech, and Signal Processing, 2006.
- [52] J.M. Bioucas-Dias, M.A.T. Figueiredo et J.P. Oliveira. Total variation based image déconvolution: a majorization-minimization approach. In European Signal Processing Conference, 2006.
- [53] J.P. Oliveira, J.M. Bioucas-Dias et M.A.T. Figueiredo. Adaptive total variation image deblurring: a majorization-minimization approach. Signal Processing, 89(9), 2009.
- [54] P.L. Combettes et J.C. Pesquet. Image restoration subject to a total variation constraint. IEEE Transactions on Image Processing, 13(9), 2004.
- [55] A. Chambolle: An algorithm for total variation minimization and applications. Journal of Mathematical Imaging and Vision, 20(1-2), 2004.
- [56] L. Rudin, S. Osher et E. Fatemi. Nonlinear total variation based noise removal algorithms. Physica D, 60(1-4), 1992.
- [57] L. Vese et S.J. Osher. Image denoising and decomposition with total variation minimization and oscillatory functions. Journal of Mathematical Imaging and Vision, 20, 2004.
- [58] C.R. Vogel et M.E. Oman. Iterative methods for total variation denoising. SIAM Journal on Scientific Computing, 17(1), 1996.
- [59] Y. Li et F. Santosa. A computational algorithm for minimizing total variation in image restoration. IEEE Transactions on Image Processing, 5(6), 1996.
- [60] C.R. Vogel et M.E. Oman. Fast, robust total variation-based reconstruction of noisy, blurred images. IEEE Transactions on Image Processing, 7(6), 1998.
- [61] J.M. Bioucas-Dias: Bayesian wavelet-based image déconvolution: A GEM algorithm exploiting a class of heavy-tailed priors. IEEE Transactions on Image Processing, 15(4), 2006.
- [62] M.A.T. Figueiredo et R.D. Nowak. A bound optimization approach to wavelet based image déconvolution. In International Conference on Image Processing, 2005.
- [63] M. Banham et A. Katsaggelos. Spatially adaptive wavelet-based multiscale image restoration. IEEE Transactions on Image Processing, 5(4), 1996.
- [64] M. Mignotte. An adaptive segmentation-based regularization term for image restoration.

In International Conference on Image Processing, 2005.

[65] R. Neelamani, H. Choi et R. Baraniuk. ForWaRD: Fourier-wavelet regularized déconvolution for ill-conditioned systems. *IEEE Transactions on Signal Processing*, 52(2), 2004.

[66] S. Geman et D.E. McClure. Bayesian image analysis: an application to single photon emission tomography. *Proceedings of the American Statistical Association. Statistical Computing Section*, p. 12 18, 1985.

[67] P. Perona et J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7), 1990.

[68] S. Lintner et F. Malgouyres. Solving a variational image restoration model which involves L1 constraints. *Inverse Problems*, 20(3), 2004.

[69] L. Bar, N. Sochen et N. Kiryati. Image deblurring in the presence of impulsive noise. *International Journal of Computer Vision*, 70(3), 2006.

[70] T.F. Chan et S. Esedoglu. Aspects of total variation regularized L1 function approximation. *SIAM J. Appl. Math.*, 65(5), 2005.

[71] H. Fu, M. K. Ng, M. Nikolova et J. L. Barlow. Efficient minimization methods of mixed l2-l1 and l1-l1 norms for image restoration. *SIAM Journal on Scientific Computing*, 27(6), 2006.

[72] M. Welk, D. Theis et J. Weickert. Variational Deblurring of Images with Uncertain and Spatially Variant Blurs, p. 456. *Pattern Recognition: Lectures Notes on Computer Sciences*, 2005.

[73] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40(3), 2002.

[74] N. Mastronardi, P. Lemmerling, A. Kalsi, D. P. O'Leary et S. V. Huffel. Implementation of the regularized structured total least squares algorithms for blind image deblurring. *Linear Algebra and its Applications*, 391, 2004

[75] A. Pruessner et D. P. O'Leary. Blind déconvolution using a regularized structured total least norm. *SIAM Journal on Matrix Analysis and Applications*, 24(4), 2002.

[76] M. Welk, D. Theis, T. Brox et J. Weickert. PDE-based déconvolution with forward-backward diffusivities and diffusion tensors. In *Scale Space*, p. 585-597, 2005.

[77] J. Weickert. *Anisotropic diffusion in image processing*. ECMI Series, 1998.

[78] E. Salari et S. Zhang. Neural network-based image restoration using scaled residual with space-variant regularization. *International Journal of Imaging Systems and Technology*, 12, 2003.

[79] J.A. Hogbom. Aperture synthesis with a non-regular distribution of interferometer baselines. *Astronomy and Astrophysics Supplement Series*, 15, 1974.

[80] P. Magain, F. Courbin et S. Sohy: Déconvolution with correct sampling. *Astrophysical Journal*, 494, 1998.

[81] Z. Telatar. Adaptive filter design for image deblurring by using multi-criteria blurred image information. *Digital Signal Processing*, 15(1), 2005.

[82] V. Katkovnik, K. Egiazarian et J. Astola. A spatially adaptive nonparametric regression image deblurring. *IEEE Transactions on Image Processing*, 14(10), 2005.

[83] V. Katkovnik, K.Egiazarian et J. Astola. Adaptive window size image denoisingbased on intersection of confidence intervals ICI rule. *Journal of Mathematical Imaging And Vision*, 16(3), 2002.

[84] A. Goldenshluger et A. Nemirovski. On spatial adaptive estimation of nonparametric

regression. *Mathematical Methods of Statistics*, 6, 1997.

[85] M. A. Figueiredo et R. D. Nowak. An EM algorithm for wavelet-based image restoration. *IEEE Transactions on Image Processing*, 12(8), 2003.

[86] H. Takeda, S. Farsiu et P. Milanfar. Deblurring using regularized locally adaptive kernel regression. *IEEE Transactions on Image Processing*, 17(4), 2008.

[87] R. G. Lane et R. H. T. Bates. Automatic multidimensional déconvolution. *Journal of the Optical Society of America A*, 4(1), 1987.

[88] G. Ayers et J. C. Dainty. Iterative blind déconvolution method and its applications. *Optics Letters*, 13(7), 1988.

[89] D. L. Snyder, T. J. Schultz et J. A. O'Sullivan: Deblurring subject to nonnegativity constraint. *IEEE Transactions on Signal Processing*, 40(5), 1992.

[90] T. J. Holmes. Blind déconvolution quantum-limited incoherent imagery: maximum likelihood approach. *Journal of the Optical Society of America A*, 9(7), 1992.

[91] B. C. McCallum. Blind déconvolution by simulated annealing. *Optics Communications*, 75(2), 1990.

[92] R. G. Lane. Blind déconvolution of speckle images. *Journal of the Optical Society of America A*, 9(9), 1992.

[93] Y.L. You et M. Kaveh: A simple algorithm for joint blur identification and image restoration. In *International Conference on Image Processing*, p. 167-171, 1994.

[94] T.F. Chan et C.K.Wong: Total variation blind déconvolution. *IEEE Transactions on Image Processing*, 7(3), 1998.

[95] Y.L. You et M. Kaveh: Blind image restoration by anisotropic diffusion. *IEEE Transactions on Image Processing*, 8(3), 1999.

[96] L. He, A. Marquina et S.J. Osher. Blind déconvolution using TV regularization and Bregman iteration. *International Journal of Imaging Systems and Technology*, 15(1), 2005.

[97] A. Marquina. Inverse scale space methods for blind déconvolution. *Rapport technique*, Université de Californie - Los Angeles, 2006.

[98] L. Bar, N. Sochen et N. Kiryati. Semi-blind image restoration via Mumford-Shah regularization. *IEEE Transactions on Image Processing*, 15(2), 2006.

[99] G. Aubert et P. Kornprobst. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*. Applied Mathematical Sciences. Springer-Verlag, 2002.

[100] L. Chen et K.H. Yap. A soft double regularization approach to parametric blind image déconvolution. *IEEE Transactions on Image Processing*, 14(5), 2005.

[101] Y.L. You et M. Kaveh. A regularization approach to joint blur identification and image restoration. *IEEE Transactions on Image Processing*, 5(3), 1996.

[102] D. Kundur et D. Hatzinakos. Blind image déconvolution. *IEEE Signal Processing magazine*, 5 1996.

[103] M.K. Ng, R.J. Plemmons et S. Qiao. Regularization of RIF blind image déconvolution. *IEEE Transactions on Image Processing*, 9(6), 2000.

[104] G. Spaletta et L. Caucci. Constrained iterations for blind déconvolution and convexity issues. *Journal of Computational and Applied Mathematics*, 197(1), 2006.

[105] S. Colonnese, P. Campisi et G. P. G. Scarano. Blind image deblurring driven by nonlinear processing in the edge domain. *EURASIP Journal on Applied Signal Processing*, 2004(16), 2004.

[106] C. Vural et W. A. Sethares. Recursive blind image déconvolution via dispersion

- minimization. *International Journal of Adaptive Control and Signal Processing*, 19(8), 2005.
- [107] C. Vural et W. A. Sethares. Blind image déconvolution via dispersion minimization. *Digital Signal Processing*, 16(2), 2006.
- [108] S. Umeyama. Blind déconvolution of blurred images by use of ICA. *Electronics and Communications in Japan, Part 3*, 84(12), 2001.
- [109] I. Kopriva, D. J. Garrood et V. Borjanovic. Single frame blind image déconvolution by non-negative sparse matrix factorization. *Optics Communications*, 266(2), 2006.
- [110] I. Kopriva. Approach to blind image déconvolution by multiscale subband decomposition and independent component analysis. *Journal of the Optical Society of America A*, 24(4), 2007.
- [111] S. Gabarda et G. Cristobal. An evolutionaty blind image déconvolution algorithm through the pseudo-Wigner distribution. *Journal of Visual Communication and Image Representation*, 17(5), 2006.
- [112] F. Sroubek et J. Flusser. Multichannel blind iterative image restoration. *IEEE Transactions on Image Processing*, 12(9), 2003.
- [113] M. Sorel et J. Flusser. Space-variant restoration of images degraded by camera motion blur. *IEEE Transactions on Image Processing*, 17(2), 2008.
- [114] S. Babacan, J. Wang, R. Molina, and A. Katsaggelos. Bayesian blind déconvolution from differently exposed image pairs. In *Proc. IEEE Int. Conf. Image Process.*, Cairo, Egypt, Jul. 2009, pp. 133–136.
- [115] N. Joshi, R. Szeliski, and D. Kriegman. Psf estimation using sharp edge prediction. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, p. 1 - 8.
- [116] L. Yuan, J. Sun, L. Quan, and H.Y. Shum. Image deblurring with blurred/noisy image pairs. In *Proc. SIGGRAPH Conf.*, New York, 2007, p. 1.
- [117] P. E. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *Proc. 24th Annu. Conf. Comput. Graph. Interact. Techniques*, New York, 1997, p. 369 - 378.
- [118] A. Rav-Acha and S. Peleg: Two motion-blurred images are better than one. *Pattern Recognit. Lett.*, vol. 26, no. 3, p. 311 - 317, 2005.
- [119] M. Tico and M. Vehvilainen. Image stabilization based on fusing the visual information in differently exposed images. In *Proc. IEEE Int. Conf. Image Process.*, Oct. 19, 2007, vol. 1, p. 117 - 120.
- [120] J. Chen, L. Yuan, C.-K. Tang, and L. Quan. Robust dual motion deblurring. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, p.1 - 8.
- [121] F. Sroubek and J. Flusser. Multichannel blind déconvolution of spatially misaligned images. *IEEE Trans. Image Process.*, vol. 14, no. 7, p. 874 - 883, Jul. 2005.
- [122] F. Sroubek, G. Cristobal, and J. Flusser. A unified approach to superresolution and multichannel blind déconvolution. *IEEE Trans. Image Process.*, vol. 16, no. 9, p. 2322 - 2332, Sep. 2007.
- [123] J. Miskin. Ensemble learning for independent component analysis. Phd. dissertation, Astrophys. Group, Univ. Cambridge, 2000.
- [124] S. D. Babacan, R. Molina, and A. Katsaggelos. Parameter estimation in TV image restoration using variational distribution approximation. *IEEE Trans. Image Process.*, vol. 17, no. 3, p. 326 - 339, Mar. 2008.
- [125] R. Molina, J. Mateos, and A. K. Katsaggelos. Blind déconvolution using a variational approach to parameter, image, and blur estimation. *IEEE Trans. Image Process.*, vol. 15, no.

12, p. 3715 - 3727, Dec. 2006.

[126] M. E. Tipping and C. M. Bishop. Bayesian image super-resolution. In *Advances in Neural Information Processing Systems 15 (NIPS)*. Cambridge, MA: MIT Press, 2003.

[127] S. Babacan, R. Molina, and A. Katsaggelos. Variational Bayesian blind déconvolution using a total variation prior. *IEEE Trans. Image Process.*, vol. 18, no. 1, p. 12 - 26, Jan. 2009.

[128] A. Kanemura, S.-I. Maeda, and S. Ishii. Superresolution with compound markov random fields via the variational EM algorithm. *Neural Netw.*, vol. 22, no. 7, p. 1025 - 1034, 2009.

[129] A. Levin, Y. Weiss, F. Durand, and W. Freeman. Understanding and evaluating blind déconvolution algorithms. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, p. 1964 - 1971.

[130] M. Elad and A. Feuer. Restoration of a Single Super resolution Image from Several Blurred, Noisy, and Undersampled Measured Images. Senior Member, *IEEE Transactions on image processing*, Vol. 6, No. 12, December 1997.